Get Off My Prefix!
Using Gerontocratic Policies to Improve the Security and Stability of Internet Routing

Edmund L. Wong and Vitaly Shmatikov
Dept. of Computer Sciences, The University of Texas at Austin
{elwong,shmat}@cs.utexas.edu

Abstract

Inter-domain routing in today's Internet is plagued by security and reliability problems such as prefix hijacking and route oscillation. The reasons include Byzantine faults (often caused by misconfiguration) and deliberate actions by malicious autonomous systems.

We propose a simple family of dynamic route selection policies which can be adopted unilaterally by any autonomous system (AS), do not make major infrastructural assumptions, and do not require each AS to have an accurate model of the AS connectivity graph. In contrast to static policies, which prefer one route to another regardless of the dynamics of the routing protocol, our policies are *gerontocratic*: they take into account observed route lifetimes and can combine them with static preferences, if necessary.

We empirically demonstrate that, if adopted, these policies would be as effective at avoiding prefix hijacks as other solutions while yielding significantly more stable routes. Furthermore, we show that (a) static route selection policies cannot guarantee route convergence in the presence of Byzantine faults, while (b) gerontocratic policies guarantee convergence for non-Byzantine participants.

1 Introduction

The Internet consists of numerous domains or autonomous systems (ASes) which rely on the Border Gateway Protocol (BGP) [26] to establish inter-domain routes connecting them. One of the most important features of BGP is the almost complete autonomy it gives individual ASes in selecting the route to any destination from among the routes advertised by its neighbors.

Inter-domain routing has become a major source of security and reliability problems in the Internet. Route updates are not authenticated, and it is difficult for the recipient of an update to verify that the AS advertising a route to a certain part of the IP address space (identified by a prefix) has the right to do so. Prefix "hijacks" [2]—caused either by misconfiguration or malicious actions—resulted in a number of very visible incidents, including the AS 7007 incident in 1997 [3], the hijacks of Yahoo's prefixes by a Malaysian ISP in 2004 [22], of over 106,000 prefixes by a Turkish ISP in December 2004 [18, 22], of Google's prefix by Cogent in May 2005 that lasted for almost 2 days [18, 33], of 41 prefixes by RCN in January 2006 [18], and the infamous hijack of YouTube by Pakistan Telecom in February 2008 [30]. Unauthorized route advertisements, both long- and short-lived, are often associated with spam activity [22, 25]. Finally, misconfigurations and Byzantine faults can cause an AS to oscillate between routes, preventing BGP from converging and resulting in route instability [20].

A number of proposals aim to improve the security and stability of inter-domain routing. Techniques such as S-BGP [15, 16], soBGP [34], psBGP [32], and SPV [13] enable cryptographic validation of route updates. They have not found wide adoption due to their infrastructural requirements (*e.g.*, existence of a global PKI in the case of S-BGP), the need for ASes to cooperate with each other in order to deploy the solution, and other logistical challenges. Furthermore, while cryptographic authentication of route updates prevents malicious hijacks of IP address space, it does not ensure route stability since a Byzantine AS can switch between different authorized routes and thus cause major oscillations in the routes selected by other ASes. Cooperation between

ASes is also required by other BGP security mechanisms [23, 24, 29, 38, 39]. Purely local methods for detecting anomalies in BGP route updates [17, 28, 40] require ASes to maintain an accurate model of the AS connectivity graph and/or prefix ownership and can suffer from false positives.

Our contributions. First, we present a new family of *dynamic* route selection policies. These policies are purely local; adopting them does not require any global infrastructure or inter-AS cooperation. Existing policies used by ASes to choose among routes advertised by their peers are based on static preferences (*e.g.*, [8]). By contrast, our policies take into account the observed dynamics of route establishment.

Our policies are *gerontocratic* because they incorporate information about each route's "age," *i.e.*, the length of time it has been continuously available. This information can be combined with static preferences determined by peering and customer-provider business relationships, providing a simple, low-overhead policy routing solution which requires minimal modifications to routers and yields significant benefits even if adopted independently by an individual AS.

Second, we empirically evaluate the performance of gerontocratic policies using historical Route Views data [1]. We demonstrate that, if adopted, gerontocratic policies would have resulted in much more stable routes, with average route lifetimes almost an order of magnitude longer than those associated with other policies and, in fact, close to optimal. Gerontocratic policies significantly outperform Pretty Good BGP (PGBGP) [14], which simply depreferences routes originating from unknown ASes. Furthermore, we demonstrate that in the cases of actual hijack events, an AS using a gerontocratic policy almost never chooses a hijacked route.

Third, we show that dynamic policies are *necessary* for route stability by proving that a single Byzantine AS can prevent route convergence with *any* nontrivial combination of static policies (*i.e.*, a combination of static policies in which at least one AS prefers an indirect route to a particular destination), even if they satisfy the conditions which are sufficient for convergence in an all-rational environment. Therefore, robustness against Byzantine faults requires the route selection policy of each AS to incorporate the observed behavior of other ASes into the route selection algorithm.

Fourth, we show that a simple gerontocratic policy which chooses routes based simply on their age is guaranteed to achieve convergence even in the presence of Byzantine ASes.

The rest of the paper is organized as follows. We cover related work in Section 2. We describe our model of inter-domain routing in Section 3, show how static policies fail in the presence of Byzantine ASes in Section 4, and describe gerontocratic policies that are robust against Byzantine faults in Section 5. In Section 6, we demonstrate that gerontocratic policies greatly increase route stability while effectively avoiding hijacked routes. Section 7 concludes.

2 Related Work

Stability-based approaches. Gerontocratic policies may appear superficially similar to other techniques intended to improve route stability, such as route flap damping (RFD) [31], minimum route advertisement interval (MRAI) timers [26], and withdrawal rate-limiting (WRATE). RFD assigns to every neighboring AS and prefix a penalty which increases when a route flaps (*e.g.*, is advertised or withdrawn) and decays exponentially over time. If a route flaps too quickly, route updates from this AS/prefix are suppressed until the penalty decays past some low watermark. MRAI/WRATE suppress advertisements/withdrawals of new routes from a particular AS or prefix until a certain interval elapses.

RFD, MRAI, and WRATE are all-or-nothing: they either completely suppress the selection, advertisement, and withdrawal of a route, or they do nothing to depreference it. They do not guarantee route convergence and provide little benefit beyond reducing the number of repeated advertisements and withdrawals of unstable routes. We show in Section 6 that, in contrast to gerontocratic policies, adding RFD to a shortest-path policy does not substantially improve stability of the resulting routes. Furthermore, in pathological cases a few flaps may result in routes being suppressed for a long time, possibly resulting in a loss of connectivity [21].

Stability of popular Internet routes was observed in [27]. Stable Route Selection (SRS) considers stability as part of the local route preferences in order to avoid short-lived route instabilities [9]. Instead of using shortest path length as a tiebreaker, SRS suggests sticking with the current route, the shortest route, and then the route with the longest uptime if two routes have the same local preference.

While SRS and other previous work on route stability focused on empirical performance arguments,

we both demonstrate the practical benefits of our approach *and* argue that route stability is fundamentally necessary for BGP convergence in the presence of Byzantine misbehavior. We give a family of simple route selection policies that combine route stability with local preferences, and show that these policies tolerate Byzantine faults in the control plane. We also emphasize that convergence with Byzantine faults, at whatever cost, *cannot* be achieved without dynamic policies.

Prefix-hijack defenses. Pretty Good BGP (PGBGP) is based on the observation that prefix-hijack attacks are often ephemeral [14]. Therefore, PGBGP temporarily depreferences route advertisements that contain ASes which did not originate route advertisements to this prefix or any of its sub-prefixes in recent history. PGBGP guarantees neither convergence, nor stability in the presence of Byzantine ASes. As we show in Section 6, gerontocratic policies significantly outperform PGBGP in terms of route longevity and are equally effective in avoiding transient hijacks. It is also worth mentioning that, unlike gerontocratic policies, PGBGP cannot avoid hijacks that last longer than a preset interval, such as Cogent's hijack of Google's prefix [33, 18].

PHAS [18] collects control-plane data from BGP feeds and logs in order to detect suspicious changes in a prefix's origin AS. Other approaches use multiple vantage points in the data plane to acquire and cross-check fingerprints of selected destinations, such as route hop counts [12, 39]. These techniques require cooperation between ASes. If a hijack affects a significant fraction of the Internet, the victim network may be able to detect the attack by probing various destinations and checking how many replies are routed back correctly [37]. These data-plane techniques are vulnerable to intelligent adversaries who can recognize and re-route probes. They also do not prevent other ASes from selecting hijacked routes to the victim. Most importantly, they focus strictly on detecting prefix hijacks rather than improving route stability in general; they are thus complementary to our policy-based approach and can be used alongside it.

Cryptographic validation of route updates. Several proposals aim to secure BGP against invalid route advertisements [13, 16, 32, 34]. Some of them consider route stability but only as an aid to reduce route authentication costs [4] rather than a fundamental principle of route selection. In general, cryptographic validation of route updates is not sufficient for convergence because oscillation can be caused by policy conflicts even if all advertised routes are legitimate. Furthermore, these techniques require cooperation between ASes (*e.g.*, shared keys) and a cryptographic infrastructure such as a global PKI or a "web of trust."

By contrast, gerontocratic policies are local and provide significant benefits even if adopted by a single AS. They are compatible with virtually any proposed method for securing route updates and would still be beneficial (by ensuring route convergence) even if one of the secure BGP solutions were deployed throughout the Internet.

Incentive-compatibility. There has been a lot of research on conditions under which static route selection policies ensure BGP convergence. The most prominent are the Gao-Rexford conditions [8] and the "no dispute wheel" [11], which is implied by Gao-Rexford [7]. If every AS in the network is rational, their route selection policies do not conflict, and ASes are prevented from falsely advertising routes that were not advertised to them, BGP is guaranteed to converge to a stable set of routes [10, 19]. In this setting, BGP is incentive-compatible, *i.e.*, ASes derive no rational benefit from advertising routes other than those they actually prefer.

The assumption that every AS is rational may not hold in today's Internet. Unintentional misconfiguration may cause an individual AS to misbehave in a Byzantine fashion, preventing convergence [20]. Prefix hijacking and, in general, advertisement of non-existent or unavailable routes invalidate one of the conditions required for incentive-compatibility to hold. We leave formalization of incentive-compatibility in the setting where route stability is explicitly considered as part of ASes' route selection policies to future work.

3 Model

We use a simplified model of inter-domain routing. Consider a graph of N ASes. We assume that for each destination d, every AS has a route selection *policy* that prefers some routes to d over others. Unlike previous work, which assumed that route preferences are static, we allow preferences to evolve in response to observed route dynamics.

Formally, we say that at time t, every AS v has some preference ρ_v over the routes to d. ρ_v is a function that assigns an integer value to each route R; $\rho_v(R,t) > \rho_v(R',t)$ iff v prefers R to route R' (we

will drop the subscript v when the AS is clear from the context). For simplicity, we assume that an AS is connected to each of its peers by a single link; in Section 5, we discuss how to extend our metrics to the case of ASes that have multiple links between them.

We assume that some of the ASes in the network may be *Byzantine*. A Byzantine AS may arbitrarily deviate from BGP. In particular, it may advertise or withdraw any route, including routes that do not exist or have not been advertised to it by its neighbors. It may also advertise different routes to different neighbors. We assume that any AS (except the destination; this does not restrict the generality of our approach) may be faulty in this way, and the identity of the faulty AS is not known to other ASes. It is worth noting that in the real Internet, even fairly large ASes can suffer from Byzantine faults due to misconfiguration [33].

Since any AS may be faulty, we do not consider the situation when some AS x completely relies on another AS y to reach d (*e.g.*, when x is connected to the rest of the Internet via y). It is obvious that in this situation, no routing protocol can protect x from y's Byzantine misbehavior. Thus, we focus on ASes that have multiple disjoint paths to d.

The main theoretical property we are interested in is *convergence*. Our objective is to define a set of route selection policies such that, if all non-Byzantine ASes follow these policies, the routing protocol eventually produces a set of stable routes for the non-Byzantine ASes which are not affected by the Byzantine ASes' (mis)behavior. While we cannot provide a theoretical guarantee of convergence when our policies are adopted by a single AS, we demonstrate that even in this case they yield a substantial improvement in the security and stability of chosen routes (see Section 6).

We make the simplifying assumption that the only faults in the network are those caused by Byzantine misbehavior of some AS. Naturally, link failures may necessitate route changes even in the absence of a Byzantine AS. This is unavoidable. What we want to avoid is oscillation caused by policy conflicts [11] and/or faulty route advertisements inserted by a Byzantine AS. In the presence of link failures not caused by a Byzantine AS, the best one can hope for is that the non-faulty ASes converge to a set of routes which remains stable until a "genuine" link failure occur.

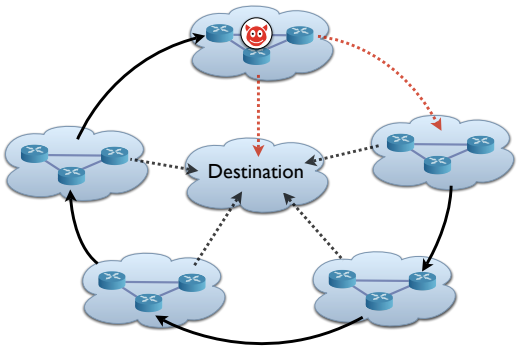

Figure 1: The standard policy dispute wheel. Solid/dotted lines denote the preferred/alternate path. The Byzantine AS (denoted by the devil icon) has no preferences; however, the path it chooses to advertise can prevent convergence in the network.

4 Static Policies

A route selection policy is *static* if the preference order it imposes on routes does not depend on the observed dynamics of the routing protocol. We give a simple argument that no combination of static policies is robust in the presence of a single Byzantine AS (the only exception is the trivial policy in which all ASes are directly connected to the destination and prefer the direct link over any indirect route). The intuition is simple: if the next hop in a preferred indirect route continuously advertises and withdraws its own route, the AS that prefers this route will never converge.

Below, we use parentheses to indicate concatenation, *e.g.*, (v, R) is AS v prepended to route R, and (R, R') is the concatenation of routes R and R'.

Theorem 4.1. *In the presence of a single Byzantine AS, the only network topology and static route selection policies that converge to a stable set of the routes are the ones in which every AS has a direct link to d and prefers this route over any indirect route.*

Proof. Suppose there exists some AS v which prefers to use a route (w, R_w), where $w \neq d$ and R_w is a route to d that does not contain v, over another route R_v which does not contain w. If w is Byzantine, then it can prevent convergence as follows. When v advertises (v, R_w), w advertises (w, R_v) to v, causing it to abandon R_w (since the route through w now contains a loop) and switch to R_v. Then w advertises R_w to v, causing it to abandon R_v and go back to R_w. This leads to continuous oscillation [11].

Thus, convergence cannot be guaranteed whenever some AS prefers an indirect route. Static policies which prefer the direct route are trivially stable, completing the proof. □

Theorem 4.1 shows that with a single Byzantine AS, *policy dispute wheels*, or routing loops caused by route selection policies, can be easily formed unless all ASes prefer direct routes. Figure 4 shows an example of a policy dispute wheel formed by a Byzantine AS. The actual impact of a Byzantine AS w on the network depends heavily on the number of ASes that directly or indirectly prefer routes that include w, since these are the ASes whose routes w can cause to oscillate. As noted earlier, large ASes have failed in this way before [33]; in such cases, many ASes would be affected by such "misbehavior." Moreover, the routes w is advertising may not even be legitimate. Unfortunately, even if cryptographic solutions such as S-BGP prevent ASes from advertising invalid routes, stability is not guaranteed since a Byzantine AS can block convergence by advertising an authorized indirect route and introducing a dispute wheel.

5 Gerontocratic Policies

We now introduce gerontocratic route selection policies, which consider the length of time that a route has been available when choosing among routes. If adopted by non-Byzantine participants, they guarantee convergence of BGP to a stable set of routes in the presence of Byzantine ASes.

As shown in Section 4, no static route selection policy guarantees convergence when Byzantine ASes are present. Thus, any Byzantine-tolerant policy must be *dynamic*: it must take into account the behavior of ASes and associated links when selecting routes. We posit two design principles for Byzantine-tolerant policies:

1. Withdrawing a route should incur some penalty in its preference by other ASes. Otherwise, a Byzantine AS could repeatedly withdraw and instantaneously re-advertise a desirable route, causing other ASes to withdraw and re-advertise their own routes.

2. A route that has been available for a sufficiently long time should have a higher preference than a shorter-lived route. In the absence of this requirement, a Byzantine AS could introduce never-seen-before, possibly even fake routes and cause previously stable, long-lived routes to be withdrawn, preventing eventual convergence. This requirement can be relaxed if ASes are prevented from advertising unauthorized routes.

Gerontocratic route selection is a simple example of a Byzantine-tolerant policy. At some point in time t, let $\ell(R,t)$ be the amount of time that a route has been continuously propagated to the given AS (*i.e.*, advertised by its neighbor, and neither explicitly nor implicitly withdrawn since it was advertised). Let $\rho_0(R)$ be some bounded function representing this AS's static preferences, and α be some constant weight such that $0 < \alpha < 1$. Consider a family of policies based on the following route preference metric:

$$\rho(R,t) = \alpha\ell(R,t) + (1-\alpha)\rho_0(R) \qquad (1)$$

Policies based on this metric satisfy both of our design principles. Long-lived routes eventually achieve an ℓ value large enough to overcome any difference in ρ_0, while withdrawing a route immediately resets ℓ.

This metric is implementable in current routers, which already maintain the AS's static preferences, as well as alternate routes to a given destination (in Adj-RIBS-In, as described in [26]). A router needs to simply store an extra timestamp for each of these alternate routes, indicating when it was advertised. In ASes with multiple edge routers, a timestamp associated with the BGP advertisement must be exchanged along with the advertisements themselves in order for each router to come to the same policy decision.

We prove that an AS using this metric eventually converges to a stable route to a destination d even in the presence of Byzantine ASes, as long as there exists some downstream neighbor that:

1. Is the destination d itself, or

2. Eventually converges on a stable route.

Thus, if every non-Byzantine AS uses a gerontocratic route selection policy, network-wide convergence is guaranteed.

We first prove that a network in which all ASes follow a route policy based on (1) eventually converges.

Theorem 5.1. *Given route preferences based on* (1) *and a network consisting entirely of honest ASes using these preferences, every AS in the network eventually converges on some stable set of routes.*

Proof. By induction on the minimum number of hops from d. First consider some AS $v \in \mathcal{N}^d$. Since v is

d's neighbor and has a direct physical link to d, there exists a route R_1 of the form (v, d). Consider some indirect route R_2 that v prefers over (v, d). In order for R_2 to remain preferred over R_1 at any time t, $\rho(R_2, t) > \rho(R_1, t)$ and thus

$$\alpha\ell(R_2, t) + (1-\alpha)\rho_0(R_2) > \alpha\ell(R_1, t) + (1-\alpha)\rho_0(R_1)$$

Moving the terms around, we have

$$\ell(R_2, t) > \ell(R_1, t) + K \tag{2}$$

for $K = (1-\alpha)/\alpha\,(\rho_0(R_1) - \rho_0(R_2))$; note that K is constant with respect to time. Since we assume no link failures in the non-Byzantine part of the network (see Section 3), $\ell(R_1, t)$ is increasing during every unit of time, *i.e.*, $d\ell(R_1, t)/dt = 1 \geq d\ell(R_2, t)/dt$. There must exist some time t_+ where the right-hand side of (2) is greater than 0; at this point, (2) holds for all $t \geq t_+$ only if $\ell(R_2, t) > \ell(R_1, t) + K > 0$, *i.e.*, if R_2 is not withdrawn again. Thus, if R_2 is withdrawn at some point $t \geq t_+$, v converges on R_1; otherwise, v converges on R_2.

Now suppose that every AS within distance j eventually converges on a particular route. We now prove that every AS within distance $j+1$ must converge as well. Consider some ASes v and w such that $v \in \mathcal{N}^w$, v is $j+1$ hops away from d, and w is j hops away from d. By the induction hypothesis, w must eventually converge on some route R_3 to d. From this point on, v has a stable route $R_4 = (v, R_3)$. By the same argument as before, there is some point at which, for any route R_5 that v prefers over R_4, a withdrawal would cause R_4 to be preferred. If R_5 is withdrawn after then, v converges on R_4; otherwise, v converges on R_5. □

We now show that these policies can tolerate a Byzantine AS who may be trying to create route instabilities.

Theorem 5.2. *Given route preferences based on* (1)*, every non-Byzantine AS eventually converges on a stable set of routes in the presence of one Byzantine AS.*

Proof. (Sketch) Similar to Theorem 5.1. Since every AS v has multiple disjoint routes to a particular destination d, there must be at least one route that does not involve a Byzantine AS. Every AS on the non-Byzantine path must eventually converge on some stable route. Similar to the proof of Theorem 5.1, if after some point in time the Byzantine route is withdrawn, v converges on the non-Byzantine route; otherwise, v converges on either (stable) route. □

Theorem 5.2 can be easily generalized to show that in the presence of more than one Byzantine AS, policies that use (1) as a metric converge on a stable set of routes as long as every AS has some route to d that does not involve a Byzantine AS.

Corollary 5.3. *Given route preferences based on* (1) *and the presence of multiple Byzantine ASes, every non-Byzantine AS in any network eventually converges on some stable set of routes as long as every non-Byzantine AS has some route to the destination that does not involve a Byzantine AS.*

Proof. (Sketch) Proof is similar to the proof of Theorem 5.1 and 5.2. Eventually, every non-Byzantine AS v will have some stable non-Byzantine route available to it. If, after some point in time, some other preferred route is available and remains stable, v converges on this route; otherwise, v converges on the stable non-Byzantine route. □

Theorem 5.2 and Corollary 5.3 assume that no physical link failures necessitating route updates occur during the convergence period, *i.e.*, the only source of instability is the behavior of the Byzantine AS(es). If physical link failures occur infrequently, there will still be periods of stability from the time when routes with no Byzantine ASes achieve higher preference (according to ρ) than those with Byzantine ASes until one of the links in these non-Byzantine routes fails.

ASes may want to occasionally probe their statically preferred routes if they believe that misconfigurations, faults, etc. may be correctable. This can be accomplished by decaying a route's ℓ score at certain intervals.

Although our family of route selection policies ensures eventual convergence, the above theoretical analysis guarantees neither how long convergence takes nor the quality of the resulting routes (it is worth noting that in certain network topologies, route flap damping, which is an example of a very simple dynamic policy, can lead to pathological increases in the duration of convergence and outright loss of connectivity [21]). Furthermore, Byzantine ASes can delay convergence by delaying the withdrawal of routes in which they are involved. Even without Byzantine ASes, convergence time will depend on the policies of other ASes, on how much they choose to bias their own static preferences (the α parameter), and on the network topology. Nevertheless, in Section 6 we show that, empirically, gerontocratic policies tend to per-

form very well on realistic topologies and produce very stable routes.

Our policies do guarantee a stable set of routes. Assuming independent link failures, our metric tends to prefer shorter routes, which have a lower likelihood of failing. There is no theoretical guarantee, however, that the resulting routes correspond to actual physical paths or that they are consistent with what other ASes have advertised. As mentioned in Section 2, there are many mechanisms in the literature that address these issues and are complementary to our approach [14, 16, 34]. It has been empirically observed that invalid or inconsistent advertisements are often transient and short-lived [2, 14]. As we show in Section 6, gerontocratic policies are very successful at filtering out invalid route advertisements without any cryptographic mechanisms or the need to cooperate with other ASes.

Our route selection metric allows an AS to strongly bias its route selection towards its static preferences. Nevertheless, our policies do restrict the freedom an AS has in selecting routes by requiring it to take the route's age into account. This is necessary for convergence. In reality, route selection policies may be governed by business considerations which override even the need for convergence. This is beyond the scope of our theoretical analysis, since our goal is to show the conditions under which convergence can be achieved.

6 Evaluation

In Section 5, we showed that gerontocratic policies assure route convergence for non-Byzantine ASes if adopted universally. In this section, we empirically demonstrate that they provide substantial benefits even if adopted unilaterally by a single AS, without any cooperation from other ASes. By simulating the behavior of a gerontocratic AS in response to actual Internet route updates, we show that gerontocratic policies yield routes that are significantly more stable than those selected by other policies and are less likely to include routes advertised by hijackers.

6.1 Implementation and setup

We implemented a multithreaded discrete event simulator in order to simulate the behavior of various routing policies. Our simulator, available for download at [35], models an AS receiving route updates from its neighboring peers and selecting routes to various prefixes. We use routing updates and table dumps from the Route Views project [1], specifically from `route-views2.oregon-ix.net`, a node that has approximately 52 neighboring peer ASes including AT&T, Level3, Sprint, and UUNet.

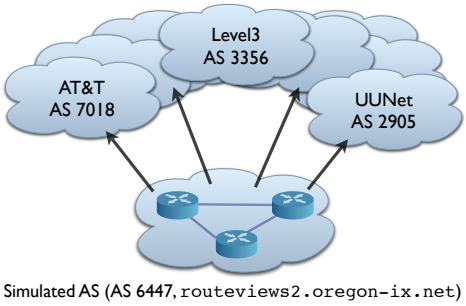

Figure 2: The simulation setup. Using data from the Route Views project, we simulate AS 6447 as an edge network that is selecting routes.

For a given year y, we generate the initial routing table by first loading the last table dump from year $y-1$ and updating the table with route updates whose timestamps lie between the route table's timestamp and the beginning of year y. We finish initializing the routing table by setting all timestamps to the first timestamp of year y; we do this because the entries of the routing table dump have timestamps that do not seem to correspond to when a route was first advertised.[1]

We begin the simulation by selecting a route from the initial routing table in accordance with various route selection policies. We continue processing route updates and selecting new routes until we reach updates from year $y+1$. We update the route table entry if and only if a route advertised by a particular AS has changed, *i.e.*, we do not change an entry's timestamp if the route being advertised is the same as previously advertised. Additionally, for *Damped Shortest*, we update the route flap damping penalty even if the route has not changed.

As in previous work (*e.g.*, [36]), we rely on Route Views data, which reflects actual updates as observed on the Internet. One limitation of completely relying on Route Views is that we cannot model the effects of our simulated AS advertising routes to its peers since

[1] In other words, if some AS advertises some route r during time t and then advertises r again at time $t' > t$, the corresponding routing table entry in the table dump does not always have a timestamp of t.

the Route Views AS does not advertise any routes on its own. Thus, our simulation effectively models an edge AS which joins the network at the beginning of the simulation. Figure 2 illustrates this setup.

We implement and test the following policies:

- *Gerontocratic*: Prefer oldest routes first; break ties with route length, then advertising peer's AS number.
- *Shortest (AS)*: Prefer shortest routes first; break ties with peer's AS number.
- *Shortest (Age)*: Same as *Shortest (AS)*, except use route's age as the tie-breaking metric before peer's AS number. Hybrid of *Gerontocratic* and *Shortest (AS)*.
- *Damped Shortest*: Same as *Shortest (AS)*, except use route flap damping with parameters set to Cisco's default as listed in [21].
- *PGBGP Lite* [14]: Same as *Shortest (AS)*, except depreference, for s hours, all routes to a given prefix that do not contain an AS which originated a route advertisement for this prefix in the last h days. Our route lifetime simulations only simulate route selection to a destination prefix, not the path that is chosen for a particular IP address. Thus, we do not implement the sub-/super-prefix hijack checks described in [14]; we discuss this simplification in Section 6.3. We use $h = 10$ and $s = 24$ as in [14].
- *Random*: Randomly generate static local preferences among peer ASes at the beginning of a simulation, which are then used to prefer routes advertised by some peers over others. These policies are known as next-hop policies and are commonly studied (*e.g.*, [5]) and deployed. Note that these policies capture the well-known customer/provider/sibling/peer relationship [6]. The results from this policy are the average of simulating 100 different static preferences.
- *Static Optimal*: Prefer peers that advertise the longest-living routes to a particular prefix. This policy is unimplementable because it assumes that the choosing AS knows how long the routes advertised by a given peer will survive after having been chosen. We include it as it defines the upper bound on how well any policy can maximize route lifetimes if local preferences among peer ASes must be statically fixed ahead of time.

Policy name	Avg. lifetime (days)	Avg. # hops
Gerontocratic	82.68	2.92
Shortest (Age)	69.85	2.23
Shortest (AS)	9.57	2.23
Damped Shortest	9.83	2.23
PGBGP Lite	9.57	2.23
Random	38.68	3.35
Static Optimal*	187.64	3.26
Optimal*	221.59	3.02

Table 1: Summary of policies tested and their average route lifetimes and number of hops.

- *Optimal*: Prefer the route that will last the longest. This unimplementable policy defines the upper bound on how well any policy, static or dynamic, can do in maximizing route lifetimes.

We simulate route selection for popular destinations in the US (Google [64.233.160.0/23]; Microsoft [207.46.192.0/18], Germany (GMX [213.165.64.0/19]), Brazil (Universo Online [200.98.192.0/18]), and China (QQ [60.28.0.0/15]). We simulated years 2005 to 2008 individually and entirely, with the exception of 2007, where we simulated up to Dec. 16 due to corruption in the Route Views data.

6.2 Route lifetimes and lengths

For each policy, we simulate a series of route updates and record the route selected by a particular policy. For every year, destination, and policy combination, we calculate:

- Average route lifetime, calculated by taking the amount of time that our simulated AS had a route to a particular destination and dividing by the number of times the policy switched to a different route; and
- Weighted average route length, calculated by taking the number of hops in each of the routes selected by the policy, multiplying by the time during which this was the chosen route, and dividing by the time that our simulated AS had a route to a particular destination.

Table 1 shows the overall results of the tested policies across all destinations and years. Figures 4 and 5 illustrate the lifetimes achieved by the policies for

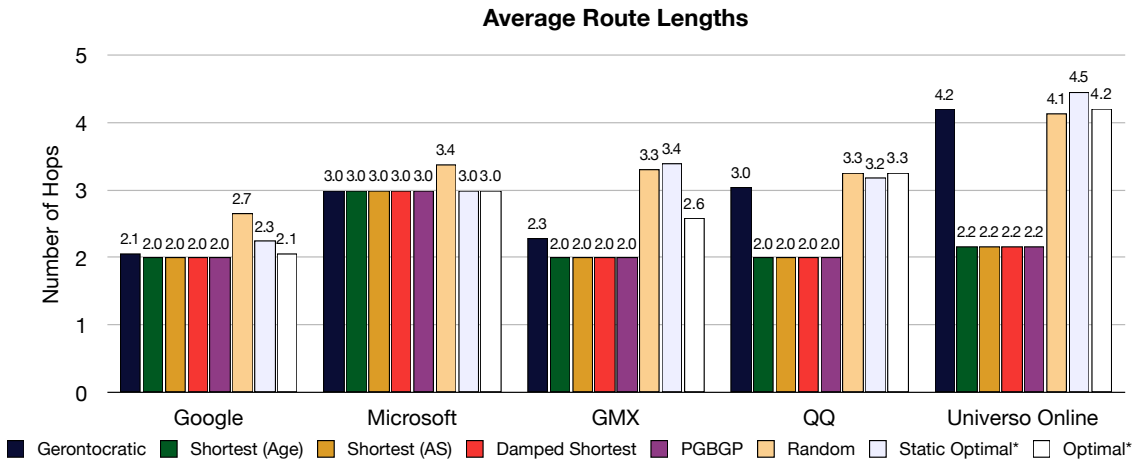

Figure 3: Lengths of routes selected by various policies to various destinations, averaged over the years.

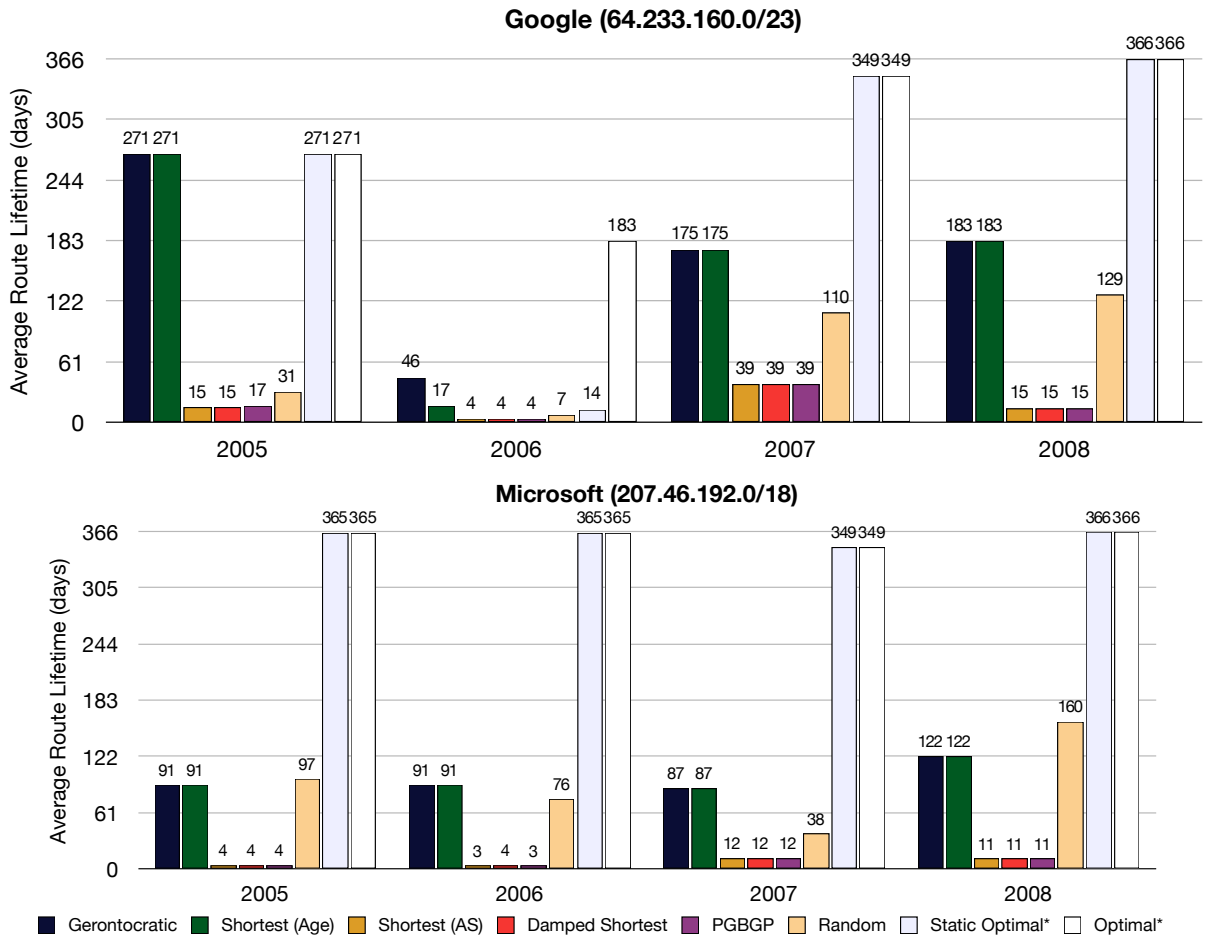

Figure 4: Route lifetimes for various policies, US prefixes, and years. Note that there were no routes to 64.233.160.0/23 until sometime in April.

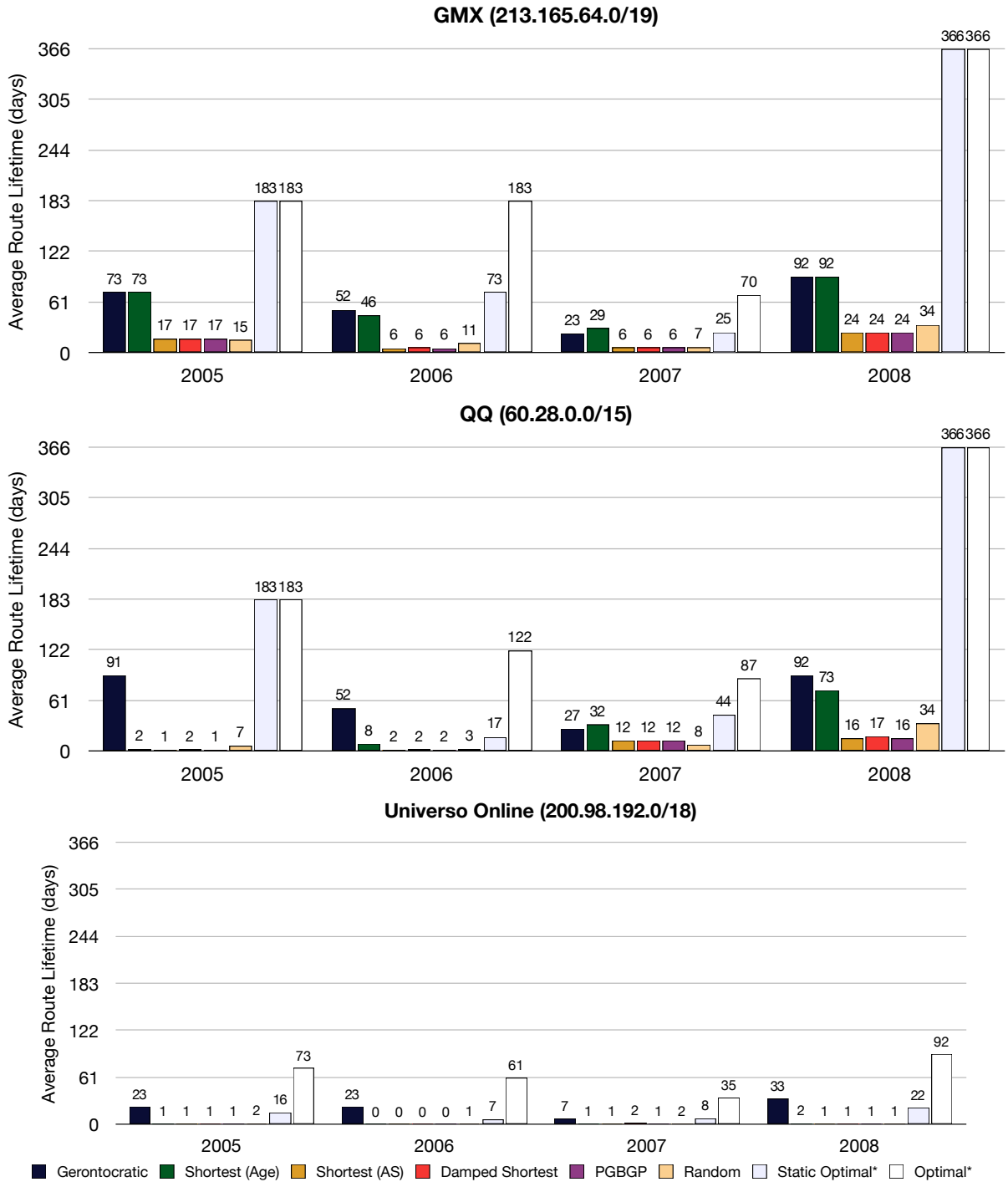

Figure 5: Route lifetimes for various policies, international prefixes, and years.

each destination and year. Figure 3 shows, for each policy, the route lengths to each destination, averaged over the years. We use asterisks (*) to denote policies that are unimplementable in practice, as they require perfect knowledge of future events.

Comparison with shortest-path policies. For the destinations and time periods tested, *Gerontocratic* selects routes that last, on average, over 8 times longer than *Shortest (AS)*, *Damped Shortest*, or *PGBGP Lite*. These policies greedily switch to the "latest and greatest" shortest route, regardless of how long it has been advertised. Although they always select the shortest routes, they frequently have to change their selection when their choices are withdrawn.

Although both *Damped Shortest* and *PGBGP Lite* employ techniques that use limited dynamic information to avoid short-term route instability, they are still vulnerable to longer-term instability, causing them to select shorter-lived routes on average. Our simulations indicate that route flap damping provides only a marginal benefit over *Shortest (AS)*, providing an extra 7-8 hours of route lifetime on average. Moreover, *PGBGP Lite* provides no benefit over *Shortest (AS)* during periods when there are no hijacks occurring. Since most prefix hijacks are short-lived and relatively uncommon, *PGBGP Lite*'s ability to avoid them provides little benefit with respect to route lifetime.

Overall, *Gerontocratic* significantly outperforms *Shortest (AS)*, *Damped Shortest*, and *PGBGP Lite*. For Google or Microsoft, *Gerontocratic* selects routes that last 10 times longer than *Shortest (AS)*, *Damped Shortest*, or *PGBGP Lite*. *Gerontocratic* benefits from the existence of routes that are both short and stable, selecting routes that were on average only 0.06 hops longer to Google and equal length to Microsoft. For GMX, *Gerontocratic* selects routes that lasted 6 times longer yet were only 0.29 hops longer.

Gerontocratic achieves even better improvements in route stability when selecting routes to Universo Online, where its selections last on average 20 times longer than those selected by *Shortest (AS)*, *Damped Shortest*, and *PGBGP Lite*. Although the routes selected by the latter policies are around 2 hops shorter than those selected by *Gerontocratic*, the shortest-length routes are, in this case, often the shortest-living ones as well, given that *Static Optimal* and *Optimal* both select long-living routes that have more hops than those selected by *Gerontocratic*. Similarly, *Gerontocratic* selects routes to QQ that lasted 8 times longer but, like *Static Optimal* and *Optimal*, are one hop longer.

Applying gerontocratic metrics on top of shortest-path routing provides significant improvement in route lifetime. By considering the age of a route as a tie-breaker between two equally long routes, *Shortest (Age)* is able to avoid long-term route flapping that plagued *Shortest (AS)*, *Damped Shortest*, and *PGBGP Lite*. *Shortest (Age)* performs comparably to *Gerontocratic* when choosing a path to Google, Microsoft, or GMX and performs slightly better than *Gerontocratic* on occasion. This is because these destinations have many short routes that were also stable. By converging on routes with fewer hops, which has a minor effect on the likelihood of a route being withdrawn, and using gerontocratic policies to select the more stable routes, *Shortest (Age)* is competitive with *Gerontocratic* and is able to provide significant improvement over other shortest-path policies, while still selecting the shortest routes on average.

When selecting routes to QQ and Universo Online, however, *Shortest (Age)* performs noticeably worse than *Gerontocratic*. These destinations, located in China and South America, have both longer and more diverse routes available. As described above, the available shorter routes turn out to be less stable than their longer counterparts. As a result, the gerontocratic component of *Shortest (Age)* is used less frequently, and the pitfalls of optimizing for shortest path are once again prominent.

Comparison with network engineering and oracle policies. Many ASes perform network engineering in order to prioritize certain neighbors over others. As described above, we use *Random* to simulate next-hop policies with different static, local preferences. The optimal set of local preferences for any given year and destination with respect to route lifetimes is represented by *Static Optimal*, which assumes oracle knowledge of future route stability.

Running 100 different random static preference policies, we observe that *Random*, through careful choices of which peer ASes to prefer, can achieve route lifetimes that exceed most shortest-path policies. Out of an average of 100 different random static policies, *Random* achieves average lifetimes of 38.68 days, roughly 4 times better than *Shortest (AS)*, *Damped Shortest*, and *PGBGP Lite*.

Unfortunately, it can be difficult to select the static preferences that will result in good performance. We measure the average route lifetimes achieved by individual ASes during each individual year that we

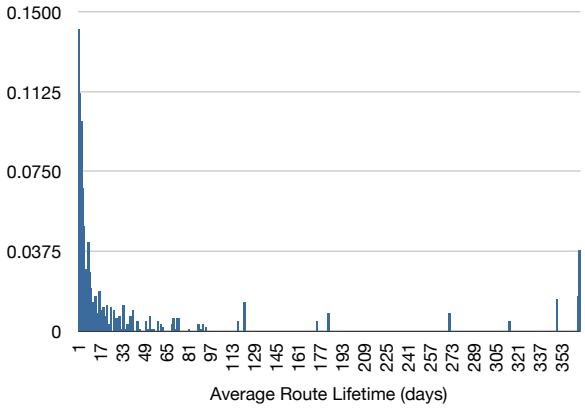

Figure 6: Probability density function of average route lifetimes achieved by peer ASes.

tested. As Figure 6 illustrates, over 14% of all ASes advertise routes that last for less than a day, and nearly 50% advertise routes that last less than 6 days. Route selection must be done carefully in order to achieve good performance, as many choices lead to very short-lived routes. On the other hand, *Gerontocratic* outperforms *Random* by more than a factor of 2 on average with minimal effort. Moreover, *Gerontocratic* typically selects routes that are shorter than those selected by *Random*.

Static Optimal does beat *Gerontocratic* during most simulations, which is not surprising since *Static Optimal* assumes perfect knowledge of future lifetimes of the routes advertised by any given peer. It may seem odd that *Gerontocratic* occasionally outperforms *Static Optimal*. The reason is that *Static Optimal* is based on static preferences among peers, whereas long-lived routes may be advertised by different ASes at different times. In these scenarios, *Static Optimal*, with its fixed preferences, does not have the flexibility to adapt to the AS that is currently more stable, whereas *Gerontocratic*, which only considers how long a route has been around and not which peer advertised it, does.

As expected, *Optimal* always outperforms all other policies. Overall, *Gerontocratic* performs within a factor of 1/3-1/2 of *Static Optimal* and *Optimal*, with similar, if not shorter, route lengths.

6.3 Prefix hijack avoidance

We next evaluate how well *Gerontocratic* avoids prefix hijacking attacks as compared to other policies. We run simulations for years 2004-2008, selecting routes to the destinations tested in Section 6.2 as well as four additional destinations: Google (64.233.161.0/24), YouTube (208.65.153.0/24), the University of Hong Kong (HKU) (147.8.0.0/16), and the Department of Computer Sciences at the University of Texas at Austin (UTCS) (128.83.0.0/16).

Table 2 shows the prefix hijacks that we observe in the data and summarizes how the various policies perform. Since most prefix hijacks are short-lived, *Gerontocratic* is able to avoid most of them by choosing older routes. Even in the case of Cogent's hijack of Google's prefix, which lasted over a day, the availability of other routes allow *Gerontocratic* to evade the attack.

On the other hand, since false route updates tend to advertise routes with fewer hops, policies that prefer shorter routes such as *Shortest (AS)*, *Damped Shortest*, and even *Shortest (Age)* are vulnerable to many of the hijacks we observed, including Cogent's hijack of Google; TTNet's hijack of Microsoft and UTCS; OJSC NW Telecom's hijack of HKU and UTCS; and Pakistan Telecom's hijack of YouTube. Note that *Random* is trivially vulnerable depending on the (randomly generated) preferences.

Although *PGBGP Lite* depreferences short-lived updates from unfamiliar origin ASes and thus avoids most hijacks, *PGBGP Lite* does fall victim to Cogent's hijack of Google, which lasted approximately 1.84 days, much longer than the 1-day window during which *PGBGP Lite* considers a route suspicious. The reason is that, much like route-flap damping, *PGBGP Lite* is all-or nothing: either a route is suspected and thus heavily depreferenced, or it is considered without any qualms. Although our approach does not explicitly suspect unknown ASes, gerontocratic metrics have a similar effect, resulting in *Gerontocratic* depreferencing and ignoring Cogent's false advertisements.

None of the policies we test are able to avoid the YouTube hijack; in fact, *no* unilaterally adopted policy is able to avoid this hijack. On Feb. 24, 2008, Pakistan Telecom took over a subset of YouTube's IP address space (208.65.152.0/22) by advertising 208.65.153.0/24, which was more specific than any other advertised prefix. Although every policy did select a non-hijacked route to 208.65.152.0/22, data packets are ultimately forwarded along the route to the most specific prefix that matches the destination IP address. In this case, all neighboring ASes advertised both a legitimate route to 208.65.152.0/22

Hijacker	Destination	Policies affected (excl. *Random*)	Duration
Cogent AS 174 - May 7, 2005	Google (64.233.161.0/24)	*Shortest (AS)*, *Damped Shortest*, *PGBGP Lite*	1.84 days
TTNet AS 9121 - Dec. 24, 2004	Google (64.233.161.0/24) Microsoft (207.46.192.0/18) UTCS (128.83.0.0/16)	— *Shortest (AS)*, ***Shortest (Age)***, *Damped Shortest* *Shortest (AS)*, ***Shortest (Age)***, *Damped Shortest*	49.3 min. 35.3 min. 24.8 min.
OJSC NW Telecom AS 8997 - Sep. 22, 2008	QQ (60.28.0.0/15) HKU (147.8.0.0/16) UTCS (128.83.0.0/16)	— *Shortest (AS)*, *Damped Shortest* *Shortest (AS)*, *Damped Shortest*	6.68 min. 6.55 min. 6.48 min.
Pakistan Telecom AS 17557 - Feb. 24, 2008	YouTube (208.65.153.0/24)	*All*	1.64 hrs.
IFX Comm. AS 18747 - Nov. 17, 2004	Google (64.233.161.0/24)	—	24 sec.
Hutchinson Global AS 9304 - Aug. 11, 2005	Microsoft (207.46.192.0/18)	—	8.67 min.
Columbus Net AS 23520 - Apr. 9, 2006	Microsoft (207.46.192.0/18)	—	29 sec.
INDOSAT AS 4761 - Nov. 30, 2006	Microsoft (207.46.192.0/18)	—	1.67 min.

Table 2: Hijacks (known and suspected) observed in simulations.

and the hijacked route to 208.65.153.0/24. Any traffic destined for 208.65.153.0/24 had to pass through a neighboring AS, which would use a hijacked route to forward the data.

For clarity of exposition, we presented simplified versions of both gerontocratic policies and *PGBGP Lite*. When selecting a route to a particular prefix, the simplified policies ignore advertisements of routes to the sub-prefixes. Such policies, which include all of the policies we tested, are susceptible to sub-prefix hijacking as a result of the most-specific-prefix forwarding rules. The complete PGBGP policy attempts to address this by (1) suspecting (and depreferencing) any route which is advertised for a sub-prefix of some known prefix p but does not contain a known origin AS, and (2) depreferencing all routes for p from neighboring ASes that have advertised the suspicious sub-prefix, even if p itself is not suspicious. As we explain above, these techniques would not have been sufficient to evade Pakistan Telecom's hijack of YouTube (unless also adopted by other ASes).

An equivalent extension to *Gerontocratic* provides it with a similar capacity to avoid sub-prefix hijacks by prioritizing updates for a particular prefix (and associated sub-/super-prefixes) depending on whether the origin AS is suspicious and/or disseminates long-lived route advertisements for this IP address space. The proof that this modification does not affect the theoretical convergence properties of *Gerontocratic* is straightforward and omitted for brevity.

7 Conclusion

In this paper, we assert that dynamic observations of the inter-domain routing protocol must be explicitly incorporated into the route selection policies of individual autonomous systems (AS) in order to tolerate Byzantine misbehavior by other ASes, which is an increasingly common problem in the Internet. We propose a family of simple dynamic policies which take into account both local, static preferences and observed protocol dynamics. Unlike static policies, our policies are guaranteed to converge on a stable set of routes in the presence of Byzantine ASes. Modeling the behavior of our policies on actual Internet route updates, we demonstrate that they result in very stable routes and ameliorate the effects of prefix-hijacking attacks without cryptographic infrastructure, cooperation from other ASes, or local knowledge of the Internet topology.

References

[1] University of Oregon Route Views project. `http://www.routeviews.org`.

[2] H. Ballani, P. Francis, and X. Zhang. A study of prefix hijacking and interception in the Internet. In *SIGCOMM*, 2007.

[3] V. Bono. 7007 explanation and apology. `http://www.merit.edu/mail.archives/nanog/1997-04/msg00444.html`, 1997.

[4] K. Butler, P. McDaniel, and W. Aiello. Optimizing BGP security by exploiting path stability. In *CCS '06: Proceedings of the 13th ACM conference on Computer and communications security*, pages 298–310, New York, NY, USA, 2006. ACM.

[5] J. Feigenbaum, R. Sami, and S. Shenker. Mechanism design for policy routing. In *PODC '04: Proceedings of the twenty-third annual ACM symposium on Principles of distributed computing*, pages 11–20, New York, NY, USA, 2004. ACM.

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

[7] L. Gao, T. Griffin, and J. Rexford. Inherently safe backup routing with BGP. In *INFOCOM*, 2001.

[8] L. Gao and J. Rexford. Stable internet routing without global coordination. *IEEE/ACM Trans. Netw.*, 9(6):681–692, 2001.

[9] P. B. Godfrey, M. Caesar, I. Haken, Y. Singer, S. Shenker, and I. Stoica. Stable Internet route selection. NANOG 40, 2007.

[10] S. Goldberg, S. Halevi, A. Jaggard, V. Ramachandran, and R. Wright. Rationality and traffic attraction: Incentives for honest path announcements in BGP. In *SIGCOMM*, 2008.

[11] T. G. Griffin, F. B. Shepherd, and G. Wilfong. The stable paths problem and interdomain routing. *IEEE/ACM Trans. Netw.*, 10(2):232–243, 2002.

[12] 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.

[13] Y.-C. Hu, A. Perrig, and M. Sirbu. SPV: Secure path vector routing for securing BGP. In *Proc. SIGCOMM*, 2004.

[14] J. Karlin, S. Forrest, and J. Rexford. Pretty Good BGP: Improving BGP by cautiously adopting routes. In *ICNP*, 2006.

[15] S. Kent, C. Lynn, J. Mikkelson, and K. Seo. Secure Border Gateway protocol (S-BGP) — real world performance and deployment issues. In *Proc. NDSS*, 2000.

[16] S. Kent, C. Lynn, and K. Seo. Secure Border Gateway protocol (Secure-BGP). *IEEE Journal on Selected Areas in Communications*, 18(4), 2000.

[17] C. Kruegel, D. Mutz, W. Robertson, and F. Valeur. Topology-based detection of anomalous BGP messages. In *Proc. RAID*, 2003.

[18] M. Lad, D. Massey, D. Pei, Y. Wu, B. Zhang, and L. Zhang. PHAS: a prefix hijack alert system. In *USENIX-SS'06: Proceedings of the 15th conference on USENIX Security Symposium*, Berkeley, CA, USA, 2006. USENIX.

[19] H. Levin, M. Schapira, and A. Zohar. Interdomain routing and games. In *STOC*, 2008.

[20] R. Mahajan, D. Wetherall, and T. Anderson. Understanding BGP misconfiguration. In *SIGCOMM*, 2002.

[21] Z. M. Mao, R. Govindan, G. Varghese, and R. H. Katz. Route flap damping exacerbates internet routing convergence. In *SIGCOMM*, 2002.

[22] C. D. Marsan. Six worst internet routing attacks. `http://www.networkworld.com/news/2009/011509-bgp-attacks.html`, Jan. 15 2009.

[23] A. Mizrak, Y. Cheng, K. Marzullo, and S. Savage. Fatih: Detecting and isolating malicious routers. In *Proc. DSN*, 2005.

[24] S. Qiu, F. Monrose, A. Terzis, and P. McDaniel. Efficient techniques for detecting false origin advertisements in inter-domain routing. In *Proc. NPsec*, 2006.

[25] A. Ramachandran and N. Feamster. Understanding the network-level behavior of spammers. In *SIGCOMM*, 2006.

[26] Y. Rekhter and T. Li. A border gateway protocol (BGP-4). `http://www.ietf.org/rfc/rfc1771.txt`, 1995.

[27] J. Rexford, J. Wang, Z. Xiao, and Y. Zhang. Bgp routing stability of popular destinations. In *IMW '02: Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment*, pages 197–202, New York, NY, USA, 2002. ACM.

[28] G. Siganos and M. Faloutsos. Neighborhood watch for internet routing: Can we improve the robustness of internet routing today? In *Proc. INFOCOM*, 2007.

[29] L. Subramanian, V. Roth, I. Stoica, S. Shenker, and R. Katz. Listen and Whisper: Security mechanisms for BGP. In *Proc. NSDI*, 2004.

[30] P. Svensson. Pakistan causes worldwide YouTube outage. `http://www.msnbc.msn.com/id/23339712`, Feb. 25 2008.

[31] C. Villamizar, R. Chandra, and R. Govindan. BGP route flap damping. `http://www.ietf.org/rfc/rfc2439.txt`, 1998.

[32] T. Wan, E. Kranakis, and P. van Oorschot. Pretty secure BGP (psBGP). In *Proc. NDSS*, 2005.

[33] T. Wan and P. van Oorschot. Analysis of BGP prefix origins during Google's May 2005 outage. In *2nd International Workshop on Security in Systems and Networks*, 2006.

[34] R. White. Secure Origin BGP (soBGP). `ftp://ftp-eng.cisco.com/sobgp/index.html`, 2002.

[35] E. L. Wong and V. Shmatikov. Edge AS policy simulator. `http://www.cs.utexas.edu/~elwong/edge-policy-sim`.

[36] W. Xu and J. Rexford. MIRO: Multi-path interdomain routing. *SIGCOMM Comput. Commun. Rev.*, 36(4):171–182, 2006.

[37] 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.

[38] X. Zhao, D. Pei, L. Wang, D. Massey, A. Mankin, F. Wu, and L. Zhang. Detection of invalid routing announcement in the Internet. In *Proc. DSN*, 2002.

[39] C. Zheng, L. Ji, D. Pei, J. Wang, and P. Francis. A light-weight distributed scheme for detecting IP prefix hijacks in real-time. In *Proc. SIGCOMM*, 2007.

[40] D. Zhu, M. Gritter, and D. Cheriton. Feedback based routing. In *Proc. HotNets*, 2002.