Route Bazaar: Automatic Interdomain Contract Negotiation

Ignacio Castro[♣†♠] Aurojit Panda[‡] Barath Raghavan[♠] Scott Shenker[♠‡] Sergey Gorinsky[♣]

[♣]IMDEA Networks Institute [†]Open University of Catalonia [♠]ICSI [‡]UC Berkeley

Abstract

While it is widely acknowledged that the Border Gateway Protocol (BGP) has many flaws, most of the proposed fixes focus solely on improving the stability and security of its path computation. However, because interdomain routing involves contracts between Autonomous Systems (ASes), this paper argues that contractual and routing issues should be tackled jointly. We propose Route Bazaar, a backward-compatible system for flexible Internet connectivity. Inspired by the decentralized construction of trust in cryptocurrencies, Route Bazaar uses a decentralized public ledger and cryptography to provide ASes with automatic means to form, establish, and verify end-to-end connectivity agreements.

1 Introduction

It's Friday night at Ballroom B at the Sheraton in Albuquerque, NM. In the vast room dwarfing the few dozen people gathered within, pairs of lonely eyes meet up to decide whether they would be a good match. Such speed dating events are about connecting, but the connections being discussed tonight are not of the usual sort because the participants are Internet Service Providers (ISPs), members of an Internet eXchange Point (IXP) attending an event organized to help them negotiate peering agreements with other IXP participants. *This is how network interconnection contracts are negotiated in the Internet of 2015.*

How did we arrive at this sorry state? After infrastructure privatization in the early 1990s, the Internet consisted of multiple independent networks or Autonomous Systems (ASes). Global end-to-end reachability thus required the negotiated interconnection of separately owned and administered networks. Delivering the traffic of another network is a service that has its costs and can be provided with different levels of Quality of Service (QoS). As private for-profit entities, ASes seek compensation for their traffic delivery services and sign contacts that include a Service-Level Agreement (SLA). To establish conditions and accountability, the SLA determines the interconnection type, service conditions, compensation arrangements, and penalties for contractual violations.

During the same transition in the 1990s, the main protocol for interdomain routing changed from the centralized Exterior Gateway Protocol (EGP) [31] to the distributed BGP [20]. In BGP, each AS independently decides which traffic-delivery paths it will offer its neighbors. An AS only announces paths to those neighbors with whom the AS has a bilateral interconnection agreement, even when the available physical connectivity is richer.

This interconnection model enabled the Internet's rapid growth, but BGP's distributed realization of bilateral contracts suffers from numerous drawbacks, including route instability, convergence slowness, policy inflexibility, and configuration complexity. While protocol improvements, new contract types, and new interconnection facilities have mitigated some of these issues, the foundations of Internet routing remain mostly the same and many problems persist. Most current BGP research focuses on improvements in the basic protocol, rather than on the contractual system that BGP realizes. In contrast, we contend that many of BGP's woes arise due to the contractual model.

End-to-end traffic delivery requires coordination among multiple ASes that may not trust each other. The current contractual system deals with the problem of trust by relying on rigid bilateral contracts between neighboring ASes. However, the transitive trust of the bilateral contracts comes at the expense of routing flexibility and efficiency. Without explicit means for direct coordination among multiple networks, the realized paths are suboptimal, oscillate needlessly, and react inappropriately to traffic-demand changes and infrastructure failures.

This paper proposes Route Bazaar, a contractual system where ASes and their customers agree on QoS-aware routes in the absence of preexisting trust relations between the networks. The only trusted entity in Route Bazaar is a public ledger, which is assumed to be trustworthy in the absence of large corrupted coalitions. While prior work on cryptocurrencies showed how to construct a decentralized public ledger without a single trusted component, our use of the ledger enables networks to check the previous record of each participant in a path before agreeing on the path. Because the public ledger of Route Bazaar allows anyone to verify the trustworthiness of a network as a routing provider or customer, the verification mechanism incentivizes honest behavior by all networks and thereby supports effective interdomain routing in an untrusted environment.

2 Background

Current **interdomain routing** relies on bilateral contracts between ASes, which typically establish either transit or peering relationships. In a transit relationship, a customer network pays a provider for reaching the global Internet. Networks can also bypass transit providers and instead interconnect directly through a peering relationship. In peering, two networks obtain reachability to a restricted set of Internet addresses: peers only exchange traffic destined to their own networks and to networks for which they provide transit.

An AS uses BGP to exchange reachability information with neighbors with which it has bilateral contracts. An AS has limited control over the routes of inbound traffic; for outbound traffic, when multiple neighbors offer paths to the same destination, the AS can choose the neighbor through which to forward. Most ASes prefer to send outbound traffic through interconnections that generate revenue (*i.e.,* through transit customers), and avoid using transit providers if an alternate path through a customer or peer exists.

BGP is fully decentralized and is thus extremely scalable. However, this scalability comes at the expense of flexibility and responsiveness. In particular, events such as routing policy changes, misconfigurations, and infrastructure failures can trigger path oscillations and forwarding loops [18], and it is nearly impossible to verify that selected paths conform with global routing policies [16]. For example, BGP hijacking has in the past been used to redirect YouTube's traffic to a fake destination in Pakistan [2, 30], and to redirect Spamhaus's traffic to a hacker group [35]; in both of these cases a more global view would have alerted ASes to these problems.

The rapid growth of the Internet, both in terms of users and traffic volumes, and the increasing diversity of Internet services have strained the existing routing framework. To cope with these challenges, new types of interconnection contracts have emerged (*e.g.,* partial transit [37], paid peering [8] and remote peering [6]), IXPs (switching facilities to reduce the costs of peering) have mushroomed, and BGP has slowly evolved.

However, many of the limitations of Internet routing stem from two root causes. First, contracts must be explicitly negotiated by humans before two ASes can exchange traffic. Thus, in stark contrast to the dynamic nature of the Internet itself, interconnection negotiations are carried out at human-time scales (sometimes via social events for engineers as mentioned earlier). Second, these contracts are applied recursively: traffic that an AS sends to its neighbor is then governed by the contracts of that neighbor. The local (and thus recursive) routing decisions made by BGP are globally suboptimal due to limited visibility [22].

Cryptocurrencies, like Bitcoin [24], are secure, decentralized, anonymous digital currencies. These currencies are often built on a public ledger, commonly referred to as a block chain. The public ledger records transactions and enables checking of the account balance for each user. Cryptocurrencies have been adopted to new non-monetary uses [7, 13, 14] that leverage the public ledger as a decentralized and hard-to-corrupt log.

The public ledger needs to be resilient (*i.e.,* incorruptible) even in the presence of untrusted participants. Bitcoin's public ledger is therefore built using a consensus algorithm that is capable of solving the Byzantine consensus problem [1]. Byzantine consensus is impossible to solve in asynchronous systems in general and requires at least two thirds of participants be honest. Moreover, it assumes that the set of participants is well known. Bitcoin solves this latter problem (that of limiting who can participate) by requiring each participant in the consensus algorithm to solve a puzzle before voting, and attach a proof-of-work [23] to each vote. Generating this proof requires the participant to solve a sufficiently hard algorithmic problem[1]. Since generating this proof requires computational resources, it ensures that the number of votes a malicious user can generate is in proportion to the computing power they control. Since voting multiple times is hard, a malicious user needs to instead convince a majority of participants to affect the result of the consensus algorithm. The lack of aligned interests among malicious parties therefore allows all users to trust the values stored in the block chain.

3 Route Bazaar

In this section, we present Route Bazaar, a novel system for flexible Internet connectivity. Inspired by cryptocurrencies, Route Bazaar uses a decentralized public ledger to allow mutually distrustful ASes and customers to establish dynamic, end-to-end QoS-aware paths as overlays on the existing Internet. In Route Bazaar, the information contained in the public ledger allows each participant to verify any participant's conformance with previous path agreements, while simultaneously keeping the path agreements private. By relying on a public ledger, Route Bazaar establishes trust among participants which can compute the likelihood that another participant will honor a path agreement. Route Bazaar uses standard cryptographic tools to ensure privacy and existing techniques to establish overlays; our innovation lies in creating a trustworthy environment for the announcement, selection, and verification of end-to-end paths. In what follows, we assume that all communications with the public ledger are carried out through authenticated and encrypted channels, *i.e.,* entities cannot impersonate each other. Several existing protocols (*e.g.,* TLS [9]) can be used to meet this requirement.

[1] Bitcoin changes the problem hardness to ensure that solving it takes a certain amount of time on average.

Provider	Pathlet	Destination	Price	Latency SLO	Throughput SLO
AS1	f48d4c4	AS2	$50	5 ms (99.9%)	3 Gbps (99.9%)
AS1	d7228c5	AS3	$45	5 ms (99.9%)	3 Gbps (99.9%)
AS2	97dbd13	AS9	$10	10 ms (99.9%)	1 Gbps (99.9%)
AS3	ca22b8a	AS9	$20	8 ms (99.9%)	2 Gbps (99.9%)

Table 1: Pathlet advertisements in the public ledger.

In Route Bazaar, a *provider* is an AS that advertises connectivity over a *pathlet* (*i.e.,* a path fragment [15]). A *path* is formed by composing pathlets leading from a *source* to a *destination*. A *customer* in Route Bazaar is an entity paying for the end-to-end connectivity provided by a path. A customer may not be an AS, and might be either the source or destination of the path.

An important aim for Route Bazaar's design is to minimize the amount of information leaked about end-to-end paths and customer policies. Our design limits public information to the minimum required to support flexible routing via multilateral contracts. First, providers advertise pathlets using the public ledger. Then, customers compose end-to-end paths by combining the advertised pathlets. Finally, providers confirm or reject the agreement. We envision that these decisions are made dynamically by automated agents acting on behalf of customers and providers. Participants can hence enforce sophisticated contractual and routing policies. For instance, Route Bazaar allows these policies to exploit the historical records about forwarding performance (to choose providers) and likelihood of payment (to accept a customer) that are maintained in the public ledger. When all participants agree on a path, the public ledger records an agreement between the customer and each provider participating in the path.

As the traffic is forwarded along a computed path, the source, destination and each provider record machine-readable forwarding proofs in the public ledger. These proofs can then be used to verify that each provider delivered the desired level of service. Customers also record proofs showing that they have paid providers in the public ledger. The public ledger therefore allows potential customers and providers to compare previous performance and payment history when deciding whether or not to trust each other.

3.1 Routing

Providers advertise *pathlets* in the public ledger (Table 1). Each pathlet advertisement specifies a tag (used to refer to the pathlet), the source, destination, price and Service Level Objective (SLO) for throughput and latency. For ease of exposition, here we assume that the pathlet provider is also the source, however Route Bazaar supports pathlets where the source and provider differ.

A customer composes an end-to-end path between a source and destination by choosing from the set of advertised pathlets. The customer can enforce routing policies by filtering out policy-incompatible pathlets. For example, a customer can exclude pathlets advertised by providers who have previously not met their SLOs. Before a path can be used, each provider must agree to route traffic along the path; a provider can thus disallow the use of policy-incompatible paths. For example, a provider might deny service to customers who are unlikely to pay, or reject paths involving untrusted providers. Policies are enforced by automated agents, who act on behalf of providers and customers (and are hence aware of their policies) and can exploit the information contained in the public ledger to judge other participants.

Provider	Tag	Latency SLO	Throughput SLO
AS1	enc_m(f48d4c4)	5 ms (99.9%)	1 Gbps (99.9%)
AS2	enc_m(97dbd13)	10 ms (99.9%)	1 Gbps (99.9%)

Table 2: Pathlet commitments table in the public ledger. $enc_m(x)$ here represents the value output by a PRF with key m and value x.

Customer	Pathlet	Payment
Alice	enc_n(f48d4c4)	enc_n($50)
Alice	enc_n(97dbd13)	enc_n($10)

Table 3: Payment commitments in the public ledger. $enc_n(x)$ here represents the value output by a PRF with key n and value x.

Once the customer and all pathlet providers have agreed on a path, the participants use a symmetric key generated via key agreement (*e.g.,* Elliptic Curve Diffie-Hellman (ECDH) [3]) and the pathlets' tags as input to a cryptographic pseudorandom function (PRF) to generate an anonymous tag that is valid for only this specific path. Each provider then publishes a pathlet agreement which includes this encrypted tag, the provider's identity and the SLO offered by the pathlet to a pathlet commitments table (Table 2) in the public ledger. We use path agreement to refer to the collection of all pathlet agreements that allow routing along a path. The customer and providers also agree on a second key that is used to generate an anonymized payment tag (again derived from the pathlet tag) and prices that are used to record a set of payment agreements (Table 3) between the customer and providers.

This mechanism can accommodate a variety of end-to-end routing models including multipath routing [38], source routing [28], opportunistic routing [28] and route repositories [4]. Route Bazaar also allows customers to outsource path computation to trusted third parties, *i.e.,* routing as a service [21]. Finally, Route Bazaar supports contractual flexibility, *e.g.,* it can accommodate both cases where the sender pays for connectivity and cases where the receiver pays for connectivity.

To illustrate how customers can use Route Bazaar to form an end-to-end path, consider a case where Alice wants to route traffic from a source in AS1 to a destination in AS9. Alice uses the pathlet announcements in

Pathlet	Packet	Hash	Timestamp	Throughput
enc_m(source|A6)	50	0a9f136	420 ms	1.2 Gbps
enc_m(f48d4c4)	50	0a9f136	424 ms	1.2 Gbps
enc_m(97dbd13)	50	0a9f136	433 ms	1.1 Gbps
enc_m(destination|A9)	50	0a9f136	433 ms	1.1 Gbps

Table 4: Forwarding proofs in the public ledger. $enc_m(x)$ here represents the value output by a PRF with key m and value x.

Table 1 to find two possible paths that provide this connectivity: AS1-AS2-AS9 and AS1-AS3-AS9. While the path through AS2 is cheaper ($60 vs $65), the path through AS3 offers better latency (13ms vs 15ms). In this example, Alice's policy favors the cheapest path[2], and she therefore decides to route along path AS1-AS2-AS9. If AS1 and AS2 agree to form a path, the three (Alice and both ASes) use ECDH to compute keys m and n. AS1 and AS2 use a PRF with key m to generate anonymized tags, and then update the pathlet commitments table in the public ledger (Table 2). Similarly, Alice uses key n to compute anonymized payment tags, and encrypt prices, and updates the payment commitment table (Table 3).

Note that while in the previous example, Route Bazaar allows Alice to exclude paths going through AS3, this is not generally possible in BGP, where all traffic originating at AS1 and destined to AS9 follows the same path (which might in fact go through AS3). Route Bazaar thus provides Alice with additional routing flexibility, allowing her to choose paths based on a richer set of policies.

3.2 Forwarding

Once an end-to-end path has been agreed, providers update the data plane as required. Route Bazaar provides the mechanisms to form and agree on paths as well as to verify that forwarding conforms with the agreed paths. To verify path conformance, Route Bazaar generates *forwarding proofs* that are recorded in the public ledger. Customers, providers and intermediate ASes along a path use existing techniques for traffic sampling at routers to periodically generate a forwarding proof. The forwarding proof for a pathlet includes the path-specific anonymized forwarding tag (a pathlet might be used by several paths), a sample of the traffic, the timestamp indicating when the sample was captured, and the throughput averaged over the time since the last sample. In our current design, we envision that each provider sets up Generic Route Encapsulation (GRE) tunnels [19] across each pathlet (to ensure in-order packet transit) and samples a particular packet (*e.g.,* the 50^{th} packet in Table 4). The hash of this packet is used as a traffic sample for the forwarding proof.

Note that the inclusion of timestamps allows participants to compute the latency reported by a pathlet's ingress and egress neighbors. Furthermore, the participants in a path (*i.e.,* the customer, source, destination and pathlet providers) can use their path key to discover bottlenecks in the path by observing where a (sampled) packet was dropped. To preserve the anonymity of a path, this information is not available to non-participants.

Pathlet	Paid
enc_n(f48d4c4)	Yes
enc_n(97dbd13)	Yes

Table 5: Payment proofs in the public ledger. $enc_n(x)$ here represents the value output by a PRF with key n and value x.

When a path's agreement concludes[3], each provider is paid by the customer, and the provider registers a payment conformation in the public ledger (Table 5). The payment proof includes the pathlet's anonymized payment tag and a field indicating whether the payment was made. Alternately, the customer can record its unwillingness to pay in the public ledger, indicating that appropriate connectivity was not provided.

The payment proofs in the public ledger enable anyone to check customers' payment history. These records can also be used for offline arbitration of payment disputes. During such arbitration, the entities involved in the contract can present the arbitrator with a deanonymized version of the forwarding and payment proofs.

In the example above, the participants, *i.e.,* the source in AS1, pathlet providers AS1 and AS2, and the destination in AS9, sample every 50th packet and publish forwarding proofs as shown in Table 4. These ASes rely on NTP (Network Time Protocol) for clock synchronization to ensure reported times are comparable. Once the path agreement has concluded, Alice pays AS1 and AS2, and they record her payment in the public ledger as shown in Table 5.

3.3 Privacy

The privacy offered by Route Bazaar is comparable to BGP. Similar to BGP, Route Bazaar reveals available paths (as all possible path compositions). This information is identical to what is available in public repositories, *e.g.,* CAIDA [5]. Furthermore, Route Bazaar does not require providers or customers to reveal routing preferences and policies.

However, unlike existing interdomain routing solutions, Route Bazaar also reveals anonymized forwarding and payment proofs. If deanonymized (*e.g.,* due to a compromised participant) these proofs expose the precise paths used by customers and the volume of transferred traffic. Similar information can be revealed today by sufficiently powerful adversaries (*e.g.,* governments or large ASes). Existing mechanisms, *e.g.,* Tor [10] and Unblock [33], for anonymizing source and destination addresses can be used on top of Route Bazaar to provide stronger anonymity.

[2] Alice could have decided using other policies, *e.g.,* prior history if available, or any other reasons.

[3] In our current design, *path agreements* are for a fixed volume of traffic.

Finally, the forwarding commitments table in Route Bazaar leaks information about which pathlets are popular, and the amount of overall traffic transmitted across a pathlet. Similarly, the payments commitment table leaks information about the number of agreements made by each customer. Route Bazaar can be extended to anonymize this information, so that contracts are established and verified out-of-band, with Route Bazaar merely serving as a record of past performance. Studying this extension and its properties is left to future work.

4 Discussion

Performance Overhead. Route Bazaar imposes minimal overhead on the data plane, requiring only that routers periodically sample traffic. This feature is commonly supported in most routers (to aid in debugging), and Route Bazaar does not require the samples to be transmitted in real-time. The communication overhead imposed by requiring ASes to record forwarding proofs with the public ledger is relatively small and can be controlled by adjusting the sampling rate. Furthermore, our current design also requires the use of GRE tunnels, these are supported by most existing interdomain routers.

Because decision making in Route Bazaar is not local, routers merely forward traffic and are not responsible for control-plane decisions. Instead, control-plane decisions can be done externally by computers, or at cloud computing facilities. The operations required to access and update Route Bazaar today is comparable to what is performed by a modern web browser when connecting to a website over HTTPS [29]. The primary computation overhead during path computations is therefore a function of the policy complexity, and Route Bazaar's control plane therefore imposes modest performance overheads.

Sybil Attacks. Participants can circumvent the trust mechanisms of the public ledger by creating pseudonymous identities, *i.e.,* they can perform a Sybil attack [11]. While ASes and large organizations (who are the main participants in Route Bazaar) are unlikely to jeopardize their reputation by forging their identities, Route Bazaar can protect against Sybil attacks by adopting existing solutions, *e.g.,* SybilGuard [39].

Rich routing policies. Route Bazaar can further enrich its supported routing policies, *e.g.,* by linking pathlet prices to the customers' requested traffic volume, payment history, or other conditions. To attract customers, the pathlets can also expose specific salient features of the provided connectivity, *e.g.,* its Software-Defined Networking (SDN) implementation. While Route Bazaar separates routing from forwarding, the main challenge in enriching the routing policies is not their storage or processing but their expression in a machine-readable language.

Backward compatibility. Because Route Bazaar can operate on top of today's Internet, it is backward compatible with traditional bilateral contracts and BGP routing. Still, Route Bazaar diversifies contractual options, *e.g.,* by enabling IXP members to exchange traffic not only through traditional peering agreements but also with contracts formed dynamically via the public ledger.

5 Related Work

Since Detour [32] there have been many proposals to make Internet connectivity more flexible (*e.g.,* AROW [26]). The main contribution of Route Bazaar is that it simultaneously addresses end-to-end routing and contracts, enabling adoption of previously proposed routing innovations. Among the prior work on interdomain routing, the most closely related to Route Bazaar are pathlet routing [15], ICING [25,34], and Platypus [27]. Pathlet routing introduces the concept of composing paths from pathlets; Route Bazaar adds the mechanism to advertise pathlets via the public ledger. Route Bazaar can adopt ICING and Platypus mechanisms to verify conformance of forwarding to routing; the usage of the public ledger reduces the data-plane changes needed for such adoption.

As in SDN, Route Bazaar cleanly separates the network data and control planes. Previous work, including RCP [12] and 4D [17], suggests such separation for interdomain routing.

Kadupul [36] uses a virtual currency to incentivize low-latency forwarding in wireless mesh networks. While both Kadupul and Route Bazaar are inspired by cryptocurrencies, Kadupul does not deal with routing or support end-to-end QoS.

6 Conclusions

The current Internet relies on explicitly negotiated bilateral agreements that are recursively applied via BGP, leading to rigid functionality and suboptimal routing behavior. In this paper, we propose Route Bazaar, an alternative that learns from cryptocurrencies to solve the decentralized trust problem inherent in connectivity contracts. Route Bazaar forms contracts for end-to-end Internet connectivity orders of magnitude faster and supports highly flexible routing.

7 Acknowledgments

This research was financially supported in part by the European Commission (FP7-ICT 288021, EINS, and H2020 644960, ENDEAVOUR), Regional Government of Madrid (S2013/ICE-2894, Cloud4BigData), and by NSF grant 1420064.

References

[1] C. Attiya, D. Dolev, and J. Gil. Asynchronous Byzantine Consensus. In *PODC*, 1984.

[2] P. Bangera and S. Gorinsky. Impact of Prefix Hijacking on Payments of Providers. In *COMSNETS*, 2011.

[3] E. Barker, L. Chen, A. Roginsky, and M. Smid. Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography. *NIST Special Publication 800-56A*, 2013.

[4] M. Caesar, M. Casado, T. Koponen, J. Rexford, and S. Shenker. Dynamic Route Recomputation Considered Harmful. *CCR*, 40(2):66–71, 2010.

[5] Cooperative Association for Internet Data Analysis (CAIDA). `http://www.caida.org/data/active/as-relationships`, 2015.

[6] I. Castro, J. C. Cardona, S. Gorinsky, and P. Francois. Remote Peering: More Peering without Internet Flattening. In *CoNEXT*, 2014.

[7] L. Chen and K. Chen. BitBill: Scalable, Robust, Verifiable Peer-to-Peer Billing for Cloud Computing. In *HotCloud*, 2014.

[8] A. Dhamdhere and C. Dovrolis. Twelve Years in the Evolution of the Internet Ecosystem. *ToN*, 19(5):1420–1433, 2011.

[9] T. Dierks and E. Rescorla. The Transport Layer Security (TLS) Protocol Version 1.2, *RFC 5246*, 2008.

[10] R. Dingledine, N. Mathewson, and P. Syverson. Tor: The Second-Generation Onion Router. In *USENIX Security*, 2004.

[11] J. R. Douceur. The Sybil Attack. In *IPTPS*, 2002.

[12] N. Feamster, H. Balakrishnan, J. Rexford, A. Shaikh, and J. Van Der Merwe. The Case for Separating Routing from Routers. In *FDNA*, 2004.

[13] Filecoin. `http://filecoin.io/`, 2015.

[14] M. Ghosh, M. Richardson, B. Ford, and R. Jansen. A TorPath to TorCoin: Proof-of-Bandwidth Altcoins for Compensating Relays. In *HotPETs*, 2014.

[15] P. Godfrey, I. Ganichev, S. Shenker, and I. Stoica. Pathlet Routing. In *SIGCOMM*, 2009.

[16] S. Goldberg, M. Schapira, P. Hummon, and J. Rexford. How Secure are Secure Interdomain Routing Protocols? In *SIGCOMM*, 2010.

[17] A. Greenberg, G. Hjalmtysson, D. A. Maltz, A. Myers, J. Rexford, G. Xie, H. Yan, J. Zhan, and H. Zhang. A Clean Slate 4D Approach to Network Control and Management. In *SIGCOMM*, 2005.

[18] A. Haeberlen, I. Avramopoulos, J. Rexford, and P. Druschel. NetReview: Detecting when Interdomain Routing Goes Wrong. In *NSDI*, 2009.

[19] S. Hanks, T. Li, D. Farinacci, and P. Traina. Generic Routing Encapsulation (GRE), *RFC 1701*, 1994.

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

[21] K. Lakshminarayanan, I. Stoica, S. Shenker, and J. Rexford. Routing as a Service. Technical Report UCB/EECS-2006-19, *UC Berkeley*, 2006.

[22] A. Lutu. *A System for the Detection of Limited Visibility in BGP*. PhD thesis, Carlos III University of Madrid, Spain, 2014.

[23] A. Miller and J. J. LaViola Jr. Anonymous Byzantine Consensus from Moderately-Hard Puzzles: A Model for Bitcoin. Technical Report CS-TR-14-01, *University of Central Florida*, 2014.

[24] S. Nakamoto. Bitcoin: A Peer-to-Peer Electronic Cash System. `http://bitcoin.org/bitcoin.pdf`, 2008.

[25] J. Naous, M. Walfish, A. Nicolosi, D. Mazieres, M. Miller, and A. Seehra. Verifying and Enforcing Network Paths with ICING. In *CoNEXT*, 2011.

[26] S. Peter, U. Javed, Q. Zhang, D. Woos, T. Anderson, and A. Krishnamurthy. One Tunnel is (Often) Enough. In *SIGCOMM*, 2014.

[27] B. Raghavan and A. C. Snoeren. A System for Authenticated Policy-Compliant Routing. In *SIGCOMM*, 2004.

[28] B. Raghavan, P. Verkaik, and A. C. Snoeren. Secure and Policy-Compliant Source Routing. *ToN*, 17(3):764–777, 2009.

[29] E. Rescorla. HTTP Over TLS, *RFC 2818*, 2000.

[30] RIPE. YouTube Hijacking: A RIPE NCC RIS Case Study. `http://goo.gl/jKX6Bz`, 2008.

[31] E. C. Rosen. Exterior Gateway Protocol (EGP), *RFC 827*, 1982.

[32] S. Savage, T. Anderson, A. Aggarwal, D. Becker, N. Cardwell, A. Collins, E. Hoffman, J. Snell, A. Vahdat, G. Voelker, and J. Zahorjan. Detour: Informed Internet Routing and Transport. *Micro*, 19(1):50–59, 1999.

[33] W. Scott, R. Cheng, J. Li, A. Krishnamurthy, and T. Anderson. Blocking-Resistant Network Services using Unblock. `http://unblock.cs.washington.edu/unblock.pdf`, 2012.

[34] A. Seehra, J. Naous, M. Walfish, D. Mazieres, A. Nicolosi, and S. Shenker. A Policy Framework for the Future Internet. In *HotNets*, 2009.

[35] A. Shaw. Spam? Not Spam? Tracking a Hijacked Spamhaus IP. `http://goo.gl/GE0c05`, 2013.

[36] M. Skjegstad, A. Madhavapeddy, and J. Crowcroft. Kadupul: Livin' on the Edge with Virtual Currencies and Time-Locked Puzzles. *arXiv:1412.4638*, 2014.

[37] V. Valancius, C. Lumezanu, N. Feamster, R. Johari, and V. Vazirani. How Many Tiers? Pricing in the Internet Transit Market. In *SIGCOMM*, 2011.

[38] D. Wendlandt, I. Avramopoulos, D. G. Andersen, and J. Rexford. Don't Secure Routing Protocols, Secure Data Delivery. In *HotNets*, 2006.

[39] H. Yu, M. Kaminsky, P. B. Gibbons, and A. Flaxman. SybilGuard: Defending Against Sybil Attacks via Social Networks. In *SIGCOMM*, 2006.