Provable Security of S-BGP and other Path Vector Protocols: Model, Analysis and Extensions

Alexandra Boldyreva
Georgia Institute of Technology, Atlanta, USA
sasha.boldyreva@cc.gatech.edu

Robert Lychev
Georgia Institute of Technology, Atlanta, USA
rlychev@cc.gatech.edu

ABSTRACT

This paper provides the provable-security treatment of path vector routing protocols. We first design a security definition for routing path vector protocols by studying, generalizing, and formalizing numerous known threats. Our model incorporates three major security goals. It is quite strong, yet simple to use. We prove by reduction that S-BGP satisfies two out of the security model's three goals, assuming the underlying signature scheme is secure. Under the same assumption, we next show how the protocol can be modified to meet all three security goals simultaneously. Finally, we study security of partial PKI deployment of path vector protocols when not all nodes have public keys. We investigate the possibilities of relaxing the PKI requirement and relying on the non-cryptographic physical security of the protocol in order to achieve possibly weaker, but still well-defined, notions of security. We also present the necessary and sufficient conditions to achieve full security in the partial PKI deployment scenario. We believe our conclusions will prove useful for protocol developers, standards bodies and government agencies.

Categories and Subject Descriptors

C.2.2 [**Network Protocols**]: Routing Protocols, Protocol Verification

Keywords

Secure BGP, path vector protocols, provable security

1. INTRODUCTION

MOTIVATION AND RELATED WORK. The Border Gateway Protocol (BGP) is currently the de facto standard for routing across the Internet. Its current version, version 4, is defined in a draft standard [44] and is in wide use. In the protocol, each router associated with a particular autonomous system (AS)—an independent network managed by a single administrative entity such as a Content Provider or an Internet Service Provider (ISP)—maintains a list of possible paths to various IP prefixes. Information about serviceable prefixes is advertised to neighboring AS's who propagate it to their neighbors, and so on, so that reachability information is updated globally.

BGP was designed to enable routing between parties that trust each other and thus it lacks security features. Nowadays, however, commercial interests invalidate the assumption of trust on the Internet. Accordingly, the security of BGP has come under much scrutiny [21, 38, 12, 40] because honest failures or malicious router compromises may cause serious problems throughout the Internet. For example, on April 25, 1997 an incorrect route map was injected into the Internet forcing most Internet traffic to be routed to a small Internet Service Provider (ISP) in Virginia, crippling much of the Internet for about two hours [13]. Similar misconfiguration have been recently documented for Pakistan and China [20, 25], as we will discuss later in the paper. There is a widespread agreement that due to increased importance of the Internet, it is extremely important to ensure security of its infrastructure. The Department of Homeland Security views BGP security as part of the national strategy for securing the Internet [41].

Vast related research, including [32, 50, 7, 33, 51, 49, 22, 23], incorporate additional measures to handle authenticity/ integrity and authorization issues in BGP. In particular, a major security vulnerability, such as lack of integrity of the route announcements, has been addressed. Secure BGP (S-BGP) protocol [34, 35] stands out as the most comprehensive attempt to secure the Internet's routing infrastructure to date. It is currently under consideration for standardization by the Internet Engineering Task Force (IETF) [37].

Current security proposals for S-BGP rely on the use of the public key infrastructure (PKI) [6, 36], each party holding a public-secret key pair and a digital certificate on the public key issued by a Certification Authority (CA). Public-key cryptography tools such as digital signatures and their variants should be able to ensure proper integrity/authentication and authorization verification.

However, most existing proposals and analyses do not go further than pointing out specific attacks and suggesting possible fixes. For example, a survey of BGP security [21] informally discusses such threats as message tampering, session termination, prefix hijacking, prefix deaggregation, subversion of path information, route flapping, etc. But it is not immediately clear what precisely an adversary is allowed or supposed to do. Can it peek on communication, corrupt nodes, collude, etc.? And what are its goals? Even though the proposed solutions may seem plausible, there is no provable guarantee they actually "work". For example, the proposal for secure path vector routing described in [33] without provable security analysis was later shown to suffer from attacks that could be mounted by 60% of AS's on the Internet in [39]. What is missing is the provable security analysis, which is the superior alternative to the unproductive trial-and-error as it provides security guarantees.

CCS'12, October 16–18, 2012, Raleigh, North Carolina, USA.

It is a must in modern cryptographic research and design, and it is more and more often required by the standards bodies.

The only attempt to use provable security (to the best of our knowledge) in the context of securing BGP has been done in [22] (Appendix A). However, there are no details of the security model[1] (it is not clear who is given what keys), the model is very weak: collusions are not addressed (the adversary can only corrupt one AS), route validity—when a route does not contain edges that do not physically exist in the network and no node's export policy on that route is violated—is not captured, and there are no proofs of security. Providing proper provable security treatment for routing protocols is the main focus of our work.

It may be debatable how possible is widespread, near-future adoption and deployment of S-BGP. The main technical reason is that securing BGP adds time and space complexity overhead. There are also political and economic factors, including the financial cost of secure routing. Finally, there is the problem of gradual deployment; that is the necessity of bypassing the impossibility of an instantaneous global change of configurations. But as a position paper on the subject [16] notices, many objections are inherent to any possible solution and are unavoidable. This should not by any means give specialists reasons to stop working on existing problems to make deployable secure BGP a reality, especially given the growing importance of the Internet and its security.

Moreover, several recent efforts in this field justify an optimistic view on S-BGP's deployment [8, 36, 28]. Resource Public Key Infrastructure (RPKI) [5, 36] is a major, real-life, current effort by the Secure Inter-Domain Routing (SIDR) group of the IETF [4] to protect and verify the association of AS numbers and IP prefixes to their owners via cryptographic certificates. Intuitively, RPKI is like an implementation of S-BGP that addresses only a fraction of the attacks that S-BGP is supposed to address (i.e. prefix-hijacking attacks). Results in [28] suggest that the majority of the Internet would deploy secure routing protocols, such as S-BGP, if AS's were to prefer secure routes (routes where every AS deploys S-BGP) to non-secure routes (because by adopting S-BGP, an AS could attract more traffic and increase its revenue). Thus, from an economic point of view, S-BGP could be gradually deployed starting from a small set of AS's.

The above motivation is also applicable to other path vector routing protocols, e.g. BGPSEC [37] and Secure Origin BGP (SoBGP) [50]. While S-BGP is our main focus, many of our results generalize for S-BGP variants and other path vector protocols.

Our work is the first to study path vector routing protocols in the provable-security framework. A major issue regarding S-BGP's use is the lack of understanding of its provable security guarantees in scenarios of partial deployment [23]. Studying scenarios that relax the PKI and the public-key-crypto-use requirements, while still achieving reasonable (and well-defined) security levels, is our second main contribution.

Our paper continues a line of work providing provable security treatment for practical protocols, such as SSH [14, 43] and Kerberos [9, 18]. We now describe our contributions in more detail.

INTERDOMAIN NETWORK AND PATH VECTOR PROTOCOL DEFINITIONS. We start with defining an interdomain network (a network of AS's such as the Internet) and a path vector routing protocol. Our protocol definition is general enough, and we show how both BGP and S-BGP (and SoBGP in the full version [19]) fit.

SECURITY MODEL. Next we design the security definition. We carefully study numerous known security threats and generalize them in a new formal security notion. The definition is quite strong in that the adversary we consider knows the configuration of the whole network, can observe and modify all communication on the network, can select nodes that will not have public keys (for modeling partial PKI deployment), can corrupt all but two AS's, can learn all secret information of the corrupted AS's, and can act arbitrarily on their behalf. We explain how this model takes into account adaptive corruptions. Our security definition elegantly captures scenarios when not all nodes have public keys (i.e. partial PKI deployment). At the same time, our security definition is compact and rather simple. The adversary is successful if it makes an honest node accept a route announcement that is not legitimate in at least one of the following three ways: (1) unauthentic origin, (2) unauthentic route, (3) invalid route (). If no efficient adversary can succeed with noticeable probability in the above three ways, we say that the protocol guarantees full security or (1) origin authentication, (2) route authentication and (3) route validity, respectively.

In Section 5 we explain how the numerous known vulnerabilities are captured by just these three cases. For example, case (1) captures attacks of advertising prefixes that do not belong to the corresponding origins (that were not certified by the certification authority (CA)), also known as the prefix hijacking attacks [10]. Case (2) captures all attacks that include tampering with any announcement made by an honest AS. This includes as a special case a threat known as violation of connection authentication. Case (3) captures somewhat less known attacks on S-BGP such as export policy violations, sometimes known as "route leaks", [30] and announcing a route that cannot physically exist in the network [46]. Our unified definition allows one to analyze full security of a routing protocol or consider security against each of the aforementioned classes of attacks separately.

S-BGP AND SOBGP SECURITY ANALYSES. We prove (by reduction) that S-BGP does indeed guarantee origin and route authentication if the utilized building blocks such as a certification and signature schemes are secure (we also prove that a secure certification protocol can be constructed from a secure signature scheme). This formally justifies the design of S-BGP as a means to protect against some of the major threats. However, we also show that S-BGP does not guarantee route validity by presenting explicit attacks under our definition. This is not surprising as it has been shown before, albeit without provable security analysis, and several solutions have been proposed [46, 30, 47, 26]. We propose simple fixes to S-BGP that involve the certification authority certifying links and financial relationships between AS's and we prove that the modified protocol guarantees route validity if the underlying certification protocol is secure. This is somewhat similar to AS policy certificates used in SoBGP [50]. Although requiring such certificates may seem inefficient AS's may be unwilling make their connections, business relationships and export policies known, we argue that without link-certificates route validity cannot be guaranteed in general. Furthermore, in light of current efforts of RPKI [5] to, even partially, protect approximately 400K currently existing prefixes [42, 2] with cryptographic certificates, we believe that requiring the extra management of cryptographic link certificates to protect all links, of which there are approximately only 150K [24], to be still reasonable even though they may require more frequent updates.

SoBGP [50] is another well-known effort to secure BGP. Although this has already been discussed within the community, our security model can be used to formally confirm that SoBGP guarantees origin authentication but does not guarantee route authenti-

[1]The details are promised in the technical report, but they do not appear there as well.

cation and route validity (we provide sketches in the full version [19]). Due to these weaknesses we focus only on S-BGP when considering partial PKI deployment.

RELAXING THE PKI REQUIREMENT. Of course, reliance on full PKI deployment and the use of public-key cryptography, while seemingly necessary for strong security, are quite expensive measures. We study the effect on security from having partial PKI deployment, i.e. when not all nodes have certified public keys, and put forth results that can facilitate our understanding of how gradual deployment (and even full deployment, but where, for efficiency reasons, not all parties want to execute parts of the protocol that require the use of their private keys) of secure routing protocols on the Internet could be made possible. Studying security of the partial PKI deployment of path vector protocols is our second main contribution, and the results here are more unexpected and technical.

We first show that S-BGP fails to provide route authenticity if there is at least one node without a certified public key. However, we show that the loss of PKI-related security can be compensated by exploiting physical security of links together with a trust relationship that neighboring nodes must have to establish a physical communication link between them in the first place, and we show that full security is possible if nodes do not select routes with more than one keyless node in a row at any part of those routes. We then show that such restrictions are in fact necessary. Finally, we show that if all prefixes and links are certified by a trusted certification authority, even when no node has a public key, nodes are guaranteed to discover valid routes with authentic origins, and the worst thing that can happen is that an honest node may accept a route to some prefix such that for at least one honest node on that route, the latter does not prefer that its part of that route the most. We then argue that in this setting, due the Internet's lack of any provably secure accountability mechanism, the Internet as a whole is just as protected against adversaries whose primary goal is to divert traffic onto unwanted routes as when PKI is fully deployed. Although requiring link certificates while not requiring full PKI deployment may seem to have limited practical gains, this result is a major leap toward understanding the security guarantees and efficiency tradeoffs that can be achieved even when no node has a public key. This result suggests that in the initial stages of partial deployment of secure path vector protocols, it may be more beneficial to deploy link certificates rather than have some nodes possess public keys while deploying no link certificates. We discuss this further with respect to partial and full deployment of RPKI on the Internet.

2. PRELIMINARIES

NOTATION AND CONVENTIONS. We denote by $\{0,1\}^*$ the set of all binary strings of finite length. If x, y are strings then (x, y) denotes the concatenation of x and y from which x and y are uniquely decodable. If $\kappa \in \mathbb{N}$ then 1^κ denotes the string consisting of κ consecutive "1" bits. If S is a finite set, then $s \xleftarrow{\$} \mathsf{S}$ denotes that s is selected uniformly at random from S . If $\mathcal{A}$ is a randomized algorithm and $n \in \mathbb{N}$, then $a \xleftarrow{\$} \mathcal{A}(i_1, i_2, \ldots, i_n)$ denotes that a is assigned the outcome of the experiment of running $\mathcal{A}$ on inputs $i_1, i_2, \ldots, i_n$. The empty string is denoted by ε. An adversary is an algorithm. By convention, the running-time of an adversary includes that of its overlying experiment. All algorithms are assumed to be randomized and efficient (i.e. polynomial in the size of the input).

PROVABLE SECURITY APPROACH. In this work we apply the provable security approach. Unlike the unproductive and cyclic trial-and-error approach to security, this methodology allows us to have protocols, whose security is provably guaranteed, as long as the assumption about the underlying hard problem remains true for computationally bounded adversaries. This approach consists of the following components. (1) A formal definition of a protocol's syntax. (2) A formal definition of the security task in question that includes a precise description of adversarial capabilities and when is the adversary considered successful. (3) A reduction proof showing that the only way to break the protocol according to the definition is by breaking the underlying problem, believed to be hard. Such treatment requires precise notation and definitions at each of the above steps. Hence, we introduce some notation and definitions that were not common in the networking literature, but are rather standard in the cryptographic literature. We provide informal explanations wherever possible to make the formalisms easier to follow.

We note that our work does not follow the alternative formal-methods (symbolic) approach. Such analysis has not been done to the best of our knowledge as it requires some innovations, such as dealing with lists. When done, it will allow for automatic verification, but still will not imply security in the strongest computational model (and our analysis does) as the required soundness theorems are to rely on the unrealistic properties of signatures.

PKI AND SIGNATURE SCHEMES. Whenever we use public keys, we also (implicitly) assume that a *public key infrastructure (PKI)* is supported, i.e. the public keys are valid, bound to users' identities and are publicly known.

A *digital signature scheme* $\mathcal{SS} = (\mathsf{Kg}, \mathsf{Sign}, \mathsf{Ver})$ with associated *message space* MsgSp is defined by three algorithms. The randomized *key generation* algorithm Kg takes the security parameter 1^k and outputs a public–secret key pair: $(pk, sk) \xleftarrow{\$} \mathsf{Kg}(1^k)$. The (possibly randomized) *signing* algorithm Sign takes the secret key and message $M \in \mathsf{MsgSp}$ and outputs a signature: $\sigma \xleftarrow{\$} \mathsf{Sign}(sk, M)$. The deterministic *verification* algorithm Ver takes the public key, a message and a signature and outputs a bit $b \in \{0,1\}$ indicating whether the signature is deemed valid or not: $b \leftarrow \mathsf{Ver}(pk, M, \sigma)$.

For correctness, it is required that for every (pk, sk) output by $\mathsf{Kg}(1^k)$ and every $M \in \mathsf{MsgSp}$ we have that $\mathsf{Ver}(pk, M, \mathsf{Sign}(sk, M)) = 1$.

The traditional security notion for a scheme $\mathcal{SS} = (\mathsf{Kg}, \mathsf{Sign}, \mathsf{Ver})$ considers an experiment $\mathbf{Exp}^{\text{uf-cma}}_{\mathcal{SS}}(A)$ associated with an adversary A. First, a pair of keys is generated: $(pk, sk) \xleftarrow{\$} \mathsf{Kg}(1^k)$. Then A is given pk and the signing oracle, and it has to output a message and a forgery: $(M, \sigma) \xleftarrow{\$} A^{\mathsf{Sign}(sk,\cdot)}(pk)$. The adversary wins and the experiment returns 1 iff $\mathsf{Ver}(pk, M, \sigma) = 1$, $M \in \mathsf{MsgSp}$ and A never queried M to $\mathsf{Sign}(sk, \cdot)$. We say that $\mathcal{SS}$ is *uf-cma-secure* if $\Pr\left[\mathbf{Exp}^{\text{uf-cma}}_{\mathcal{SS}}(A) = 1\right]$ is negligible in k for all efficient algorithms A.

CERTIFICATION SCHEMES. To the best of our knowledge, the certification scheme primitive has not been explicitly defined, but it has been considered as parts of other protocols, e.g. certified encryption and digital signature schemes in [17]. (In the application we consider we will involve the certification protocols to certify prefix ownership for the origins (known as address attestation) as well neighbor relations of AS's and, if they are, the type of business relationship they have with each other.)

A two-party *certification protocol* $\mathcal{CP} = (\mathsf{Kg}_{\mathsf{CA}}, (\mathsf{CA}, \mathsf{U}),$

Vercert) is defined by a key generation algorithm, a pair of (possibly) interactive randomized algorithms executed between the certification authority and a user (in our case, an AS), and a verification algorithm. The protocol is associated with an ID space IDSp and data space DSp. $\mathsf{Kg}_{\mathsf{CA}}$ takes the security parameter 1^k and outputs a public-secret key pair $(pk_{\mathsf{CA}}, sk_{\mathsf{CA}})$ for the CA.

CA takes as input a secret key sk_{CA}, the identity of user $ID \in \mathsf{IDSp}$ and data $D \in \mathsf{DSp}$. A node's ID is the unique AS number given to the AS associated with that node by the Internet Assigned Numbers Authority (IANA) [3], as is done for every AS on the Internet. U takes as input the public key pk_{CA}, the identity $ID \in \mathsf{IDSp}$ and data $D \in \mathsf{DSp}$. As result of the interaction, the outputs of both parties are $\perp$, if something went wrong, or (ID, D, cert), where cert is an issued certificate. We write $((ID,\ D, \mathsf{cert}), (ID, D, \mathsf{cert})) \stackrel{\$}{\leftarrow} (\mathsf{CA}(sk_{\mathsf{CA}}, ID, D),\ \mathsf{U}(pk_{\mathsf{CA}},\ ID,\ D))$ for the result of an honest interaction. Vercert takes as input $(pk_{\mathsf{CA}}, ID, D, \mathsf{cert})$ and outputs a bit.

The correctness requirement states that for any pair $(pk_{\mathsf{CA}}, sk_{\mathsf{CA}})$ output by $\mathsf{Kg}_{\mathsf{CA}}(1^k)$, any $ID \in \mathsf{IDSp}$ and $D \in \mathsf{DSp}$, the result of certification $((ID, D, \mathsf{cert}), (ID, D, \mathsf{cert})) \stackrel{\$}{\leftarrow} (\mathsf{CA}(sk_{\mathsf{CA}}, ID, D), \mathsf{U}(pk_{\mathsf{CA}}, ID, D))$ passes verification, i.e. Vercert $(pk_{\mathsf{CA}}, ID, D, \mathsf{cert}) = 1$.

We now define the security of the certification protocol $\mathcal{CP} = (\mathsf{Kg}_{\mathsf{CA}}, (\mathsf{CA}, \mathsf{U}), \mathsf{Vercert})$ with IDSp, DSp. We call the notion *unforgeability under chosen-data attack*. Consider the following experiment $\mathbf{Exp}^{\text{uf-cda}}_{\mathcal{CP}}(A)$ associated with an adversary A.

First, the CA's keys are generated: $(pk_{\mathsf{CA}}, sk_{\mathsf{CA}}) \stackrel{\$}{\leftarrow} \mathsf{Kg}_{\mathsf{CA}}(1^k)$. A gets pk_{CA} and after that can repeatedly output (ID, D) so that $ID \in \mathsf{IDSp}, D \in \mathsf{DSp}$ and for each pair participate in $(\mathsf{CA}(sk_{\mathsf{CA}},\ ID, D),\ A(pk_{\mathsf{CA}}, ID, D))$ on behalf of the user interacting with the CA.

The experiment outputs 1 iff A at some point returns $(ID', D', \mathsf{cert}')$ so that $ID' \in \mathsf{IDSp}, D' \in \mathsf{DSp}$, $\mathsf{Vercert}(pk_{\mathsf{CA}}, ID', D', \mathsf{cert}') = 1$ and CA never output $(ID', D', \mathsf{cert}'')$, for any cert$''$.

We define A's advantage $\mathbf{Adv}^{\text{uf-cda}}_{\mathcal{CP}}(A)$ in this experiment to be $\Pr\left[\mathbf{Exp}^{\text{uf-cda}}_{\mathcal{CP}}(A) = 1\right]$. We say that $\mathcal{CP}$ is *uf-cda-secure* if $\mathbf{Adv}^{\text{uf-cda}}_{\mathcal{CP}}(A)$ is negligible in k for all efficient algorithms A. Note that one could define a stronger security notion, but that would be an overkill for the purposes of our application.

CONSTRUCTION 2.1. Let $\mathcal{SS} = (\mathsf{Kg}, \mathsf{Sign}, \mathsf{Ver})$ be a signature scheme with MsgSp. We define the corresponding $\mathcal{CP}_s = (\mathsf{Kg}, (\mathsf{CA}, \mathsf{U}), \mathsf{Vercert})$ with IDSp, DSp so that for every $ID \in \mathsf{IDSp}$ and $D \in \mathsf{DSp}$, $(ID, D) \in \mathsf{MsgSp}$. $(\mathsf{CA}, \mathsf{U})$ is then as follows. The CA sends $\mathsf{cert} = \mathsf{Sign}(sk_{\mathsf{CA}}, (ID, D))$ to the user. The user verifies $\mathsf{Ver}(pk_{\mathsf{CA}}, (ID, D), \mathsf{cert})$ and, if correct, both output cert: (ID, D, cert), otherwise they both output $\perp$. $\mathsf{Vercert}(pk_{\mathsf{CA}}, ID, D, \mathsf{cert})$ returns $\mathsf{Ver}(pk_{\mathsf{CA}}, (ID, D), \mathsf{cert})$.

THEOREM 2.2. *Let $\mathcal{SS} = (\mathsf{Kg}, \mathsf{Sign}, \mathsf{Ver})$ be a signature scheme with message space* MsgSp *and let $\mathcal{CP}_s = (\mathsf{Kg}, (\mathsf{CA}, \mathsf{U}), \mathsf{Vercert})$ be its corresponding certification scheme with identity and data spaces* IDSp, DSp *as per Construction 2.1. Then, $\mathcal{CP}_s$ is uf-cda-secure if $\mathcal{SS}$ uf-cma-secure.*

The proof is simple and is presented in the full version [19].

3. INTERDOMAIN NETWORK ROUTING

We define syntaxes for interdomain networks and path vector protocols which we use to model the Internet, BGP and S-BGP.

AN INTERDOMAIN NETWORK. We model an *interdomain network* as a tuple $\mathcal{I} = (\mathbf{G} = (\mathsf{AS's}, \mathsf{link}), \mathsf{Prefixes}, \mathsf{OrforPr}, \mathsf{relation}, \mathsf{preferto}, \mathsf{policy})$ where $\mathbf{G}$ is a finite, connected graph consisting of a set of nodes, AS's, representing autonomous systems and a set of edges defined by a function $\mathsf{link}\colon \mathsf{AS's} \times \mathsf{AS's} \to \{0,1\}$ returning 1 iff the nodes are *neighbors*, Prefixes is a set of strings in $\{0,1\}^*$ representing *prefixes*, which specify sets of IP addresses, The origin-for-prefix function $\mathsf{OrforPr}\colon \mathsf{Prefixes} \to \mathsf{AS's}$ takes a prefix and returns a node designated to own that prefix (called *origin*), and $\mathsf{relation}\colon \mathsf{AS's} \times \mathsf{AS's} \to \mathsf{BR}$ is a function that takes two nodes and returns their business relationship if they are neighbors and $\perp$ otherwise [2]. Here BR defines the set of all possible pairwise business relationships in $\mathcal{I}$ between neighbors. For example, the neighbors could be *peers* or one can be a *provider* and the other its *customer* [27]. Before defining the last components of $\mathcal{I}$, we introduce comments and auxiliary definitions.

Note that $\mathcal{I}$ implicitly defines the set of origins $\mathsf{Origins} \subseteq \mathsf{AS's}$ as the image set of function OrforPr. We denote the set of neighbors of a node N as $\mathsf{Neighbors}(N)$.

A *route* in $\mathcal{I}$ is a sequence of nodes $(N_n, N_{n-1}, \ldots, N_2, N_1)$, for some $n \in \mathbb{N}$ and $N_i \in \mathsf{AS's}$ for all $1 \le i \le n$, such that $N_1 \in \mathsf{Origins}$. Here N_1 is the destination of traffic and N_i is a possible source of traffic for every $2 \le i \le n$. Unless otherwise specified, nodes on routes will be indexed in increasing order right-to-left, starting with the origin, which is convenient for path vector protocols. We say that N_i is up- or down-stream from node N_j on a particular route, if $i < j$ or $i > j$ respectively. A *subroute* of some route $R = (N_n, \ldots N_2, N_1)$ is a sequence of nodes $(N_i, \ldots N_1)$, for any $1 \le i \le n$, that is defined as the i right-most entries of R. A route is said to be *feasible* if for every pair of consecutive nodes (N_{i+1}, N_i) in that route, $\mathsf{link}(N_{i+1}, N_i) = 1$ for $n < i \le 1$, i.e. the nodes are neighbors. A route $(N_n, \ldots N_2, N_1)$ is said to be *to* some prefix $P \in \mathsf{Prefixes}$ if $\mathsf{OrforPr}(P) = N_1$.

preferto specifies total and transitive binary relations $\mathsf{preferto}_N$ on routes to the same prefix in Prefixes for each node $N \in \mathsf{AS's}$. policy specifies functions policy_N that define export policy rules for each node $N \in \mathsf{AS's}$; policy_N takes a route to some prefix P together with the output of relation on N and the first node on that route (the second parameter is ignored if N owns P) and outputs a set of nodes to which that node is allowed to export (i.e. advertise) that route. With this syntax we consider only next-hop export policy functions whose outputs depend on the routes and business relationships of neighbors of the node exporting the route, since they quite reasonably approximate the export policy rules that AS's on the Internet use to advertise their routes to different neighbors [27, 30, 28]. We comment on how our results could be extended for more complicated export policy functions in Section 7.

We say that $N_i \in \mathsf{AS's}$ *prefers* some route R to some other route R', both to the same prefix P, if $R\ \mathsf{preferto}_{N_i}\ R'$, and we say that a route $R = (N_{n-1}, \ldots, N_2, N_1)$ to prefix $P \in \mathsf{Prefixes}$ is node N_n's i^{th} *most preferred* route to P, for some $i \ge 1$, if there are exactly $i-1$ distinct routes $R' = (M_\ell, \ldots, M_1, N_1)$ to P such that $R'\ \mathsf{preferto}_{N_n}\ R$. We say that R is N_n's most preferred route to P if $i = 1$. For any node N_n, for any route $R = (N_{n-1}, \ldots, N_2, N_1)$, $R\ \mathsf{preferto}_{N_n}\ \varepsilon$ if and only if $\mathsf{OrforPr}(P) = N_1$, and ε is N_1's most preferred route to P if $\mathsf{OrforPr}(P) = N_1$.

A route $R = (N_n, \ldots N_2, N_1)$ is *valid* if it is feasible and consistent with policy of every node on that route, i.e. $N_i \in \mathsf{policy}_{N_{i-1}}((N_{i-1}, \ldots N_2, N_1), \mathsf{relation}(N_{i-1}, N_{i-2}))$ for all $2 \le i \le n$.

A ROUTING PATH VECTOR PROTOCOL. Let $\mathcal{I} = (\mathbf{G} = (\mathsf{AS's}, \mathsf{link}), \mathsf{Prefixes}, \mathsf{OrforPr}, \mathsf{relation}, \mathsf{preferto}, \mathsf{policy})$ be an interdo-

[2] link may be redundant given relation, but we keep the former to maintain a general graph definition

main network. An interactive and stateful *path vector protocol* $\mathcal{PV} = (\mathsf{Init}, \mathsf{An})$ is defined by two algorithms.

• Init is an optional randomized algorithm run by a node (or a CA) that takes the security parameter 1^k and generates the corresponding public and secret keys for the node (or the CA).

• An is a stateful and possibly randomized, interactive multiparty algorithm run between the nodes and possibly the CA. Each node $N \in \mathsf{AS's}$ is given inputs $(N, \mathsf{Neighbors}(N), \mathsf{relation}_N, \mathsf{preferto}_N, \mathsf{policy}, \mathbf{P}_N, pk_{\mathsf{CA}}, \mathbf{pk})$, where $\mathsf{relation}_N$ outputs $\mathsf{relation}(N, N')$ for all $N' \in \mathsf{Neighbors}(N)$ and $\perp$ otherwise. $\mathbf{P}_N \subseteq \mathsf{Prefixes}$ is the set of prefixes N owns, pk_{CA} is the optional public key of the CA and $\mathbf{pk}$ denotes the optional set of public keys of all nodes in $\mathsf{AS's}$. The optional CA takes as inputs $(\mathcal{I}, pk_{\mathsf{CA}})$. During the execution, N_i sends messages known as *announcements* to $N_j \in \mathsf{Neighbors}(N_i)$, in accordance with policy_{N_i}, of the form (N_i, N_j, R, P, W, Aux), where R is a route to $P \in \mathsf{Prefixes}$ known as the *path attribute*, $W \in \{0, 1\}$ is the withdrawal flag, and $Aux \in \{0, 1\}^*$ holds any additional information. Upon receipt of a message, N_j can *reject* that message by outputting $\perp$. N_j *accepts* a message if N_j does not reject it.

Note that although export policy function of each node is given as input to each node, nodes cannot find out other node's decisions with respect to exporting arbitrary routes, because they do not know business relationships of remote nodes. We comment on how our results could be extended for scenarios when other nodes' policies are not publicly known in Section 7.

We say that $\mathcal{PV}$ is *correct* for a class of networks if when every node in $\mathsf{AS's}$ follows $\mathcal{PV}$, every announcement during its execution is accepted for every network $\mathcal{I} \in \mathcal{C}$. One could consider a stricter notion of correctness that would require path vector protocols to be useful and allow nodes to learn routes to various destinations, e.g. in practice path vector protocols such as BGP are considered useful for the Internet only if they converge—reach a point after which no node receives an announcement with a route that is preferred to a route that it has previously selected as its most preferred. However, such requirement may be unnecessarily complicated and is outside of the scope of this paper. Also, as explained in Section 5, essential vulnerabilities of path vector protocols preventable with crypto stem from honest nodes accepting bogus announcements and, therefore, as with all crypto protocols, our security definitions are adequate even when a protocol may not be useful by itself or due to adversaries.

4. BGP AND S-BGP

In this section we first describe BGP, and then show how S-BGP extends it to incorporate security features. Although in our model we do not require communication to be either concurrent or asynchronous, for the rest of the paper we assume only asynchronous communication as it captures delays and re-ordering ubiquitous in real life scenarios.

BORDER GATEWAY PROTOCOL. We present the essential aspects of the the Border Gateway Protocol (BGP) that is used to establish routes on the Internet of today [27, 30, 28]. Let $\mathcal{I} = (\mathbf{G} = (\mathsf{AS's}, \mathsf{link}), \mathsf{Prefixes}, \mathsf{OrforPr}, \mathsf{relation}, \mathsf{preferto}, \mathsf{policy})$ be an interdomain network. BGP uses no PKI and no CA, so the optional algorithm Init is never invoked. The An algorithm is as follows.

Every node $N \in \mathsf{AS's}$ maintains state in the form of a table T_N, called the *routing table*, which is initially empty. Each field $T_N[P]$ indexed by a prefix $P \in \mathsf{Prefixes}$, for which $\mathsf{OrforPr}(P) \neq N$, is a list consisting of routes to P that N has received as announcements from neighbors. Each route in $T_N[P]$ is ranked such that $T_N[P][i]$ contains N's i^{th} most preferred route to P.

If the node's input $\mathbf{P}_N$ is nonempty (i.e. $N \in \mathsf{Origins}$), then for every prefix $P \in \mathbf{P}_N$, N sends an announcement $(N, N', (N), P, 0, \varepsilon)$, advertising access to P, to every neighbor $N' \in \mathsf{policy}_N((N), \varepsilon)$.

During BGP's execution, when a node receives an announcement advertising a new route to some prefix, that announcement is ignored if that node is contained in the announced route or if the new route is already contained in that node's routing table to that prefix. Otherwise, that node determines the new route's rank in its routing table to the same prefix, records that route and its rank, and, if necessary, updates the ranks of the other routes to that prefix. If the announced route becomes the most preferred route to that prefix, that node propagates that route to its neighbors in accordance with its export policy rules. If a node receives an announcement that is a notification of a withdrawal of a route (i.e. that route should not to be used by the receiving node) stored in its routing table, then that node deletes that entry from its table and propagates that route's withdrawal to its neighbors in accordance with its export policy rules. We describe BGP more concretely below.

For every route announcement $(N', N, R, P, W, \varepsilon)$ that N receives from neighbor N', if R and $T_N[P]$ do not contain N and R respectively, N sends a route announcement to every neighbor as per policy_N and updates $T_N[P]$ according to rules (1)-(3) below.

(1) If the announcement presents the most preferred route to P, i.e. $W = 0$ and R $\mathsf{preferto}_N$ $T_N[P][1]$, then N:

   (a) sends a route withdrawal announcement $(N, N', (N, T_N[P][1]), P, 1, \varepsilon)$ to every neighbor as per policy_N,[3]

   (b) sends a route advertisement $(N, N', (N, R), P, 0, \varepsilon)$ to every neighbor as per policy_N,

   (c) increments by one the rank of every route in $T_N[P]$ and makes an update $T_N[P][1] \leftarrow R$.

(2) If the announcement presents a route to P that is not the most preferred, i.e. $W = 0$ and $T_N[P][1]$ $\mathsf{preferto}_N$ R, then N determines rank i such that R is the i^{th} most preferred route out of all routes in $T_N[P]$, increments by one the rank of every route in $T_N[P]$ that is less preferred than R, and makes an update $T_N[P][i] \leftarrow R$.

(3) If the announcement is a withdrawal of a route that N has stored, i.e. $W = 1$ and $R \in T_N[P]$, then N:

   (a) if $R = T_N[P][1]$, sends a withdrawal announcement $(N, N', (N, R), P, 1, \varepsilon)$ to every neighbor as per policy_N,

   (b) if $R = T_N[P][1]$ and $T_N[P][2] \neq \varepsilon$, sends a route advertisement $(N, N', (N, T_N[P][2]), P, 0, \varepsilon)$ to every neighbor as per policy_N,

   (c) removes R from $T_N[P]$ and decrements the rank of every route in $T_N[P]$ ranked higher than R.

N ignores new announcements in all other cases. In the absence of adversaries and errors, no message in BGP should be rejected, so BGP should be correct for various interesting classes of networks described in [27, 30, 28] that are believed to closely capture how routing is done on the Internet.

SECURE BORDER GATEWAY PROTOCOL. The Secure Border Gateway Protocol (S-BGP) [34] is an extension to BGP that relies on the full deployment of PKI (each AS should know authentic and valid public keys of other AS's). In S-BGP, public-key cryptography is used to bind prefixes to their origins with certificates, called

[3]Although in practice withdrawals in this specific scenario may be implicit, we make them explicit here for clarity.

address attestations, issued by a third trusted party as well as to generate *route attestations*—certificates generated by intermediate nodes on a route announcements they propagate. Route announcements recipients verify the origin of the prefix in that announcement and the certificates of the nodes on the route that announcement has traversed. We present the essential operations of S-BGP more concretely below.

CONSTRUCTION 4.1. Let $\mathcal{I} = (\mathbf{G} = (\mathsf{AS's}, \mathsf{link}), \mathsf{Prefixes}, \mathsf{OrforPr}, \mathsf{relation}, \mathsf{preferto}, \mathsf{policy})$ be an interdomain network, let $\mathcal{SS} = (\mathsf{Kg}, \mathsf{Sign}, \mathsf{Ver})$ be a signature scheme with $\mathsf{MsgSp} = \{0,1\}^*$, and let $\mathcal{CP}_s = (\mathsf{Kg}_{\mathsf{CA}}, (\mathsf{CA}, \mathsf{U}), \mathsf{Vercert})$ be the corresponding certification protocol as per Construction 2.1. In S-BGP $= (\mathsf{Init}, \mathsf{An})$, as part of Init the CA runs $\mathsf{Kg}_{\mathsf{CA}}(1^k)$ to generate $(pk_{\mathsf{CA}}, sk_{\mathsf{CA}})$ and each AS runs $\mathsf{Kg}(1^k)$ to generate (pk, sk). An is defined as follows.

If node N_j's input $\mathbf{P}_{N_j}$ is nonempty (i.e. $N_j \in \mathsf{Origins}$), then for every prefix $P \in \mathbf{P}_{N_j}$, N_j does the following:

• CA and N_j interact according to $(\mathsf{CA}, \mathsf{U})$, N_j being U. The input to U is $(pk_{\mathsf{CA}}, N_j, P)$, the input to CA is $(sk_{\mathsf{CA}}, N_j, P)$ and the outputs of both parties are (N_j, P, cert). Address attestation $AA^P_{N_j} \equiv \mathsf{cert}$ is N_j's certificate of ownership of P.

• Next, for every $N_i \in \mathsf{policy}(N_j, \epsilon)$, N_j runs $\mathsf{Sign}(sk_{N_j}, (N_i, N_j, P))$ to produce a route attestation, $RA^i_{R_j}$, and sends $(N_j, N_i, R = (N_j), P, 0, Aux = (RA^i_{R_j}, AA^P_{N_j}))$ to N_i; here R^i_j is R's subroute authorized by N_j for N_i to use and announce.

For every new route announcement $(N_{j-1}, N_j, R = (N_{j-1}, \ldots, N_1), P, W, Aux = (RA^j_{R_{j-1}}, \ldots, RA^2_{R_1}, AA^P_{N_1}))$ that N_j receives from N_{j-1}, N_j first performs address attestation and route attestation verification steps as follows. N_j runs $\mathsf{Vercert}\ (pk_{\mathsf{CA}}, N_1, P, AA^P_{N_1})$ and outputs $\perp$ if the output of this computation is 0. Otherwise, N_j runs $\mathsf{Ver}(pk_{N_i}, (N_{i+1}, \ldots N_1, P), RA^{i+1}_{R_i})$ for every $1 \le i \le j-1$ and outputs $\perp$ if at least one such computation outputs 0. If none of the verification steps above results in $\perp$, then N_j performs the same operations as N_j would do in BGP upon receipt of $(N_{j-1}, N_j, R, P, W, \varepsilon)$, as per rules (1)-(3) specified in Section 4. Then, for every message $(N_j, N_{j+1}, R', P, W', \varepsilon)$ that N_j would send to N_{j+1} in BGP, N_j now runs $\mathsf{Sign}(sk_{N_j}, (N_{j+1}, R', P))$ to get $RA^{j+1}_{R'_j}$ and sends $(N_j, N_{j+1}, R', P, W', Aux')$ to N_{j+1} instead, where $R' = (N_j, R)$ and $Aux' = (RA^{j+1}_{R'_j}, Aux)$.

If the underlying signature scheme $\mathcal{SS}$ is correct, the execution of S-BGP is the same as that of BGP in terms of how nodes update their routing tables and how they decide which routes to announce to their neighbors. Therefore, S-BGP is correct for the same classes of networks as BGP if the underlying signature scheme $\mathcal{SS}$ used to generate address and route attestations is correct.

5. ROUTING PROTOCOL SECURITY

In this section we provide a security definition for path vector protocols, show how it captures their security vulnerabilities, and discuss the attacks not captured in our model because they cannot be solved with cryptography.

INTUITION FOR THE FORMAL SECURITY MODEL. In our model, we do not consider malicious CA's, but we do consider malicious AS's. We consider an adversary which is given the CA's public key and the description of the network $\mathcal{I}$ with at least two nodes. The adversary also specifies the number of nodes that will not have public keys and the partition of the nodes into the subsets of corrupted and honest nodes, so that the size of the latter is at least two. In practice, it is unlikely that a malicious party knows the complete configuration of the network including the relations, and can corrupt almost all AS's, but in the definition we target a very strong adversary. We allow the adversary to corrupt multiple nodes to capture collusion. On the Internet, collusion is certainly a plausible scenario, given that multiple AS's could be managed by a single administration with presence in different geographic locations. The adversary is given all the public and secret keys of the corrupted nodes. We assume that the adversary is stateful, i.e. it can preserve state in between stages. All nodes and the CA can interact: the honest nodes and the CA follow the protocol, while the adversary can act arbitrarily on behalf of the corrupted nodes. It can observe and modify all communication.

We note that even though it may not be obvious, our model does take into account *adaptive* corruptions and does not assume *synchronous* communication model. To model attackers who adaptively corrupts AS's, our adversary, who can corrupt all but two nodes, can just behave legitimately on behalf of AS's not yet corrupted in the original setting until the time they are corrupted. Similarly, the adversary can delay, drop and re-order communication.

The adversary wins if it sends a route announcement to an honest node, the node accepts it and either (1) the prefix in the announcement does not belong to the corresponding origin, (2) there is a honest node on the route that never sent the corresponding announcement for the same prefix, and (3) the route is invalid. The latter includes the possibilities of a non-existing (not-connected) route and a route that does not satisfy the export policies of at least one node on that route.

PATH VECTOR PROTOCOL SECURITY DEFINITION. Let $k \in \mathbb{N}$ be the security parameter, $\mathcal{I} = (\mathbf{G} = (\mathsf{AS's}, \mathsf{link}), \mathsf{Prefixes}, \mathsf{OrforPr}, \mathsf{relation}, \mathsf{preferto}, \mathsf{policy})$ be an interdomain network, of size polynomial in k, such that $|\mathsf{AS's}| \ge 2$, and let $\mathcal{PV} = (\mathsf{Init}, \mathsf{An})$ be a path vector protocol that is correct for $\mathcal{I}$. We define the experiment $\mathbf{Exp}^{\text{sec-rout-m}}_{\mathcal{I},\mathcal{PV}}(A)$, for $0 \le m \le |\mathsf{AS's}|$, involving a stateful adversary A as follows.

Given the description of $\mathcal{I}$, A selects the set $\mathsf{nopubk} \subsetneq \mathsf{AS's}$ of nodes that will not have public keys, such that $|\mathsf{nopubk}| = m$. Then, the public–secret key pairs for the CA and all nodes in $\mathsf{AS's} \setminus \mathsf{nopubk}$ are generated via $\mathsf{Init}(1^k)$. Here and further in the paper $\mathbf{pk}$ denotes the vector of public keys of nodes in $\mathsf{AS's} \setminus \mathsf{nopubk}$ and $\mathbf{pk}[i]$ denotes its i'th component. Given all public keys, A outputs the sets of corrupted and (at least two) honest nodes which form a partition of $\mathbf{G}$:

$(\mathsf{Honest}, \mathsf{Corrupted}) \xleftarrow{\$} A(\mathcal{I}, pk_{\mathsf{CA}}, \mathbf{pk})$, so that $|\mathsf{Honest}| \ge 2$, $\mathsf{Honest} \cup \mathsf{Corrupted} = \mathsf{AS's}$ and $\mathsf{Honest} \cap \mathsf{Corrupted} = \emptyset$.

Next A is given all the secret keys of the corrupted nodes $\{\mathbf{sk}[i]: \mathbf{sk}[\mathrm{i}]\ \text{belongs to a corrupted node}\}$, and it starts the execution of An on behalf of all nodes in $\mathsf{Corrupted}$ with the CA and also with the nodes in Honest. The CA and the honest nodes follow the protocol legitimately, while the adversary can act arbitrarily. In particular, A is allowed to intercept and modify announcements exchanged between neighboring honest nodes as well as send messages on behalf of any honest node. The adversary is given transcripts of all communication (as it happens).

The goal of the adversary is to have an honest node, say $N_\ell \in \mathsf{Honest}$, accept an announcement of the form $(N_{\ell-1}, N_\ell, R = (N_{\ell-1}, \ldots, N_1), P, W, Aux)$, so that at least one of the following conditions is true (the indexing of the nodes on the route is not essential for the definition and is done for simplicity only).

1. *Unauthentic origin:* $\mathsf{OrforPr}(P) \ne N_1$. In this case the experiment outputs 1.
2. *Unauthentic route:* there exists $1 \le i \le \ell - 1$ so that $N_i \in \mathsf{Honest}$ and N_i never sent announcement $(N_i, N_{i+1}, R' =$

$(N_i, \ldots, N_1), P, W', Aux')$ for any W', Aux' to N_{i+1}. In this case the experiment outputs 2.

3. *Invalid route:* R is invalid. In this case the experiment outputs 3.

$\mathbf{Exp}^{\text{sec-rout-m}}_{\mathcal{I},\mathcal{PV}}(A)$ returns an output as soon as A wins; if more than one condition above holds, $\mathbf{Exp}^{\text{sec-rout-m}}_{\mathcal{I},\mathcal{PV}}(A)$ outputs the smallest number. We define A's advantage $\mathbf{Adv}^{\text{sec-rout-m-b}}_{\mathcal{I},\mathcal{PV}}(A)$ in this experiment as $\Pr\left[\mathbf{Exp}^{\text{sec-rout-m}}_{\mathcal{I},\mathcal{PV}}(A) = b\right]$, for $b \in \{1,2,3\}$.

We define $\mathcal{C}_m^{\mathcal{PV}}$ to be the class of all networks which have m nodes without public keys and for which a path vector protocol $\mathcal{PV}$ is correct, for $m \leq |\mathsf{AS's}|$. $\mathcal{PV}$ guarantees *origin authentication*, *route authentication*, and *route validity* with m-partial deployment (m-PD) for a class of networks $\mathcal{C}_m^{\mathcal{PV}}$, if for every $\mathcal{I} \in \mathcal{C}_m^{\mathcal{PV}}$, for every efficient adversary A, the probability that $\mathbf{Exp}^{\text{sec-rout-m}}_{\mathcal{I},\mathcal{PV}}(A)$ returns 1, 2 and 3 respectively is negligible in k. $\mathcal{PV}$ is *fully secure* with m-PD for a class of networks $\mathcal{C}_m^{\mathcal{PV}}$, if it guarantees origin authentication, route authentication and route validity with m-PD for $\mathcal{C}_m^{\mathcal{PV}}$, i.e. for every $\mathcal{I} \in \mathcal{C}_m^{\mathcal{PV}}$, for every efficient adversary A the probability of $\mathbf{Exp}^{\text{sec-rout-m}}_{\mathcal{I},\mathcal{PV}}(A)$ returning 1, 2 or 3 is negligible in k. When $m = 0$, we omit the suffix 0-PD when qualifying security of protocols.

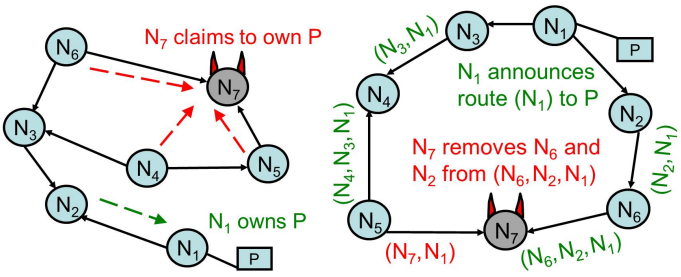
Our model does not consider rogue keys and replay attacks. This is very common as it is known that the standard measures like proofs of possession of secret keys during the key registration [6, 45] and the use of timestamps can be used to provide the additional protection. To address rogue key attacks, we could require the adversary to output the public and secret keys of corrupted users in order model the situation where users are required to perform proofs of knowledge of secret keys during key registration. However, all of our results would still trivially hold in this setting, so we do not complicate our model with this extension since rogue-key attacks are not essential to routing protocols and do not enhance the insights we get about the essential, routing-related attacks on BGP. It may be relevant to investigate whether simpler proofs of possession [45, 17] will suffice, but this is beyond the scope of this paper. We discuss relevance of rogue key attacks with respect to RPKI in Section 8.

We also note that our security notion does not guarantee that the data that nodes send to those prefixes travels along the routes that they have learned and selected, or whether it reaches those prefixes at all. As shown in [29], path vector protocols cannot guarantee that. These are not goals of path vector protocol, but of data-plane accountability and verification which is outside of the scope of this paper and is not captured in our model.

Although our security model does not take into account all complexities of routing protocols, in Sections 6 -8 we show that even a simplified model can point out what is necessary, not just sufficient, to achieve security with respect to essential, fundamental vulnerabilities in path vector protocols in full and partial PKI deployment scenarios .

KNOWN CAPTURED ATTACKS. We discuss how our compact model captures many known vulnerabilities of path vector protocols. For all figures in this section, a directed edge from N to N' indicates that N is N's customer, i.e. N pays N' for all traffic exchanged on their link.

The *Unauthentic origin* condition captures the prefix hijacking attack on BGP, where a corrupt AS claims to own a prefix or announces a more specific prefix, say P, that is owned by another AS. As a result, the corrupt AS could attract potentially all traffic destined to P. With such an attack, a malicious AS could deny access to a particular website, e.g. Pakistan Telecom hijacking YouTube's



(a) Origin Authentication Attack (b) Route Authentication Attack

Figure 1: In (a) N_7 claims to own prefix P and becomes a black hole by attracting majority of traffic destined to P and dropping it. In (b) N_7 attracts N_5's traffic by advertising a fake short route and then forwarding along a longer route via N_6.

prefix in February 2008 [20], e.g. by creating a *black hole*—a locale where all traffic destined to P disappears. In addition, the attacker could intercept sensitive, government-related traffic to analyze it for malicious reasons, as speculated by some with regards to China Telecom diverting approximately 15% of Internet's traffic in April, 2010 for about 20 minutes [25]. Prefix deaggration attacks, in which an attacker deaggregates a prefix into more specific prefixes to attract traffic, are also captured by the unauthentic route condition. This works because routers on the Internet select more specific prefixes over less specific ones by default. RPKI [5] is a major, current effort by ARIN [1] to address origin authentication attacks, but by itself RPKI is not intended to address any other types of attacks. Figure 1(a) presents an example of such an attack, where AS N_7 announces to its neighbors ownership of prefix P, whose actual owner is N_1. As a result, N_7 is able to attract traffic from N_4, N_5, and N_6, because N_7 is closer to them than N_1. This traffic never reaches N_1 because, other than through nodes N_5 and N_6, N_7 does not have an alternative route to N_1.

The *Unauthentic route* condition captures known attacks on BGP where an adversarial AS modifies the path attribute of a route announcement by adding and/or taking AS's out of this attribute as well as pretending to be a different AS altogether. By taking AS's out of the path attribute, the attacker could attract more traffic as the advertised route would seem shorter (and thus more attractive). Adding AS's to a route may make a route less attractive if it makes it seem longer, or contains the receiver of the announcement (which would present a loop and cause the receiver to drop the announcement); this is how an attacker could force an AS not to select certain routes. Figure 1(b) presents an example of such an attack, where AS N_7 removes N_6 and N_2 from the shortest route that N_7 has to P, which is owned by N_1. This makes N_5 believe that N_7 is providing a shorter route to N_1 than the one through N_4, and hence N_5 picks the route through N_7. Thus, N_5 selects a suboptimal route to P, since the route to P through N_7 is actually longer than that through N_4. The attacker benefits not only from intercepting N_5's traffic but also from receiving N_5's payment, since N_5 is N_7's customer.

In the full version [19] we discuss route feasibility [46] and export violation [30, 47, 26] attacks on S-BGP captured by our model.

ATTACKS CRYPTO CANNOT PREVENT. Here we discuss several attacks not captured by our security model for the reason that such attacks cannot be prevented using cryptography.

Path vector protocol divergence cannot be prevented with cryptographic tools since the adversary could keep on withdrawing and then re-announcing the same set of routes ad infinitum. However,

since the number of total routes to every prefix is finite, when a protocol diverges, some paths must be periodically withdrawn and then re-announced again (this is called route flapping), so protocol divergence can be mitigated with tools that prevent route-flapping, e.g. route dampening [21]. Convergence of path vector protocols to suboptimal routes, i.e. paths that are not the most preferred, also cannot be prevented with cryptographic tools since the adversary could just make sure that some nodes never receive announcements of the most preferred routes.

Bellovin and Gansner have studied link cutting attacks which involve physically (e.g. with a DDoS attack) taking out edges out of a topology so that certain route announcements fail to propagate [15]. These attacks do not involve the adversary listening and intercepting data without being noticed. Although in our security model the adversary, having access to all communication, can prevent any link from being operational, we do not capture this attack in our security model because, in general, crypto cannot resolve these attacks due to their physical nature.

Finally, contrary to common intuition, path vector protocols cannot guarantee that a particular route announcement was propagated along the route shown in that announcement. Concretely, no path vector protocol $\mathcal{PV}$ can guarantee for every network $\mathcal{I} \in \mathcal{C}_m^{PV}$, for every efficient adversary A, the following event occurs with negligible probability in $\mathbf{Exp}^{\text{sec-rout-m}}_{\mathcal{I},\mathcal{PV}}(A)$: $N_\ell \in \mathsf{Honest}$ accepts an announcement $(N_{\ell-1}, N_\ell, R = (N_{\ell-1}, \ldots, N_1), P, W, Aux)$ such that there exists $1 \leq i \leq \ell - 1$ so that N_i has never output announcement $(N_i, N_{i+1}, R' = (N_i, \ldots, N_1), P, W', Aux')$ for any W', Aux' to N_{i+1}. Here N_i is not required to be honest as it is in the unauthentic route condition in Section 5. In the full version [19] we present an attack on S-BGP in which colluding corrupted nodes avoid using their expensive link by sending a route announcement through a path of honest nodes between them, and then taking these honest nodes out of the route announcement. Colluding nodes can do that because they can sign on behalf of each other. Note that in real-life scenarios, colluding nodes could belong to a single administration with presence in different geographical locations and multiple distinct AS numbers.

6. HOW SECURE IS S-BGP?

In this section we show that S-BGP guarantees *origin* and *route authentication*, assuming security of the building blocks, but that it is not fully secure because it does not guarantee *route validity*.

Let $\mathcal{SS} = (\mathsf{Kg}, \mathsf{Sign}, \mathsf{Ver})$ be a signature scheme, let $\mathcal{CP}_s = (\mathsf{Kg}, (\mathsf{CA}, \mathsf{U}), \mathsf{Vercert})$ be the corresponding straight-forward certification scheme as per Construction 2.1. Theorems 6.1, 6.2 and 6.3 below state our results; the first two are positive and the last one is negative. The proofs with concrete security statements are presented in the full version [19].

THEOREM 6.1. *S-BGP per Construction 4.1 guarantees* origin authentication *for $\mathcal{C}_0^{\text{S-BGP}}$ if the underlying $\mathcal{SS}$ is uf-cma-secure.*

THEOREM 6.2. *S-BGP per Construction 4.1 guarantees* route authentication *for $\mathcal{C}_0^{\text{S-BGP}}$ if the underlying $\mathcal{SS}$ is uf-cma-secure.*

THEOREM 6.3. *S-BGP as defined in Construction 4.1 does not guarantee route validity for $\mathcal{C}_0^{\text{S-BGP}}$*

7. FULLY SECURE BGP

To address the attack in the proof of Theorem 6.3, we suggest modification to S-BGP and show that the resulting protocol *provably* guarantees route validity assuming the underlying signature scheme is secure. We argue that this modification is necessary. The modified protocol is fully secure (according to our security definition from Section 5) under the same assumption, so we call it *fully secure* BGP or FS-BGP.

CONSTRUCTION 7.1. Let $\mathcal{I} = (\mathbf{G} = (\mathsf{AS's}, \mathsf{link}), \mathsf{Prefixes}, \mathsf{OrforPr}, \mathsf{relation}, \mathsf{preferto}, \mathsf{policy})$ be an interdomain network, let $\mathcal{SS} = (\mathsf{Kg}, \mathsf{Sign}, \mathsf{Ver})$ be a signature scheme, and let $\mathcal{CP}_s = (\mathsf{Kg}_{\mathsf{CA}}, (\mathsf{CA}, \mathsf{U}), \mathsf{Vercert})$ be the corresponding certification protocol as per Construction 2.1. Let S-BGP $= (\mathsf{Init}, \mathsf{An})$ be the construction from Section 4. FS-BGP $= (\mathsf{Init}, \mathsf{An}')$ is defined exactly like S-BGP, but An' requires a few extra operations.

After all address attestations are generated and before any announcement is sent, each node N_j interacts with the CA via $(\mathsf{CA}, \mathsf{U})$. In what follows, smaller input is always on the left corresponding to any link (N_j, N_i), and for convenience only, suppose that $N_j = \min(N_j, N_i)$, for every $N_i \in \mathsf{Neighbors}(N_j)$. For this interaction, the input to U is $(pk_{\mathsf{CA}}, N_j, ((N_j, N_i), \mathsf{relation}(N_j, N_i)))$, the input to CA is $(sk_{\mathsf{CA}}, N_j, ((N_j, N_i), \mathsf{relation}(N_j, N_i)))$ and the outputs of both parties are $(N_j, ((N_j, N_i), \mathsf{relation}(N_j, N_i)), \mathsf{cert})$. We define *link attestation* to be $LA_{N_jN_i} \equiv \mathsf{cert}$. If N_j owns prefix $P \in \mathsf{Prefixes}$, for every $N_i \in \mathsf{policy}_{N_j}((N_j), \varepsilon)$, N_j generates a route attestation $RA^i_{R_j}$ just as in S-BGP and sends $(N_j, N_i, R = (N_j), P, 0, Aux = ((\mathsf{relation}(N_j, N_i), LA_{N_jN_i}), RA^i_{R_j}, AA^P_{N_j}))$ to N_i.

For every new route announcement $(N_{j-1}, N_j, R = (N_{j-1}, \ldots, N_1), P, W, Aux = (\mathsf{relation}(N_{j-1}, N_j), LA_{N_{j-1}N_j}, RA^j_{R_{j-1}}, \ldots \mathsf{relation}(N_1, N_2), LA_{N_1N_2}, RA^2_{R_1}, AA^P_{N_1}))$ that N_j receives, N_j first performs address and route attestation verification just as in S-BGP, and, if these steps do not result in $\perp$, then N_j performs link attestation verification as follows. N_j runs $\mathsf{Vercert}\ (pk_{\mathsf{CA}}, N_i, ((N_i, N_{i+1}), \mathsf{relation}(N_i, N_{i+1})), LA_{N_iN_{i+1}})$, for every $1 \leq i \leq j-1$, and outputs $\perp$ if at least one such computation outputs 0. Otherwise, N_j outputs $\perp$ if there is at least one N_i, for $1 \leq i \leq j-1$, such that $N_{i+1} \notin \mathsf{policy}_{N_i}((N_i, \ldots, N_1), \mathsf{relation}(N_i, N_{i-1}))$.

If none of the verification steps above results in $\perp$, then N_j performs the same operations as N_j would do in S-BGP upon receipt of $(N_{j-1}, N_j, R, P, W, RA^j_{R_{j-1}}, \ldots RA^2_{R_1}, AA^P_{N_1}))$. For every message $(N_j, N_{j+1}, R', P, W', Aux')$ that N_j would send to N_{j+1} as a result in S-BGP, N_j now sends $(N_j, N_{j+1}, R', P, W', Aux'')$ to N_{j+1} instead, where $R' = (N_j, R)$ and $Aux'' = (\mathsf{relation}(N_j, N_{j+1}), LA_{N_jN_{j+1}}, RA^{j+1}_{R'_j}, Aux)$.

Note that FS-BGP is correct for the same classes of networks that BGP is correct for, if the underlying signature scheme $\mathcal{SS}$ used to generate address, route attestations and link attestations is correct. The proof of the following result along with a concrete security statement is in the full version [19].

THEOREM 7.2. *FS-BGP as defined in Construction 7.1 is fully secure for $\mathcal{C}_0^{\text{FS-BGP}}$ if the underlying $\mathcal{SS}$ is uf-cma.*

Assigning link attestations for every link in the Internet may seem impractical because the Internet contains many more edges than AS's (150K versus 40K [24]), their management is harder due to periodic reconfiguration, and AS's may be unwilling to expose their connections, business relationships and export policies. However, we argue that link attestations are necessary to prevent route feasibility attacks in general. If a path vector protocol guarantees route validity, every announcement received as part of this protocol can itself serve the role of a certificate for the links between the nodes in the route of that announcement. Since in our model arbitrary nodes on any route could be corrupted, such certificates

would have to be generated independently by trusted parties. Analogously, to guarantee route validity when export policies of nodes are not publicly known and/or are not next-hop, more sophisticated certificates and in greater amounts (potentially one for every route of every node to every origin) would have to be issued by a trusted authority to ensure that honest nodes can check for export policy violations of remote nodes.

Several plausible solutions to route leaks–unintentional export policy violations—and route validity attacks have been suggested without provable security analysis in [47, 26]. Although these solutions are more practical than FS-BGP because they are mostly based on restricted models of AS's business relationships and export policies, e.g. models presented in [27], it is not clear whether they work with respect to colluding adversarial AS's. Also, because business relationships and export policies of AS's on the Internet may be more complicated than in the model of [27], as we argued above, a more sophisticated solution than what the ones proposed in [47, 26] would be necessary.

In SoBGP [50], Origin Authorization Certificates are used to bind prefixes to certain AS's (just like address attestations in S-BGP) while AS Policy Certificates are used to allow nodes to learn of links and policies of remote nodes. Although similar to link attestations, these certificates are not generated for links by a third trusted party; instead nodes (possibly corrupted) themselves disseminate their neighborship information. In the full version [19] we outline how our security model can be used to formally define SoBGP and prove that it guarantees origin authentication but does not guarantee route authentication and route validity. The latter two points can be shown by constructing attacks similar to those in Theorems 8.2 and 6.3 respectively.

8. PARTIAL DEPLOYMENT OF PKI

In this section we study the effect on security of the partial deployment of PKI. We first show that neither S-BGP nor FS-BGP can guarantee route authenticity for networks in which there is at least a single node without a public key, and then present variants of these protocols with which full security can be guaranteed in partial PKI scenarios.

PARTIAL PKI DEPLOYMENT: INTRODUCTORY RESULTS. We modify S-BGP to allow some nodes to not have public keys.

CONSTRUCTION 8.1. Let $\mathcal{I} = (\mathbf{G} = (\mathsf{AS's}, \mathsf{link}), \mathsf{Prefixes}, \mathsf{OrforPr}, \mathsf{relation}, \mathsf{preferto}, \mathsf{policy})$ be an interdomain network and k a security parameter. We define S-BGP with partial deployment (S-BGP-PD) = $(\mathsf{Init}', \mathsf{An}')$ as a path vector protocol identical to S-BGP = $(\mathsf{Init}, \mathsf{An})$ but with the following modifications. During execution of $\mathsf{Init}(1^k)$ not every node has to generate a public key. During execution of An', nodes that do not have public keys do not generate route attestations, and route attestations of nodes without public keys are not checked during the route attestation verification.

Providing security guarantees in scenarios with partial PKI deployment is a difficult problem because nodes that do not have public keys cannot generate route attestations. With pictorial examples of simple attacks in the full version [19] we show that we can construct an attack on S-BGP-PD for any $m \geq 1$. Let us define FS-BGP-PD to account for partial PKI deployment similarly to Construction 8.1.

THEOREM 8.2. *For no $m \geq 1$ does FS-BGP-PD guarantee route authentication with m-PD for $\mathcal{C}_m^{\text{FS-BGP-PD}}$.*

The proof with a pictorial example is in the full version [19]. The attack in the proof of Theorem 8.2, deserves a special name because we later show it to be the only type of attacks that can prevent FS-BGP-PD from being fully secure later in this section. A similar type of attack was known in the networking community to prevent SoBGP from guaranteeing route authentication.

DEFINITION 8.3 (THE VALID-ROUTE-SWITCHING ATTACK). *Let $\mathcal{I} = (\mathbf{G} = (\mathsf{AS's}, \mathsf{link}), \mathsf{Prefixes}, \mathsf{OrforPr}, \mathsf{relation}, \mathsf{preferto}, \mathsf{policy})$ be a network in $\mathcal{C}_m^{\mathcal{PV}}$, for any $1 \leq m \leq |\mathsf{AS's}|$, such that $|\mathsf{AS's}| \geq 2$, let $\mathcal{PV} = (\mathsf{Init}, \mathsf{An})$ be a path vector protocol correct for $\mathcal{I}$ and let k be the security parameter such that the size of the description of $\mathcal{I}$ is polynomial in k. We consider the experiment $\mathbf{Exp}^{\text{sec-rout-m}}_{\mathcal{I},\mathcal{PV}}(A)$, involving an adversary A.*

When $\mathbf{Exp}^{\text{sec-rout-m}}_{\mathcal{I},\mathcal{PV}}(A)$ outputs 2, i.e. when $N_\ell \in \mathsf{Honest}$ accepts announcement $(N_{\ell-1}, N_\ell, R = (N_{\ell-1}, \ldots, N_1), P, W, Aux)$, such that $\exists 1 \leq i \leq \ell - 1$ so that $N_i \in \mathsf{Honest}$ has never output announcement $(N_i, N_{i+1}, R' = (N_i, \ldots, N_1), P, W', Aux')$ for any W', Aux' to N_{i+1}, if in addition $N_i \in \mathsf{nopubk}$ and R' is a valid route to P, then this event is called a Valid-Route-Switching (VRS) attack.

An honest node N_i may never announce to N_{i+1} a valid route R' to a particular prefix P because N_i may have never received any route announcements to P from its neighbors or because R' is not N_i's most preferred route to P.

WEAKENED PATH VECTOR PROTOCOL SECURITY DEFINITION. We first present a new (weaker) security definition. Next we justify that this relaxation is still reasonable by exploiting physical security of links and the trust relationship that neighboring nodes must have to establish physical links between them. Then we present refinements to S-BGP-PD and FS-BGP-PD that address the weakness pointed out in the proof of Theorem 8.2, and prove that these protocols meet our new definition. We relax the definition from Section 5 as follows.

DEFINITION 8.4. Let $\mathcal{I} = (\mathbf{G} = (\mathsf{AS's}, \mathsf{link}), \mathsf{Prefixes}, \mathsf{OrforPr}, \mathsf{relation}, \mathsf{preferto}, \mathsf{policy})$ be a network in $\mathcal{C}_m^{\mathcal{PV}}$, for any $1 \leq m \leq |\mathsf{AS's}|$, such that $|\mathsf{AS's}| \geq 2$, let $\mathcal{PV} = (\mathsf{Init}, \mathsf{An})$ be a path vector protocol and let k be the security parameter such that the size of the description of $\mathcal{I}$ is polynomial in k. We define experiment $\mathbf{Exp}^{\text{w-sec-rout-m}}_{\mathcal{I},\mathcal{PV}}(A)$ involving adversary A to be identical to the experiment $\mathbf{Exp}^{\text{sec-rout-m}}_{\mathcal{I},\mathcal{PV}}(A)$ involving an adversary A from the definition from Section 5 except for the following two additional relaxations.

1. (Physical-Link-Security Relaxation) A is not allowed to (i) send announcements on behalf of honest neighboring nodes and (ii) intercept and modify announcements exchanged between neighboring honest nodes.

2. (Trusted-Next-Neighbor Relaxation) Whenever experiment $\mathbf{Exp}^{\text{sec-rout-m}}_{\mathcal{I},\mathcal{PV}}(A)$ outputs 2, i.e. $N_\ell \in \mathsf{Honest}$ accepts announcement $(N_{\ell-1}, N_\ell, R = (N_{\ell-1}, \ldots, N_1), P, W, Aux)$, and there exists $1 \leq i \leq \ell - 1$ such that $N_i \in \mathsf{Honest}$ never output $(N_i, N_{i+1}, R' = (N_i, \ldots, N_1), P, W', Aux')$ for any W', Aux' to N_{i+1}, $N_{i+1} \in \mathsf{Honest}$ if $N_i \in \mathsf{nopubk}$.

We define A's advantage $\mathbf{Adv}^{\text{w-sec-rout-m-b}}_{\mathcal{I},\mathcal{PV}}(A)$ in this experiment as $\Pr\left[\mathbf{Exp}^{\text{w-sec-rout-m}}_{\mathcal{I},\mathcal{PV}}(A) = b\right]$, for $b \in \{1, 2, 3\}$. We say that $\mathcal{PV}$ guarantees *weakened origin authentication*, *route authentication*, and *route validity* with m-PD for a class of networks $\mathcal{C}_m^{\mathcal{PV}}$, if for every network $\mathcal{I} \in \mathcal{C}_m^{\mathcal{PV}}$, for every efficient adversary A the probability experiment $\mathbf{Exp}^{\text{w-sec-rout-m}}_{\mathcal{I},\mathcal{PV}}(A)$ returns 1, 2 and 3 respectively while Relaxations 1-2 hold is negligible in k. The weakened full security is defined analogously to security definition in Section 5.

RELEVANCE OF THE WEAKENED DEFINITION. We argue that the above definition is relevant in practice because the adversary's behavior restricted by the above two conditions corresponds to the adversary's behavior limited in practice by the presence of some physical or external security features.

The first relaxation is justified if honest neighboring nodes, whether with public keys or not, can establish a communication channel that guarantees authentication and integrity. This could be indeed the case. All nodes could establish communication channels with their neighbors via IPSec that could guarantee integrity and authenticity, for which they do not need public keys as they could establish a pre-shared keys off line since they would have to establish a business relationship to have a physical connection anyway. BGP TTL security hack [48] could also be used for this purpose. Although physical attacks on links between nodes are possible and have been studied [15], they do not involve listening and intercepting data without being noticed. Thus, such attacks do not invalidate this relaxation as their only purpose is to take out links out of a topology so that certain route announcements are never made.

The second relaxation is justified if the nodes that do not have public keys have trusted down-stream neighbors with public keys, and the latter can "vouch" for the former with their signatures. This is not unrealistic, since neighboring nodes must trust in each other to establish a business relationship between themselves in the first place. Moreover, framing business partners may result in devastating consequences such as the tearing down of their business contracts and physical links connecting them, which could result in substantial financial losses.

On the Internet, most connections between AS's are made at public or private Internet Exchange Points (IXP) which, intuitively, serve the role of rendez-vous points for AS's to exchange traffic. AS's that wish to connect at a particular IXP have to physically come and make a connection at that IXP. Thus, since IXP's make a profit by providing basic infrastructure for AS's to make connections and become neighbors, it would be in their interest to facilitate the establishment of physically secure communication channels and trust between neighboring AS's, as this would guarantee longer lasting business relationships for those AS's (which would equate to longer lasting profits for the IXP connecting them).

SECURE CONSTRUCTIONS. We slightly modify S-BGP-PD and then show that it meets the above definition.

CONSTRUCTION 8.5. Let $\mathcal{I} = (\mathbf{G} = (\mathsf{AS's}, \mathsf{link}), \mathsf{Prefixes}, \mathsf{OrforPr}, \mathsf{relation}, \mathsf{preferto}, \mathsf{policy})$ be an interdomain network. We define S-BGP-PD with a restriction (S-BGP-PDR) = $(\mathsf{Init}, \mathsf{An}')$ as a path vector protocol identical to S-BGP-PD = $(\mathsf{Init}, \mathsf{An})$ but with the following restrictions in An'. When a node receives an announcement of a route, that node rejects the announcement if that route contains more than one node without public keys in a row at any part of that route. Also, a node without a public key does not propagate a route that was announced by its neighbor who also does not have a public key.

We define FS-BGP-PD with a restriction (FS-BGP-PDR) similarly; note that in S-BGP-PDR and FS-BGP-PDR, the last two nodes on a route could be without public keys. This new restriction implicitly requires that nodes reject announcements that are missing a signature for at least one node in that route who has a public key. Although checking whether a node has a public key or not may be difficult in practice, this is in fact necessary, otherwise an adversarial node could simply strip an honest node's signature and send a bogus route on its behalf. The proof of the following Theorem is in the full version [19].

THEOREM 8.6. *S-BGP-PDR as defined in Construction 8.5 guarantees weakened route authentication with m-PD for $\mathcal{C}_m^{\text{S-BGP-PDR}}$, for any $m \leq |\mathsf{AS's}|$, if the underlying $\mathcal{SS}$ is uf-cma-secure.*

COROLLARY 8.7. *FS-BGP-PDR is weakened fully secure with m-PD for $\mathcal{C}_m^{\text{FS-BGP-PDR}}$, for $m \leq |\mathsf{AS's}|$, if the underlying $\mathcal{SS}$ and $\mathcal{CP}$ are uf-cma-secure and uf-cda-secure respectively.*

The following results emphasize that the restrictions in the weakened path vector protocol security definition posed by Relaxations 1-2 and the requirement to ignore routes that have more than one node without a public key in a row, as is done in S-BGP-PDR and FS-BGP-PDR, are in fact necessary. The latter restriction, in the worst case, could cause some parts of the network to become disconnected as many routes may be ignored. The proofs are presented in the full version [19].

THEOREM 8.8. *For the statements in Theorem 8.6 and Corollary 8.7 to hold, each relaxation (Physical-Link-Security or Trusted-Next-Neighbor) is necessary given the other one.*

THEOREM 8.9. *Even when the underlying $\mathcal{SS}$ is uf-cma-secure, S-BGP-PD as per Construction 8.1 and FS-BGP-PD do not guarantee weakened route authentication with m-PD for $\mathcal{C}_m^{\text{S-BGP-PDR}}$ and $\mathcal{C}_m^{\text{FS-BGP-PDR}}$ respectively, for any $m \geq 2$.*

A significant practical implication of Theorem 8.6 and Corollary 8.7 is that new AS's who have just joined the Internet but do not have public keys, do not have to get a public key as long as they establish a trust relationship with their neighbors in the sense that for any route announcement that they make, they are sure that their neighbors who have public keys will vouch for them.

Even if we do not rely on security Relaxation 2, we can still show that it is possible to guarantee route authentication but with a very restricted version of S-BGP-PD, where only the last two nodes on any route are allowed not to have public keys. We provide the details in the full version [19].

WHAT IF THERE IS NO PKI. We show that if all prefixes and links are certified by a trusted certification authority, even when no node has a public key, nodes are guaranteed to discover valid routes with authentic origins, and that VRS attacks are the *only* attacks that prevent FS-BGP-PD from guaranteeing route authentication. In light of this result, we then discuss the feasbility of achieving reasonable security without PKI. The proof of the following is in the full version [19].

THEOREM 8.10. *If the underlying $\mathcal{SS}$ is uf-cma-secure and the underlying $\mathcal{CP}$ is uf-cda-secure, for any $1 \leq m \leq |\mathsf{AS's}|$, if $\mathbf{Exp}^{\text{sec-rout-m}}_{\mathcal{I},\text{FS-BGP-PD}}(A) = 2$ (see security definition in Section 5), then A must have carried out a VRS attack.*

The goal of path vector protocols is for nodes to learn of routes in the network to all prefixes, so the importance of Theorem 8.10 is that FS-BGP-PD guarantees that nodes learn of valid routes with authentic origins and that, even without PKI, the worst thing that can happen compared to when FS-BGP is deployed, is that due to a VRS attack, at least one honest node N_ℓ accepts at least one route $R = (N_{\ell-1}, \ldots, N_1)$ to some prefix P such that for at least one honest node N_i in R, R's subroute $(N_{i-1}, \ldots, N_1)$ is not N_i's the most preferred route to P. Although requiring link-attestations diminishes the practical gains of having no PKI, having no PKI is still very practical and facilitates gradual, Internet-wide deployment of FS-BGP-PD as it relieves nodes of storing public keys of all other nodes and generating signatures for their every announcement. It

also reduces communication overhead by getting rid of nodes' signatures.

With respect to adversarial control of the flow of traffic on the Internet, Theorem 8.10 is a major milestone in understanding the security and efficiency tradeoffs that can be achieved in full versus no PKI deployment. Although with a VRS attack an adversary could cause an honest node to send traffic along an unintended route without that node's knowledge, the adversary could do the same without a VRS attack by simply diverting traffic to an unintended route of its choosing without the source's knowledge. The latter is an issue of data-plane accountability, and if the Internet does not deploy a provably secure accountability protocol, e.g. [11, 31], then FS-BGP-PD with no PKI is just as good as with fully deployed PKI with respect to such an adversary. On the other hand, the only provably secure accountability protocols that are known to date require nodes to deploy a PKI or have shared keys, so having no PKI for FS-BGP-PD would yield no practical gains if the Internet does deploy a provably secure accountability protocol. Thus, in the beginning stages of partial deployment of secure path vector protocols, it may be more beneficial to deploy link certificates rather than have some nodes possess public keys but deploy no link certificates at all.

Currently, IETF [4] is considering standardizing a variant of S-BGP, called BGPSEC [37], that could work together with RPKI [5, 36]. RPKI consists of a hierarchy of authorities and AS's for certifying IP prefixes and AS numbers. Certificates for IP prefixes and AS numbers also contain certified public keys that are generated by the entities receiving the certificates. These keys would be used to run S-BGP (or its variants like BGPSEC), and the results in this section apply to settings when either RPKI is partially deployed (i.e. not every AS gets a certificate for a prefix and a key) or RPKI is fully deployed but some AS's choose not to use their private keys to generate S-BGP's route attestations. In the full version [19] we further discuss these scenarios. We note, however, that if an adversary is allowed to corrupt various nodes in the RPKI (i.e. entities that generate and/or certify keys may be corrupted), as we suggested in Section 5, to have well-defined, provable security guarantees in such scenarios, more sophisticated models and protocols would be needed to address rogue key and certificate attacks.

9. CONCLUSIONS

We developed the framework for the provable-security treatment of path vector routing protocols. We defined an interdomain network, a path vector protocol and designed a formal security model for such protocols, which incorporates three general security requirements and is quite strong in terms of adversarial capabilities. Using our framework we analyzed security of the Secure BGP protocol. Assuming the underlying signature scheme is secure, we proved that S-BGP meets two out of the security definition's three requirements and showed how the protocol can be modified to meet all tree security requirements at the same time. We also studied SoBGP and showed that it fails to meet two security goals. Finally, we studied security of partial PKI deployment when not all nodes have public keys. We investigated the possibilities of relaxing the PKI requirement while relying on the non-cryptographic physical security of the protocol in order to achieve possibly weaker, but still well-defined, notions of security. We also presented the necessary and sufficient conditions to achieve full security in the partial PKI deployment scenario. We believe our results fill the gap between the advances of modern cryptography and provable security methodology and practical networking protocols, and should be useful for protocol developers, standards bodies, and government agencies.

10. ACKNOWLEDGMENTS

We thank Nick Feamster, Vytautas Valancius and the anonymous reviewers for very useful comments. We also thank Mamta Upadhya for her participation in the early stages of the project.

11. REFERENCES

[1] American Registry for Internet Numbers (ARIN). `https://www.arin.net/`.

[2] BGP Routing table analysis reports. `http://bgp.potaroo.net`.

[3] The Internet Assigned Numbers Authority (IANA). `http://www.iana.org/`.

[4] Internet Engineering Task Force (IETF) secure inter-domain routing group (SIDR). `http://datatracker.ietf.org/wg/sidr/charter/`.

[5] Resource Public Key Infrastructure (RPKI). `https://www.arin.net/resources/rpki.html`.

[6] C. Adams and S. Farrell. Internet X.509 Public Key Infrastructure: Certificate management protocols, 2004.

[7] W. Aiello, J. Ioannidis, and P. McDaniel. Origin authentication in interdomain routing. In *CCS '03: Proceedings of the 10th ACM conference on Computer and communications security*, pages 165–178, New York, NY, USA, 2003. ACM Press.

[8] G. Andersen, H. Balakrishnan, N. Feamster, T. Koponent, D. Moon, and S. Shenker. Accountable internet protocol (AIP). In *ACM SIGCOMM 2008*, Aug. 2008.

[9] M. Backes, I. Cervesato, A. D. Jaggard, A.Scedrov, and J.-K. Tsay. Cryptographically sound security proofs for basic and public-key Kerberos. In D. Gollmann, J. Meier, and A. Sabelfeld, editors, *ESORICS*, volume 4189 of *Lecture Notes in Computer Science*, pages 362–383. Springer, 2006.

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

[11] B. Barak, S. Goldberg, and D. Xiao. Protocols and lower bounds for failure localization in the Internet. In *EUROCRYPT 2008*, Apr. 2008.

[12] A. Barbir, S. Murphy, and Y. Yang. Generic threats to routing protocols. *Network Working Group. IETF Request for Comments: 3962. Available at* `http://www.ietf.org/rfc/rfc4593.txt`, 2004.

[13] R. Barrett, S. V. Haar, and R. Whitestone. Routing snafu snips net service. Interactive Week, 1997. `http://www.zdnet.com/zdnn/content/inwk/0413/inwk0032.html`.

[14] M. Bellare, T. Kohno, and C. Namprempre. Authenticated encryption in SSH: provably fixing the SSH binary packet protocol. In *CCS '02*. ACM Press, 2002.

[15] S. M. Bellovin and E. R. Gansner. Using link cuts to attack internet routing. In *Tech. Rep., ATT Research, 2004, Work in Progress 2003 USENIX*, 2003.

[16] S. M. Bellovin, J. Ioannidis, and R. Bush. Position paper: Operational requirements for secured BGP. DHS Secure Routing Workshop, 2005.

[17] A. Boldyreva, M. Fischlin, A. Palacio, and B. Warinschi. A closer look at PKI: Security and efficiency. In T. Okamoto and X. Wang, editors, *Public Key Cryptography*, volume 4450 of *Lecture Notes in Computer Science*, pages 458–475. Springer, 2007.

[18] A. Boldyreva and V. Kumar. Extended abstract: Provable-security analysis of authenticated encryption in Kerberos. In *IEEE Symposium on Security and Privacy*, pages 92–100. IEEE Computer Society, 2007.

[19] A. Boldyreva and R. Lychev. Provable Security of (S-BGP) and other Path Vector Protocols: Model, Analysis, and Extenstions. Full Version of this Paper, 2012. Available from the authors' websites.

[20] M. A. Brown. Renesys blog. Pakistan hijacks YouTube, 2008. `http://www.renesys.com/blog/2008/02/pakistan_hijacks_youtube_1.shtml`.

[21] K. Butler, T. Farley, P. McDaniel, and J. Rexford. A survey of BGP security issues and solutions. Technical Report TD-5UGJ33, AT&T Labs, 2004.

[22] 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 Press.

[23] H. Chan, D. Dash, A. Perrig, and H. Zhang. Modeling adoptability of secure BGP protocols. In *SIGMETRICS '06/Performance '06: Proceedings of the joint international conference on Measurement and modeling of computer systems*, pages 389–390, New York, NY, USA, 2006. ACM Press.

[24] Y.-J. Chi, R. Oliveira, and L. Zhang. Cyclops: The internet as-level observatory. *ACM SIGCOMM Computer Communication Review*, 2008.

[25] J. Cowie. Renesys blog. China's 18-minute mystery, 2010. `http://www.renesys.com/blog/2010/11/chinas-18-minute-mystery.shtml`.

[26] B. Dickson. Route Leaks – Requirements for Detection and Prevention thereof (v2). IETF Internet Draft, 2012. Available at `http://tools.ietf.org/html/draft-dickson-sidr-route-leak-reqts-02`.

[27] L. Gao and J. Rexford. Stable internet routing without global coordination. *SIGMETRICS Perform. Eval. Rev.*, 28:307–317, June 2000.

[28] P. Gill, M. Schapira, and S. Goldberg. Let the market drive deployment: A strategy for transitioning to BGP security. In *ACM SIGCOMM 2011*, Aug. 2011.

[29] S. Goldberg, S. Halevi, A. Jaggard, V. Ramachandran, and R. Wright. Rationality and traffic attraction: Incentives for honestly announcing paths in BGP. In *ACM SIGCOMM 2008*, Aug. 2008.

[30] S. Goldberg, M. Schapira, P. Hummon, and J. Rexford. How secure are secure interdomain routing protocols? In *ACM SIGCOMM 2010*, Aug. 2010.

[31] S. Goldberg, D. Xiao, B. Barak, J. Rexford, and E. Tromer. Path-quality monitoring in the presence of adversaries. In *ACM SIGMETRICS 2008*, June 2008.

[32] G. Goodell, W. Aiello, T. Griffin, J. Ioannidis, P. McDaniel, and A. Rubin. Working around bgp: An incremental approach to improving security and accuracy in interdomain routing, 2003.

[33] Y.-C. Hu, A. Perrig, and M. Sirbu. SPV: secure path vector routing for securing BGP. In *SIGCOMM '04: Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 179–192, New York, NY, USA, 2004. ACM Press.

[34] S. Kent, C. Lynn, and K. Seo. Secure border gateway protocol (S-BGP). *IEEE Journal on Selected Areas in Communications*, 18(4):582–592, 2000.

[35] S. T. Kent, C. Lynn, J. Mikkelson, and K. Seo. Secure Border Gateway Protocol (S-BGP) - Real world performance and deployment issues. In *NDSS*. The Internet Society, 2000.

[36] M. Lepinski. An infrustructure to support secure internet routing.

[37] M. Lepinski. BGPSEC Protocol Specification (v4). IETF Internet Draft, 2012. Available at `http://tools.ietf.org/html/draft-ietf-sidr-bgpsec-protocol-04`.

[38] R. Mahajan, D. Wetherall, and T. Anderson. Understanding bgp misconfiguration. In *ACM SIGCOMM 2002*, Aug. 2002.

[39] A. Mityagin, S. Panjwani, and B. Raghavan. Analysis of the SPV secure routing protocol. Cryptology ePrint Archive, Report 2006/087, 2006. `http://eprint.iacr.org/`.

[40] S. Murphy. BGP security vulnerabilities analysis. *Network Working Group. IETF Request for Comments: 3962. Available at* `http://www.ietf.org/rfc/rfc4272.txt`, 2006.

[41] D. of Homeland Security. The national strategy to secure cyberspace, 2003. `http://www.whitehouse.gov/pcipb/`.

[42] U. of Oregon Route Views Project. `http://www.routeviews.org`.

[43] K. G. Paterson and G. J. Watson. Plaintext-dependent decryption: A formal security treatment of SSH-CTR. In H.Gilbert, editor, *EUROCRYPT*, volume 6110 of *Lecture Notes in Computer Science*, pages 345–361. Springer, 2010.

[44] Y. Rikhter, T. Li, and S. Hares. A Border Gateway Protocol 4 (BGP-4). *Network Working Group. IETF Request for Comments: 4271. Available at* `http://www.ietf.org/rfc/rfc4271.txt`, 2006.

[45] T. Ristenpart and S. Yilek. The power of proofs-of-possession: Securing multiparty signatures against rogue-key attacks. In M. Naor, editor, *EUROCRYPT*, volume 4515 of *Lecture Notes in Computer Science*, pages 228–245. Springer, 2007.

[46] L. Subramanian, V. Roth, I. Stoica, S. Shenker, and R. Katz. Listen and whisper: Security mechanisms for BGP. In *Proc. First Symposium on Networked Systems Design and Implementation (NSDI)*, San Francisco, CA, Mar. 2004.

[47] S. Sundaresan, R. Lychev, and V. Valancius. Preventing attacks on BGP policies: One bit is enough. Technical Report GT-CS-11-07, Georgia Institute of Technology, 2011.

[48] The BGP TTL Security Hack. `http://tools.ietf.org/html/draft-gill-btsh-02`.

[49] T. Wan, E. Kranakis, and P. C. van Oorschot. Pretty secure BGP, psBGP. In *NDSS*. The Internet Society, 2005.

[50] R. White. Securing BGP through secure origin BGP. *The Internet Protocol Journal*, 6(3), Sept. 2003. `http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_6-3/ipj_6-3.pdf`.

[51] M. Zhao, S. W. Smith, and D. M. Nicol. Aggregated path authentication for efficient BGP security. In *CCS '05: Proceedings of the 12th ACM conference on Computer and communications security*, pages 128–138, New York, NY, USA, 2005. ACM Press.