Efficient Security Mechanisms for The Border Gateway Routing Protocol[1]

Bradley R. Smith and J.J. Garcia-Luna-Aceves

Computer Engineering Department, Jack Baskin School of Engineering
University of California, Santa Cruz CA 95064
{brad,jj}@cse.ucsc.edu

Abstract

We analyze the security of the BGP routing protocol and identify a number of vulnerabilities in its design and the corresponding threats. We then present modifications to the protocol that minimize or eliminate the most significant threats. The innovation we introduce is the protection of the second-to-last hop information contained in the `AS_PATH` attributes by digital signatures, and the use of this predecessor information to verify the path of the selected route. With these techniques, we are able to secure complete path information in near constant space, avoiding the recursive protection mechanisms proposed for BGP in the past.

Key words: Border Gateway Routing protocol. BGP. Routing protocol security. Path-finding routing protocol.

1 Introduction

Inter-domain routing protocols are designed to perform policy-based routing in an internet of autonomous systems. An autonomous system (AS) is defined as a set of routers and networks under a single technical administration, using an interior gateway protocol and common metrics to route packets within the AS, and using exterior gateway protocols to route packets to other ASs. In practice, this definition is relaxed to allow multiple intra-domain protocols and several sets of metrics. Two inter-domain routing protocols currently defined are the Border Gateway Protocol (BGP) [21] and the Inter-Domain Routing Protocol (IDRP) [20,7]; these two protocols are of particular interest because

[1] This work was supported in part by the Defense Advanced Research Projects Agency (DARPA) under Grant F19628-96-C-0038.

of their current roles as the inter-domain protocols maintaining the global Internet routing infrastructure.

Routing protocols dynamically configure the packet forwarding function in internets, which allows for the continued delivery of packets in spite of changes in network topology and usage patterns. These changes typically occur due to the introduction, failure, and repair of network links and routing nodes. The compromise of the routing function in an internet can lead to the denial of network service, the disclosure or modification of sensitive routing information, or the diversion of network traffic via the reconfiguration of the logical routing structure in the internet, which can lead to the disclosure of network traffic to an attacker or the inaccurate accounting of resource utilization.

Current routing protocols contain few, if any, mechanisms to provide for the security of their operation. Those that exist are often incomplete. For example, the security mechanisms currently defined for BGP and RIPv2 [14] protect the transmission of routing messages across local networks; however, they do not provide integrity or authenticity of the routing information itself as it traverses an internet. These mechanisms require trust of neighbors regarding updates describing the full internet and, transitively, similar trust of all routers in an internet. More recent efforts have addressed both the security of routing message transmission and of the routing information itself; however, they have addressed only link-state protocols based on the reliable broadcast of topology information [17]. While this class of routing protocols lends itself to simple means of securing routing information in a manner that effectively limits the scope of trust, it also involves considerable computation and space overhead that limit its usability in large scale internets. Furthermore, the mechanisms for securing routing information in a link-state protocol do not apply directly to BGP, which is based on the exchange of complete path information among neighbor routers. Given the evolution of the global Internet to a commercial, production network infrastructure, and the prominent role of BGP in it, this state of affairs is clearly unacceptable. This paper presents a strategy for securing BGP using the following methodology:

(1) Analyze the protocol design to identify vulnerabilities and threats.
(2) Identify the security services needed to reduce or eliminate the vulnerability.
(3) Design the appropriate countermeasures to provide the needed services.

Section 2 states our assumptions and goals for securing BGP. Section 3 analyzes the security of BGP, and identifies its vulnerabilities and the threats to which it is susceptible. Section 4 presents our proposed strategies and countermeasures for securing BGP. Section 5 reviews related work.

2 Assumptions of the BGP Environment

There are four basic components in a BGP system: speakers, peers, links, and border routers [21].

A *BGP speaker* is a host in the network that executes the BGP protocol. *BGP peers* are two BGP speakers that form a connection and engage in a BGP dialog. A BGP peer is either an internal or external peer, depending on whether it is in the same or a different AS as the reference BGP speaker. The connections between BGP peers are called *links*, with internal and external links being defined similarly to internal and external peers. BGP links are formed using a reliable transport protocol such as TCP. This eliminates the need to implement transport services such as retransmissions, acknowledgments, and sequence numbers in the routing protocol.

A *border router* is a router with an interface to a physical network shared with border routers in other autonomous systems. Similar to BGP speakers, border routers are either internal or external. Note that BGP speakers need not be border routers (or even routers of any kind). It is possible that a non-routing host could serve as the BGP speaker, gathering routing information from internal or other external routing protocols, and advertising that information to internal and neighboring external border routers. This feature is currently in use in the Route Servers of the Routing Arbiter project [4].

We make the following assumptions in designing security mechanisms for BGP:

- The BGP version 4 protocol as defined in RFC1771 [21].
- A BGP speaker can trust its internal peers.
- A BGP speaker can trust information it receives from external speakers only concerning links incident on the AS to whom the external speaker belongs.
- Intruders have capabilities as described in Section 3.1.
- Key distribution is based on domain names, which can be efficiently and securely determined given an IP address of a host, and the key distribution mechanism provides a controllable refresh rate. [2]

3 BGP Threats and Vulnerabilities

We now identify the threats to which BGP is susceptible, and the vulnerabilities these threats exploit. We consider separately threats to the flow of routing traffic and threats to the flow of data traffic that involve portions of the routing infrastructure. We describe attacks in terms of different classes

[2] The DNS Security Extensions [3] might meet these requirements.

of internet nodes including authorized BGP speakers and intruders. Authorized BGP speakers are those nodes intended by the authoritative network administrator to perform as a BGP speaker.

3.1 Intruders

We assume that an intruder can be located at any point in the network through which all traffic of interest flows, and that the intruder has the capability to fabricate, replay, monitor, modify, or delete any of this traffic. Interpreting this description for a BGP environment, we identify the following four general classes of intruders:

Subverted BGP speaker: A subverted BGP speaker occurs when an authorized BGP speaker is caused to violate the BGP protocols, or to inappropriately claim authority for network resources. This typically occurs due to bugs in the BGP software, mistakes in the speaker's configuration, or by causing a BGP speaker to load unauthorized software or configuration information, which can be achieved by many means, depending on the design and configuration of the BGP speaker.

Unauthorized BGP speaker: An unauthorized BGP speaker exists when a node that is not authorized as a BGP speaker manages to circumvent any access control mechanisms in place, and establish a BGP link with an authorized BGP speaker. How this is achieved depends on the design and configuration of existing access control mechanisms.

Masquerading BGP speaker: A masquerading BGP speaker occurs when a node successfully forges an authorized BGP speaker's identity. This can be accomplished using the IP spoofing [15] or source routing attacks.

Subverted link: There are a number of forms that a subverted link can take. One is to gain access to the physical medium (e.g. copper or fiber optic cable-plant, the "air-waves", or the electronics used to access them) in a manner that allows some control of the channel. In addition, a link may be subverted by compromising lower layer protocols in use on the link in a manner that allows control of the channel. An example of such an attack is the TCP session hijacking attack [8].

3.2 Threats to Routing Information

Under the correct circumstances an intruder can fabricate, modify, replay, or delete routing traffic. With these capabilities, an intruder can compromise the network in a number of ways. The modification or fabrication of routing updates allows an intruder to reconfigure the logical routing structure of an internet, potentially resulting in the denial of network service, the disclosure of network traffic, and the inaccurate accounting of network resource usage. The replay or deletion of routing updates blocks the evolution of subsets of the logical routing structure (in response to topological or policy changes), or resets it to an earlier configuration with results similar to above. The vulnerability exploited by these attacks is the lack of access control, authentication, and integrity of BGP message contents.

In addition, it is relatively easy for an intruder to gain access to routing traffic. The information available from this traffic includes the appropriate next hop to reach a destination, and the path taken by traffic to different destinations. The next hop information is available from other sources, such as monitoring authorized traffic to the desired destination for the next hop it uses, and therefore cannot be protected solely by measures directed at the routing traffic. However, in some circumstances, the path used to reach different destinations may be considered confidential. The vulnerabilities exploited by these attacks are the lack of confidentiality of peer links and the level of trust placed in BGP speakers.

3.3 Threats to Data Traffic

It is relatively easy for an intruder to snoop or disclose data traffic. The vulnerability exploited to accomplish this is the lack of end-to-end or link encryption services for data traffic. We will not address the possible countermeasures to these attacks, because they should be implemented, in the link, network, or transport layer data transfer protocols such as Ethernet, IP or TCP, which is beyond the scope of our intended modifications to BGP.

It is also relatively easy for an intruder to fabricate, modify, replay, or delete data packets. The effectiveness of these attacks at deceiving or disrupting the source and destination processes depends on the end-to-end protocols in use at the transport layer and above, and is not a routing-protocol issue. However, the effectiveness of these attacks at deceiving the intermediate routing nodes is not an end-to-end protocol issue. Countermeasures to these vulnerabilities will depend on mechanisms in the network or lower layers of the protocol hierarchy. The appropriateness and effectiveness of end-to-end vs. link layer security measures is a fundamental issue in the design of the Internet protocols [11,22,26]. While in general these issues do not involve routing protocol mechanisms, two exceptions include the ability to use multiple paths to a single destination, and the inclusion of authentication and access control mechanisms in the packet forwarding function [5]; we will not address these measures further in this paper.

3.4 Goals for Securing BGP

In general, our goal in securing BGP is to provide authenticity, integrity, confidentiality, and access control of BGP message transmission. Referring to the previous sections, this goal translates specifically to preventing:

- The fabrication, modification, and replay of routing messages by all classes of intruder.
- The deletion of routing messages by subverted links and subverted speakers.
- The disclosure of routing messages by all classes of intruder.

In the following, we assume that access control is provided using the same naming and key distribution mechanism used to implement the authentication mechanism. The remaining access control design issues, such as the definition of the access control lists and their distribution mechanism, are orthogonal to the countermeasures presented here, and are not discussed further.

4 BGP Security Countermeasures

Two classes of communication occur in BGP and routing protocols in general: between neighboring speakers, and between a given speaker and an arbitrary set of remote speakers determined dynamically by routing decisions. That communication between neighboring speakers is composed of routing updates for destinations that the sender has determined are appropriate to send to the receiver. The communication between a speaker and remote speakers is composed of the fields of routing updates which describe a given destination. Accordingly, we present the following two classes of countermeasures:

BGP Message Protection Countermeasures:

- Encrypt all BGP messages between peers using session keys exchanged at BGP link establishment time. This encryption provides integrity and authenticity of all path attributes whose values are valid for at most one AS hop, and confidentiality of all routing exchanges.
- Add a message sequence number to protect against replayed or deleted messages.

BGP Update Field Protection Countermeasures:

- Add an `UPDATE` sequence number or timestamp to protect against replayed `UPDATE` messages.
- Add a `PREDECESSOR` path attribute indicating the AS prior to the destination AS for the current route. This allows the verification of the path information using the `AS_PATH` path attribute.

- Digitally sign all unchanging UPDATE fields whose values are fixed on creation by the BGP speaker originating or most recently aggregating the route. This provides for the integrity and authenticity of not only these fields, but also of the full AS_PATH.

The rest of this section presents a more detailed description of these countermeasures, and an analysis of the effectiveness of these countermeasures against the threats and vulnerabilities identified previously.

4.1 BGP Message Protection Countermeasures

The purpose of these countermeasures is to provide authentication, confidentiality, and integrity of the routing messages between BGP peers, which compose the first class of communication described above. Specifically, the message encryption and message sequence number provide corruption detection, sequencing, acknowledgment, and retransmission mechanisms. While these mechanisms are redundant to those provided by TCP, they are required due to the insecurity of the TCP mechanisms [2,10]. As discussed by Tardo [25], these countermeasures are most appropriately provided at the network or transport layers. These BGP countermeasures would no longer be required if a secure network [1] or secure transport protocol [9,19] were used.

4.1.1 Message Encryption

Upon establishment of each BGP link, a session key is exchanged by the peers to encrypt each BGP message transmitted over that link. This encryption provides confidentiality of messages, as well as authenticity and integrity of KEEPALIVE MESSAGES, NOTIFICATION messages, and some of the path attributes carried in UPDATE messages.

A number of path attributes carried in UPDATE messages are modified in each AS they transit. These include the NEXT_HOP, MULTI_EXIT_DISC, and LOCAL_PREF attributes. The use of peer-to-peer encryption for authenticity and integrity of these path attributes is based on two observations: (a) the recipient of these path attributes receives them from either the most recent modifier or via a single relay that is an internal peer, and (b) our assumption that internal peers are trusted. Given these, peer-to-peer encryption provides a high degree of security in an efficient manner. On detection of corrupted information, the link is terminated using a NOTIFICATION message.

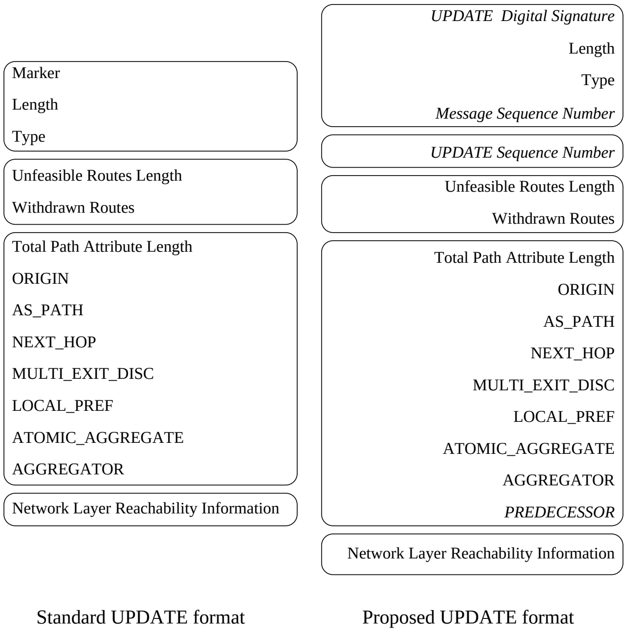

Fig. 1. Proposed UPDATE Message Changes

4.1.2 Message Sequence Number

A sequence number is added to each message; it is initialized to zero on establishment of a BGP link, and is incremented with each message. On detection of a skipped or repeated sequence number, the BGP link is terminated with a `NOTIFICATION` message. The size of the sequence number is made large enough to minimize the chance of it cycling back to zero. However, in the event that it does, the link is terminated and a new link is established, resetting the sequence number to zero and establishing a new session key.

4.2 UPDATE Field Protection Countermeasures

The countermeasures presented in this section protect the communication between a given speaker and a set of remote speakers. These countermeasures provide only for authenticity and integrity of this communication, because confidentiality of this communication is unnecessary as the potential recipients include all authorized BGP speakers in an internet. As discussed by Tardo [25], these countermeasures provide authentication and integrity of fields within a message, and are most appropriately implemented in the presentation layer.

Figure 1 illustrates the proposed modifications using the `UPDATE` message, which includes all proposed new fields, as a model.

4.2.1 UPDATE Sequence Number or Timestamp

Sequence information is added to each `UPDATE` message to protect against the replay of old routing information. This sequence information is generated for each route output from the BGP decision process, and can be in the form of a sequence number or a timestamp. While a number of `UPDATE` messages may be generated for each route (one message per peer of the originating speaker), only one sequence number or timestamp is used for all of them.

This sequence information is necessary because a remote speaker may receive the same route in multiple updates, each describing the same destination but representing different paths, and all of these `UPDATES` must be considered valid. This implies that `UPDATES` for a given destination must be considered valid if their sequence information is greater than or equal to the current sequence information. Note that sequence information must be maintained and validated on a per speaker basis. An invalid `UPDATE` message is dropped silently.

In a BGP environment, sequence numbers would have a potentially long life. Given the recommended value of BGP's `MinASOriginationInterval` timer (15 seconds) the sequence number can be relatively small and still be assured of not cycling. Setting this timer to as low as 8 seconds, and assuming a new `UPDATE` is originated at the end of every interval, a four octet sequence number would last for over 1000 years. The main difficulty introduced by a sequence number consists of maintaining it in the context of arbitrary software and hardware failures. Techniques such as those proposed by Perlman [18] could be used; however, if cycling of the sequence number must be supported, the following process can be used:

> Each BGP speaker maintains an `UPDATE` message sequence number database on a per BGP speaker $< domainname, publickey >$ pair basis. When the cycling of a sequence number approaches, a new public-key pair is generated. The key distribution mechanism and BGP speaker are updated with the new key pair, and the speaker's `UPDATE` sequence number is reset to zero. On detection of a change in the public key for an originating speaker, the receiving speaker will add an entry to its `UPDATE` sequence number database for the new originating speaker $< domainname, publickey >$ pair with a sequence number of zero. It will continue to use the old sequence number entry until a sequence number failure occurs where the digital signature validation succeeds using the new entry. At this time the old entry is purged, and the conversion to a new sequence number is complete. Further work is needed on a mechanism to load the database of a newly-booted BGP speaker.

A timestamp could be used instead of a sequence number. The main benefit of a timestamp would be the ease of administration provided by the well-defined external reference for use in resetting lost state. The life of even a small timestamp, while not as long as for sequence numbers, is still significant; assuming a granularity as small as one second, a four octet timestamp still has a life longer than 130 years.

4.2.2 New PREDECESSOR Path Attribute

To ensure the authenticity of the `AS_PATH` attribute, we augment `UPDATE` messages with a `PREDECESSOR` attribute identifying the AS prior to the destination AS for the current route. We call this AS the predecessor to the destination AS. By including the predecessor information, and a digital signature of this information calculated by the originating speaker (described in Section 4.2.4), the authenticity and integrity of the complete path reported by a speaker to any destination can be established by the speaker's neighbors. Specifically, this can be done by means of a path traversal of the verified predecessor information reported by the route.

The `PREDECESSOR` path attribute includes: the originating AS, the predecessor AS, an IP address of the originating speaker, and a type field. The originating AS must be the same as the AS in the `AGGREGATOR` and the first AS in the first `AS_SEQUENCE` segment of the `AS_PATH` path attribute, if these attributes exist. The predecessor AS must be the same as the second AS in the first `AS_SEQUENCE` of the `AS_PATH` attribute, if it exists. The IP address of the originating speaker must be the same as the IP address in the `AGGREGATOR` attribute, if it exists.

The `TYPE` field can take on the value of either `ADD` or `DELETE`. The `ADD` version of the `PREDECESSOR` attribute is generated by the speaker that originates the `UPDATE` message, which may either be the creator of an unaggregated `UPDATE`, or the last speaker to perform an aggregation of the routing information in the current `UPDATE`. The purpose of the `ADD` type of the `PREDECESSOR` path attribute is to identify the originating BGP speaker whose key is used to digitally sign the `UPDATE`, and to identify the destination and predecessor information in the absence of `AGGREGATOR` and `AS_PATH` attributes (see below regarding transit-only `UPDATES`). The `DELETE` version of the `PREDECESSOR` path attribute serves the purpose of identifying a previously reported predecessor relationship that is no longer valid. Possible reasons for this change include the failure of an inter-AS link, or the termination of a transit traffic agreement. This segment type may be generated by either end of the deleted link; the originating AS field of the `PREDECESSOR` attribute specifies the generating BGP peer.

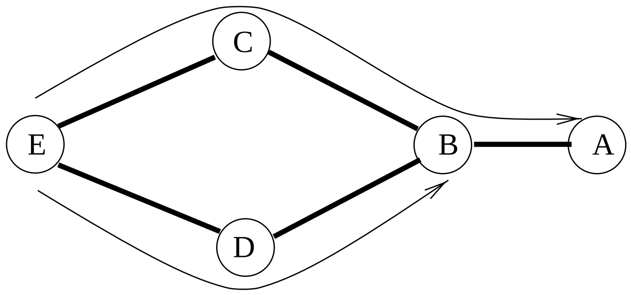

Fig. 2. Need for Multiple Predecessors

The predecessor information is used by each node to maintain a *predecessor table*. The predecessor table is a column vector containing the predecessor to the destination and to each intermediate node on the chosen path to each known destination. The information maintained in the predecessor table is used to verify `AS_PATH` attributes. Before a speaker selects a route, that route's `AS_PATH` attribute should be verified by a walk through the predecessor table. This verification is done by traversing backwards through all `AS_SEQUENCE` segments in the `AS_PATH`, starting with the first AS in the first `AS_SEQUENCE` path segment, confirming that a validated predecessor table entry exists for each predecessor AS in the `AS_PATH`. The timing of this verification is not specified, and is influenced by the expected frequency of invalid `AS_PATH` attributes, expected load, and the performance requirements of the speaker. Options for when to perform this verification include on receipt of the `AS_PATH`, or on selection of the route for use. This check could also be performed on a statistical basis if loads are excessive.

4.2.3 Policy-based Handling of PREDECESSOR Attributes

Smith, Murthy and Garcia-Luna-Aceves [24] have shown how predecessor information alone is adequate to secure intra-domain distance-vector routing protocols. This is possible in these protocols because the paths used to reach destinations downstream from a given node are extensions of the path used to reach the node itself. As a result, at most one update for any given node is passed along by upstream routers, and a chain of $< destination, predecessor >$ pairs uniquely identifies a path to any destination.

However, in inter-domain distance-vector routing protocols in which routing decisions are made based on arbitrary policies, the assumption that paths through a node are extensions of the paths used to reach that node no longer holds. In policy-based routing, the path used to reach a node as a destination does not necessarily have any relation to the path used to reach a node as a relay to a different destination. This is illustrated in Figure 2. In the figure, due to policy-based decisions, a speaker for AS E has chosen path $< C, B, A >$ to reach destination A, and path $< D, B >$ to reach destination B. To validate both paths, the speaker for AS E needs two predecessor table entries for B; one through C and one through D. As a result, more than one update for a given node, each with a different predecessor, can be passed along by upstream speakers. The only restriction on handling updates being that at most one update from a given node is used to reach any one destination. This relaxed restriction results in two additional requirements of the protocol.

First, each speaker upstream from a given predecessor link must maintain a list of destinations that will be accepted over that link. To allow these destination lists to be kept current, speakers must include a list of destinations for which they will handle traffic in updates they generate. Additionally, as speakers learn of new destinations, they must generate new updates (called transit updates) to add these destinations to the list of valid destinations for their predecessor links. A danger of transit updates is that they become hop-by-hop security for each `AS_PATH`. Fortunately, there are a number of factors that mitigate this problem. First, it is not necessary to re-authenticate the same predecessor link/destination pair as paths from the transit AS to the destination change over time. Second, it is not necessary to delete a valid destination for a predecessor link simply because the destination becomes unreachable. Lastly, the `MinASOriginationInterval` variable mentioned earlier causes updates to tend to consolidate, resulting in fewer updates describing more significant changes rather than more updates describing smaller changes. Further work is needed on these issues. Specifically in the areas of how to specify the destinations (AS, IP address prefixes or both), and how to invalidate predecessor link/destination pairs.

Second, a new mechanism must be defined for determining the correct predecessor for a given destination in the path-traversal described previously. With the relaxed restriction on the propagation of updates described above, it is now possible for an upstream speaker to have information describing multiple predecessor links to the same AS that are valid for the same destination. To allow upstream speakers to uniquely determine the correct predecessor link for the path from itself to the destination, each speaker includes information in each update it generates identifying the successor to the downstream AS of the predecessor link it uses for the destination valid on that link. Because each speaker selects only one successor for a destination, the existence of the desired destination in the list of destinations specified for both the predecessor and successor links uniquely identifies the link as part of the current path, and secures the link from both the upstream and downstream ends.

To summarize, the relaxed restriction on the propagation of updates in policy-based routing requires the predecessor information used to secure non-policy-based distance-vector routing protocols to be expanded to include information identifying the destinations traffic is accepted for over that link, and information specifying the successor AS to be used in reaching each of these destinations. Specifically, each update now includes, in addition to the information listed in Section 4.2.2, a list of successor ASs and the destinations for which each AS will handle traffic.

This information is used by upstream speakers to maintain a more complicated distance table composed of, for each neighbor, a two dimensional matrix indexed by destination and originating AS pairs. Each element of this matrix contains a list of quadruplets of: predecessor AS, successor AS, destinations, and distance to the originating AS. An `AS_PATH` is verified by walking backwards through the path verifying that the appropriate overlapping predecessor/originator/successor entries exist, and that the intermediate distances are consistent. Note that in this context the originating AS can act as a relay or a destination (or both). To advertise destinations in its containing AS an originating speaker should include a null successor AS with its AS as a destination in the `PREDECESSOR` field.

4.2.4 UPDATE Digital Signature

To ensure the integrity and authenticity of the unchanging `UPDATE` message information, it is digitally-signed by the originating BGP speaker specified in the `PREDECESSOR` attribute. Without protection, trust of this information requires trust of BGP peers regarding information concerning links not incident on their AS. This is something we explicitly do not do. By including the `PREDECESSOR` attribute information in this signature we protect, in addition to the information in the current `UPDATE`, the full path information contained in the predecessor table described above.

The `UPDATE` message digital signature is stored in the `Marker` field of the header, and is calculated over the following fields: `UPDATE` sequence number, Unfeasible Route Length, Withdrawn Routes, `ORIGIN`, `ATOMIC_AGGREGATE`, `AGGREGATOR`, `PREDECESSOR`, and the NLRI. This definition of the digital signature assumes that these fields are only meaningful as a unit; that a change in one requires the re-computation of them all. If the protocol evolves to where this is not the case, and subsets of these attributes may be updated independently by different BGP speakers, additional sequence numbers and associated digital signatures will be introduced.

4.3 Countermeasure Effectiveness

We now analyze the impact of each countermeasure on the threats identified in Section 3. The message protection countermeasures provide protection against all nodes lacking the necessary cryptographic keys, specifically unauthorized speakers, masquerading speakers, and subverted links. The encryption of BGP messages protects them from fabrication, modification, and disclosure by these classes of intruders. The addition of a sequence number to BGP messages protects them from replay or deletion by these intruders.

Similarly, the UPDATE field protection countermeasures provide protection against compromise by those nodes that do have the cryptographic keys, specifically subverted speakers. The digital signature of the Withdrawn Routes, ORIGIN, AS_PATH, ATOMIC_AGGREGATE, AGGREGATOR, NLRI, and new PREDECESSOR and UPDATE Sequence Number fields protects these fields from fabrication or modification by subverted speakers. The addition of the UPDATE Sequence Number protects against the replay of these fields by a subverted speaker. The addition of the PREDECESSOR path attribute provides a means of validating a link in the internet, which can then be used to validate each link in the AS_PATH attribute.

Referring back to Section 3.4 we see that we have achieved all but a few of our goals. Specifically, a subverted speaker is still able to fabricate destination information, delete routing updates, and disclose routing information. In retrospect, we can see that these goals conflict with our basic assumptions of trust in BGP speakers regarding policy and connectivity information concerning resources for which they are authoritative, and trust to handle routing information confidentially. We believe these vulnerabilities are unavoidable, because they are inherent to the requirements of the protocol.

4.4 Performance Analysis

The cost of these countermeasures is in the space for the new sequence numbers and digital signatures, and the time for computing encryption and digital signatures, and verifying these protections. From the perspective of the actions occurring in a BGP system, the costs are the following:

Message generation and reception: **Space:** A new field is added for the peer-to-peer sequence number. **Time:** The cost of a symmetric key encryption and decryption of each message.

Initiation and reception of UPDATE messages: **Space:** The Marker field is used for the UPDATE message digital-signature. Each UPDATE message includes a new UPDATE sequence number and PREDECESSOR attribute. **Time:** The time to perform the computation and verification of the UPDATE message signature.

Route Selection: **Time:** The time to verify signatures for each link. This cost will only be incurred twice for each used link: once for the ADD and once for the DELETE.

While these costs are not constant per destination (due to the possible need for intermediate nodes to send “transit `PREDECESSOR`” path attributes), they do offer the potential for significantly lower costs than the linear growth in cost with path length of previous solutions. The factors contributing to this improvement were outlined in Section 4.2.3.

5 Related Work

Kumar [12] analyzes the security requirements of network routing protocols, and discusses the general measures needed to secure the distance-vector and link-state routing protocol classes. He identifies two sources of attacks: subverted routers, and subverted links. Since attacks by subverted routers are seen as difficult to detect and of limited value to the intruder, Kumar focuses his attention on securing protocols from attacks by subverted links. For distance-vector protocols, this translates into the modification or replay of routing updates. The specific countermeasures proposed by Kumar are neighbor-to-neighbor digital signature of routing updates, the addition of sequence numbers and timestamps to the updates, and the addition of acknowledgments and retransmissions of routing updates. Kumar and Crowcroft [13] perform a similar analysis of inter-domain protocols, and come to similar conclusions for providing security of distance-vector related routing protocols (they specifically address the path-vector routing protocol IDRP). The one addition they make is to encrypt neighbor-to-neighbor updates.

These results are similar to ours with the exception that we explicitly assume the existence of subverted routers, and provide countermeasures to protect against them. We feel this is necessary, because BGP speakers are potentially vulnerable to attacks from a number of sources, with potentially catastrophic results from success.

Murphy [16] outlines a solution for securing distance-vector protocols that involves including the information used to select a route, signed by the neighbor from which it received it, in the routing update it then signs and transmits to its neighbors. Murphy points out that this requires the validation of a number of nested signatures equal to the number of routers in the path. This results in both update size and validation computation time problems as the size of the network grows. These problems result, fundamentally, from the redundant signing of link information for paths that are supersets of paths used to reach destinations traversed in the longer path. In contrast, we avoid these problems by signing only the component link information, in the form of predecessors, and performing a path traversal to validate full paths. This results in the use of constant space, and significantly reduced computation time.

Smith and Garcia-Luna-Aceves [23,24] have presented security mechanisms for BGP and distance-vector protocols in general. The proposed solutions are similar to those presented here, without as detailed an analysis of the implications of policy-based decisions on the countermeasures.

6 Concluding Remarks

In this paper we analyze the security weaknesses of the BGP protocol, and identify a number of threats involving the deception, disruption, and disclosure of routing message traffic. We propose countermeasures that eliminate or minimize most of these threats. The primary innovation we introduce is the protection of the predecessor information contained in the `AS_PATH` attribute with a digital signature which is used to verify full paths using loop-detection techniques originally designed for path-finding protocols [6]. Using these techniques, we are able to secure full path information in constant space, and avoid the recursive protection mechanisms previously assumed to be necessary.

References

[1] R. Atkinson. Security Architecture for the Internet Protocol. RFC 1825, Aug. 1995.

[2] W. Diffie. Security for the DoD Transmission Control Protocol. *CRYPTO '85*, pages 108–127, 1985.

[3] D. E. Eastlake 3rd and C. W. Kaufman. Domain Name System Security Extensions. Internet draft: draft-ietf-dnssec-secext-10.txt, Aug. 1996.

[4] D. Estrin, J. Postel, and Y. Rekhter. Routing Arbiter Architecture. ftp://ftp.isi.edu/pub/hpcc-papers/ra/ra-arch.ps, June 1994.

[5] D. Estrin, M. Steenstrup, and G. Tsudik. A Protocol for Route Establishment and Packet Forwarding Across Multidomain Internets. *IEEE Trans. on Networking*, 1(1):56–70, Feb. 1993.

[6] J. J. Garcia-Luna-Aceves and S. Murthy. A Path-Finding Algorithm for Loop-Free Routing. *IEEE/ACM Transactions on Networking*, 5(1):148–160, Feb. 1997.

[7] International Standards Organization. *ISO/IEC 10747: Information technology - Telecommunications and information exchange between system - Protocol for exchange of inter-domain routeing information among intermediate systems to support forwarding of ISO 8473 PDUs*, Aug. 1994.

[8] L. Joncheray. A Simple Active Attack Against TCP. *Proc. 5th USENIX UNIX Security Symposium*, pages 7–19, June 1995.

[9] L. Joncheray. Public Key Encryption Support for TCP. Internet Draft: draft-joncheray-encryption-00.txt, May 1995.

[10] S. Kent. Some Thoughts on TCP and Communication Security. MIT Laboratory for Computer Science, Local Network Note No. 6, Apr. 1977.

[11] S. Kent. Comments on "Security Problems in the TCP/IP Protocol Suite". *ACM Computer Commun. Review*, 19(3):10–19, July 1989.

[12] B. Kumar. Integration of Security in Network Routing Protocols. *ACM SIGSAC Review*, 11(2):18–25, Spring 1993.

[13] B. Kumar and J. Crowcroft. Integrating Security in Inter-Domain Routing Protocols. *ACM Computer Commun. Review*, pages 36–51, 1993.

[14] G. Malkin. RIP Version 2: Carrying Additoinal Information. RFC 1723, Nov. 1994.

[15] R. T. Morris. A Weakness in the 4.2BSD Unix TCP/IP Software. Technical Report 117, AT&T Bell Laboratories, Murray Hill, New Jersey 07974, Feb. 1985. ftp://netlib.att.com/netlib/att/cs/cstr/117.ps.Z.

[16] S. L. Murphy. Presentation in Panel on "Security Architecture for the Internet Infrastructure". Symposium on Network and Distributed System Security, Apr. 1995.

[17] S. L. Murphy. Digital Signature Protection of the OSPF Routing Protocol. *Proc. Symposium on Network and Distributed System Security*, 1996. http://bilbo.usy.edu/sndss/sndss96.html.

[18] R. Perlman. *Network Layer Protocols with Byzantine Robustness.* Report MIT/LCS/TR 429, Massachusetts Institute of Technology, Oct. 1988.

[19] P. Rangan. Trust Requirements and Performance of a Fast Subtransport-Level Protocol for Secure Communication. *IEEE Transactions on Software Engineering*, 19(2):181–186, 1993.

[20] Y. Rekhter. Inter-domain Routing Protocol (IDRP). *Internetworking: Research and Experience*, 4:61–80, 1993.

[21] Y. Rekhter and T. Li. A Border Gateway Protocol 4 (BGP-4). RFC 1771, Mar. 1995.

[22] J. H. Saltzer, D. P. Reed, and D. D. Clark. End-to-End Arguments in System Design. *ACM Trans. on Computer Systems*, 2(4):277–288, Nov. 1984.

[23] B. R. Smith and J. J. Garcia-Luna-Aceves. Securing the Border Gateway Routing Protocol. *Proc. IEEE Global Internet '96*, Nov. 1996.

[24] B. R. Smith, S. Murthy, and J. J. Garcia-Luna-Aceves. Securing Distance-Vector Routing Protocols. *Proc. ISOC Symposium on Network and Distributed System Security*, pages 85–92, Feb. 1997.

[25] J. J. Tardo. Standardizing Cryptographic Services at OSI Higher Layers. *IEEE Communications Magazine*, 23(7):25–29, July 1985.

[26] V. L. Voydock and S. T. Kent. Security Mechanisms in High Level Network Protocols. *ACM Computing Surveys*, 15(2):135–171, June 1983.