Digital Signature Protection of the OSPF Routing Protocol *

S. L. Murphy
Trusted Information Systems
Glenwood, MD 21738

M. R. Badger
Trusted Information Systems
Glenwood, MD 21738

Abstract

The routing protocols used to disseminate routing information throughout the Internet are not protected from intruders or faulty router participants. This paper reports on work in progress to protect the OSPF routing protocol through the use of cryptography, specifically, digital signatures. The routing information is signed with an asymmetric cryptographic algorithm, allowing each router recipient to check the source and integrity of the information. This paper discusses the fundamental issues in security of routing protocols, reviews the basics of OSPF operation, describes the proposed design and discusses remaining vulnerabilities.

1 Introduction

Routing protocols distribute information regarding the topology of the network among the routers of the network. The information each router receives serves as the basis for forwarding packets from their source to their destination. Without accurate routing information, packet transmission through the network is at best inefficient and at worst may fail completely. Despite the criticality of this part of the infrastructure, routing protocols are not well protected against deliberate or accidental propagation of incorrect routing information. The routing protocols, as most of the infrastructure protocols, were designed in a more benign era in the Internet. They function with an implicit trust both in their peers and in the information they receive. Neither trust is suited to the current Internet environment.

The routing protocols that operate in the Internet are all subject to certain sorts of attack. Because the routing protocols all function cooperatively based on the information they receive from their peer routers, they are all threatened by the possibility that routing information might be modified, that old routing information might be replayed, or that new bogus routing information might be generated and inserted into the communication. This paper addresses protection from threats related to the routing information; it does not address protection from other sorts of threats such as overwhelming the routers by generating spurious excess traffic. An implementation of this design is underway at TIS, with an expected completion time in the second quarter of 1997.

When routing is attacked, the effect on the network can vary. Links in the network can become more congested because, for example, a link is incorrectly advertised as the best route to many networks. The overall load on the network can increase. This could be caused by routes being computed that contain either loops or paths that are longer than necessary, increasing the time that data remains in the network. The load on network applications and delays in the network itself can increase. This could be caused by data never reaching its intended destination, making retransmissions necessary. Incorrect routing information might also make it appear that portions of the network are unreachable when usable routes do exist. If the routing protocol contains provisions to purge information from the distributed routing database, incorrect routing information could possibly decimate the routing databases everywhere in the network.

There are two possible sources of incorrect routing information. One is an external intruder who manages to gain access to the communications between two routers. Such an intruder can modify or delete the routing information packets the two routers exchange, or pose as one of the two to insert bogus routing packets. It might modify the source or destination of data as it passes by. An even greater danger comes from a router, a participant in the protocol, that is not behaving correctly. Such a router may transmit any amount of faulty information or forward data in an arbitrary manner. Because routers pass along the information they receive to other routers in the network, the effect of one intruder or one misbehaving router can indirectly affect a large area of the network.

Clearly, the routing process must be protected for the Internet as a whole to function properly. The insertion of incorrect routing information into the distributed exchange of information can be countered by protecting the authenticity and integrity of routing information. Presently, many different Internet routing protocols (RIP[5], OSPF[6], ISIS[2], IDRP[3]) reserve fields in the packet format for authentication use. However, the strongest authentication mechanism defined for these fields is some protocols (RIP, OSPF, ISIS) is clear-text passwords. IDRP, an inter-autonomous system protocol, defines an authentication mechanism consisting of a checksum of the data and a shared secret. The sniffer attacks of the last few years demonstrate that clear-text passwords are not

*This work supported by ARPA contract DABT-94-C-0001, Internet Infrastructure Protection

strong enough protection. Cryptographic protection of source authenticity and integrity provides stronger protection and can counter both sources of attack.

Routing protocols use different algorithms for gathering and disseminating information about their area. The algorithms are characterized as using one of two techniques, distance vector or link state routing. In the distance vector technique, each router gathers information from all its neighbors as to the topology of the network, presented as their best routes to all possible destinations. The router uses this set of information to decide its own best route to all possible destinations, and transmits this summary information to all its neighbors. In the link state technique, each router gathers information on the state of its links to its neighbors and sends this information to the entire network. As the routers are not fully connected, the information is sent to the entire network by a process called flooding, whereby by each router forwards every link state packet it receives, without change. One consequence of the link state routing flooding technique is that each router eventually receives a complete topology of the network. The difference between the two techniques can be summarized as sending information about the whole network to one's neighbors vs. sending information about one's neighbors to the whole network.

The amount of protection that can be derived from using cryptographic protection differs for the two techniques. In both cases, cryptography can be used to protect the information that is exchanged between neighboring routers from external intruders. For example, the two routers can share a secret used in a symmetric cryptographic algorithm to protect the information. Because the external intruder does not know the shared secret, it cannot believably modify information or produce acceptable bogus information. (Note: Some routing protocol messages exchanged between neighboring routers do not carry routing information but are still crucial to the operation of the protocol. These messages can be protected from external intruders by the same cryptographic techniques.) Protecting against internal sources of incorrect routing information is a more difficult problem. Preventing a participant in the routing protocol from generating bogus information or modifying information it propagates requires protection of the authenticity of the ultimate source of the information, rather than just the immediate source, as well as protection of the integrity.

For distance vector techniques, it is difficult to use cryptography to protect the authenticity of the ultimate source of the information. Because distance vector techniques summarize the information they receive before transmitting their own information to the net, they obscure all trace of the ultimate source of the information. Cryptographic protection is of little use when the source to be authenticated can not be determined.

Because link state techniques flood the same information to everyone, the source of the information can still be determined and protected. In this case, cryptographic protection of the information provides the source authenticity and integrity that is needed. Because each router would wish to verify the single source of the information and each router's information reaches each other router, using symmetric cryptography would require a $O(N^2)$ set of secret keys, one set for each pair of routers. Using asymmetric cryptographic, by applying a digital signature to the routing information, is a better choice. The digital signature would provide protection of the source authenticity and integrity, as well as detection of the source of any incorrect information.

2 Related work

The idea of signing routing information is not new. Foremost, of course, there is the design that Radia Perlman reported in her thesis [7] and in her book [8] for signing link state information and for distribution of the public keys used in the signing. IDPR [10] also recommends the use of public key based signatures of link state information. Kumar and Crowcroft [4] discuss the use of secret and public key authentication of inter-domain routing protocols. Finn [1] discusses the use of secret and public key authentication of several different routing protocols. The design reported here uses the same basic concepts as that reported in [7] and [10]. It should be noted that [7] also presents techniques for protecting the forwarding of data packets, a topic that is not considered here, as we consider it a separate problem from protecting the OSPF routing protocol.

3 OSPF basics

OSPF is a link state routing protocol used among routers that all belong to one autonomous system (in ISO terminology, a routing domain). OSPF defines an aggregate of routers in the autonomous system called an *area*. OSPF establishes a two level hierarchy among these areas. with the top level defined as the *backbone* area and the second level consisting of many areas attached to the backbone. Routers that belong to more than one area by definition belong to the backbone and are known as *Area Border Routers (ABR)*. Various of the routers, within an area or within the backbone, may be connected to points outside the autonomous system and are known as *Autonomous System Boundary Routers (ASBR)*. Figure 1 shows an OSPF autonomous system and the terms used in OSPF systems.

Within each area and within the backbone, OSPF operates as a pure link state protocol with information advertised by each router eventually being flooded to every router in the area. Between the areas and the backbone, OSPF operates more like a distance vector algorithm. An ABR advertises to the backbone a summary of all the networks it can reach in its attached area. The ABR advertises to its attached area a summary of all the networks it can reach through the backbone. Finally, an ASBR advertises destinations it can reach that are outside the autonomous system.

All this information is used to determine the shortest path to any desired destination. Routers within an area know from the ABR advertisements which ABR would give the shortest path to any desired destination

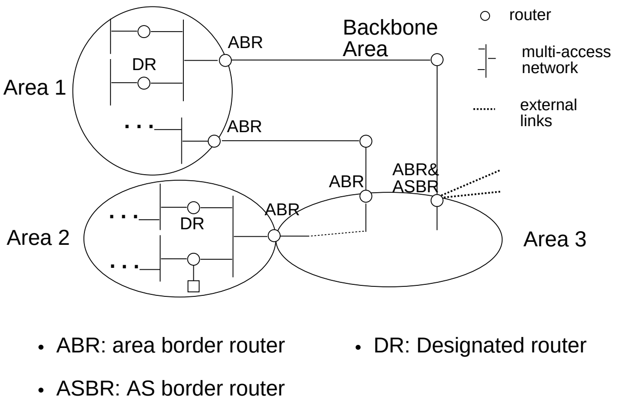

Figure 1: OSPF Terminology

located within the autonomous system but outside the area. Similarly, routers know from the ASBR advertisements which ASBR would give the shortest path to a destination outside the autonomous system.

OSPF defines five link state advertisement (LSA) types which contain the routing information. The routing information includes not only the address (of an interface, network, router, etc.) being advertised but also a metric for the link to that address.

- **Type 1:** Each router advertises a Router Links Advertisement to its area, describing the state of each of the router's interfaces in the area.
- **Type 2:** Each multi-access network selects a Designated Router through which to communicate so as to reduce traffic on the network. The Designated Router advertises the list of routers connected to the network in a Network Links Advertisement.
- **Type 3:** Each ABR advertises a Summary Link Advertisement to each of its attached areas, describing routes to networks outside that area (but within the autonomous system).
- **Type 4:** An ABR advertises a Summary Link Advertisement to each of its attached areas, describing routes to ASBR's outside that area.
- **Type 5:** Each ASBR advertises many AS External Link Advertisements, each describing a route to a destination in another autonomous system.

Each router discovers its router neighbors via a Hello Protocol and exchanges its routing information database (the LSA's it has generated or received) with its neighbors. This is called "forming an adjacency" and "synchronizing databases". The routing databases local to each of the routers together form a distributed routing database. An LSA is flooded throughout the network, both periodically and whenever the information it carries changes. A router may receive multiple copies of the same LSA; only the most recent, based on the sequence number and age fields, is stored and propagated. The sequence number is created by the originating router and incremented with each periodic propagation of the LSA. Each LSA is aged at the originating router and all other routers through which it travels. When an LSA reaches a set maximum age (MaxAge) at a router, it is purged from the router's database and flooded to the network to purge the distributed routing database.

The OSPF protocol is capable of providing type of service (TOS) based routing. A TOS capable router will produce LSA's that list a metric for a destination for each of several types of service. It is not required that all routers be "TOS-capable" but those that are will compute a different SPF tree for each type of service, using the TOS metrics in the LSA's.

4 Using digital signatures in OSPF

The basic idea of this design is to add digital signatures to OSPF LSA data, and to recommend the use of a neighbor-to-neighbor authentication algorithm (like keyed MD5) to protect all protocol exchanges. Link state information will be signed by the originator of that information and the signature will stay with the data in its travels via OSPF flooding. This will provide end-to-end integrity and authentication for LSA data. Routers providing digital signatures will be "authenticated routers", and can be mixed with non-authenticated routers. An application will be able to specify authenticated routing as an IP TOS, and have packets forwarded accordingly.

A digital signature attached to an LSA by the source router provides assurance that the data really does come from the advertising router. It will provide assurance that the data has not been modified in transit. In the case where incorrect routing data is distributed by a faulty router, the signature provides a way to trace the problem to its source.

Digital signatures for OSPF LSA's can be implemented with the following major elements:

1. Support for a digital signature algorithm in authenticated routers.
2. Support for a signed version of all routing information LSA's (see Section 4.1).
3. Support for a new LSA: Router Public Key LSA (see Section 4.2).
4. Addition of an IP TOS for authenticated routing.
5. Support for TOS routing and forwarding in authenticated routers (see Section 4.3).
6. Configuration or supplied data:
   - Public Key of Trusted Entity
   - At least one set of the following (can be one or one per attached area):
     - Router Private Key
     - Router Public Key, Id and Role certified by a Trusted Entity

- Signature Algorithm Information (type, parameters)
- Hash Algorithm Information (type, parameters)
- Environment flag (authenticated, non-authenticated, mixed)

4.1 Authenticated routing information

Authenticated OSPF routers perform all the normal functions of a standard OSPF router. In addition to the standard functionality, an authenticated OSPF router generates signed routing information LSA's, sends a new key information LSA, manages key and signature algorithm information, and verifies signatures received. An authenticated OSPF router must support TOS routing, specifically for TOS=authenticated routing, as explained in Section 4.3.

Content: The signed routing information LSA is called an authenticated LSA. The content of an authenticated LSA is:

- **Normal LSA Header:** fields identifying the specific link, the instance of the LSA, the capabilities of the router, and the length of the LSA.
- **Signature Information:** information necessary for the proper interpretation of the signature
- **Link State Data:** (Variable according the the LSA type)
- **Signature:** The signature of the LSA.

The fields that identify the specific link are the LSA type, an identification of the link (Link ID), and an identification of the advertising router (Advertising Router ID). The fields that identify a specific instance of an LSA are the sequence number, checksum, and age. A currently defined capability is the ability to process TOS information. We will discuss using the TOS capability in Section 4.3.

The signature information includes an identification of the signature algorithm (RSA and DSA are the currently defined choices) and of the hash algorithm (MD5 and SHA are the currently defined choices), the key size, and a key identifier. The identification of the signature and hash algorithms and the key size provide flexibility for use in areas of an autonomous system that may have access to different signature algorithms. This also provides extensibility if other asymmetric cryptographic algorithms become available or a key size is determined to provide insufficient security. The key identifier is used to provide an easy identification of which one of multiple keys was used to produce the signature. This can ease transfer to a new key.

Obviously, the LSA Data and the Advertising Router ID (the source of the information) must be protected by the signature. Modification of any of the other fields may also adversely affect the proper behavior of the protocol. Modification of the fields that identify the link could lead to improper interpretation or use of the LSA data. For example, modification of the link ID would make the LSA's metric apply to some other link associated with that router. Modification of the fields that identify the specific instance could lead to abandoning current routing information in favor of outdated routing information. For example, modification of the sequence number could allow outdated information to be replayed and replace the current information. Modification of the length field could lead to information being ignored or misinterpreted. Modification of the signature information could result in failure of the verification of the signature. Therefore, every part of the LSA should be protected by the signature. However, the age field is modified by all routers which propagate the LSA. Therefore, the signature covers all parts of the LSA but the age field, except in circumstances explained in Section 4.1.

Processing: When the router receives a routing information LSA, the signature should be verified using the current public key of the advertising router. Distribution of the public keys of the routers is discussed in Section 4.2. If there is no key stored for the advertising router, then the signed LSA must be discarded. If the signature verification fails, the LSA must be discarded. If the signature verifies, then the signed LSA is stored for use in the routing calculations. The TOS = Authenticated Routing metrics in the signed LSA will be used in the construction of an SPF tree for this TOS, as explained in Section 4.3. The computed routes will be put into the OSPF routing table.

Processing MaxAge: The age field in the OSPF LSA header is used to keep track of how long a given LSA has been in the system. The originating router and all routers that propagate an LSA will increment the age of the LSA. The age, along with the sequence number and the checksum, allows a router to determine which LSA's are more recent. When the age of an LSA stored in any one router's database reaches MaxAge, the router purges the LSA from its own database, and it floods the MaxAge LSA through the network.

As the maximum age (MaxAge) is deemed more "recent" than any other age, it will be accepted at every router it reaches and cause the corresponding LSA to be purged from each local database. This mechanism is used to purge outdated information from the distributed routing database. When a router fails, eventually its LSA's will reach MaxAge in some router in the area and will be purged. If a router wants to purge an outdated LSA of its own from the system, it can prematurely set the age to MaxAge and flood the LSA.

This element of the protocol is difficult to protect using digital signatures. The age field cannot generally be included in the signature, because it must be updated by routers other than the originating router. For the same reason, the age field is not included in the checksum computation. The age field should be protected, because if a faulty router aged another router's LSA's (modified the value of the age field to MaxAge), it would disrupt routing through that router.

To protect the age field, the signature should include the age field when, and only when, the age field value is MaxAge. Verification of the signature on a signed LSA should include the age field when, and only when, the age field value is MaxAge.

Note that a received LSA with the age field set to MaxAge could have been sent by the originating router or by any other router which had aged the LSA to MaxAge in its local routing database.

For authenticated routers, only the MaxAge LSA sent by the originating router would be recognized as valid, as only the originating router can generate a signature covering the age field. A signed LSA with age MaxAge flooded by a router that is not the LSA's originating router will be ignored by all authenticated routers. In this way, the originating authenticated router can purge an LSA from the distributed database by prematurely or normally aging it, but prevent other routers from prematurely aging its LSA.

However, a non-originating router's flooding of signed LSA's that have normally reached MaxAge in its local database will be also be ignored. If an authenticated router goes down, its signed LSA's must be aged out by each remaining router's local database individually. This will slow database convergence when an authenticated router goes down, but the databases will still converge, and a fairly obvious security hole will be closed.

4.2 Key management and distribution

This design relies on Public Key cryptography. The common examples are RSA and DSA, but a specific algorithm is not mandated by this design. There are some good books on the subject [6], but the following discusses some of the important issues.

Each router has a private key that is secret and a public key that everyone may know. A signature can be generated with the private key, and verified using the public key. The verification assures that the data signed has not been altered in transit (integrity), and that it was signed by the router having the correct private key (source authentication). The assurance of source authenticity and integrity relies on two things: first, that the private key is known only to one router, and second, that the public key is reliably known to belong to that one router. Protection of the private key is a matter for local implementation.

Clearly, then, the distribution of the public key and the binding between that key and the router to which it belongs, is very important to the security of this system. It is also important that the paradigm of the OSPF protocols be maintained in the key distribution, i.e., that a router need not know of the name, location, or even existence of every other router in the network. If key management and distribution mechanisms that are independent of the routing protocol exist, then they could be used to provide the set of public keys to each router external to the routing protocol. The key distribution mechanism suggested here employs the flooding mechanism already defined in OSPF.

The design assumes that there is a Trusted Entity somewhere in the autonomous system that has a secret private key, and a public key that all routers are provided as part of their configuration data. It is the responsibility of this Trusted Entity to verify, according to autonomous system policy, the binding between a Router ID and a public key. It is not necessary that this Trusted Entity be online or accessible electronically. Each router must be configured with it's own pair of keys (public and private), and with the public key of the Trusted Entity. It must obtain from the Trusted Entity a copy of the Trusted Entity's certification of the binding between the router's Router ID and its public key. The mechanism by which it obtains this certification is not described here, but could be an e-mail message, a phone call, a letter, a smart card, etc.

To certify the binding between a router's Router ID and its public key, the Trusted Entity signs this information, as well as the identification of the router's role in the area (internal router, ABR or ASBR) and the key identifier and expiration time.

An authenticated router sends its certified public key in a Router Public Key LSA via OSPF flooding. The public key LSA includes:

- **Normal LSA Header:** This includes all the fields described before for an LSA. The Link ID field contains the key identifier.
- **Signature Information:** This includes the signature and hash algorithm and key size information as described before for an LSA. The algorithm and key sizes are assumed to be the same for the router and the Trusted Entity. It also includes the length of the certification field and the length of the signature field.
- **Certified Information:** The information that the Trusted Entity has certified: the router's Router ID (which must be the same as in the LSA header), the router's role(internal router, ABR or ASBR), the Router's public key, key identifier and the key expiration time.
- **Certification:** This is a signature produced by the Trusted Entity of the certified information.
- **Signature:** The router's signature of the LSA, excluding the age field as before.

All authenticated routers receiving this LSA verify the certification using the Trusted Entity's public key, which, again, all routers must be provided. They store the advertising router's public key and it to use in verifying the advertising router's signatures on LSA's, including the signature on this LSA. It is required that the key identifiers be strictly increasing. If more than one Router Public Key is received by a router, only the one with the greatest key identifier should be used to verify incoming LSA's. Note that this gives a quick method of discarding incoming LSA's signed with an old key.

Periodically, keys will have to be changed, and the new router public key will have to be certified by the Trusted Entity. A router could generate its own new key pair, or could receive them via a key distribution scheme. If the router generates its own key pair, there is no need for it to communicate the private key to the Trusted Entity; only the public key must be certified.

Two situations must be accounted for in any key distribution mechanism: key rollover and de-certification of a compromised private key. It is to suit these two purposes that key identifiers are required to be strictly increasing. When a new key is being propagated through the network, it supersedes all other keys because of its greater value. If a key has been compromised, the originating router can purge the corresponding Router Public Key LSA from the distributed database by prematurely aging it to MaxAge. If this was the only mechanism used to de-certify a compromised key, then the thief could then introduce a new Router Public Key LSA for the victim router containing the stolen private key and the previously published certification, signing it with the stolen private key. The use of increasing key identifiers allows a router to determine which of two received keys is the most recent. Including the key identifier in the certification protects that field from being used by the thief in constructing a new Router Public Key LSA with a new key identifier.

When forming an adjacency or synchronizing databases, the Router Public Key LSA's should be be sent and requested before other LSA's. This provides that the keys are available to verify signed LSA's when they arrive. The Router Public Key LSA is sent at intervals like all other LSA's, and it is sent immediately if a router obtains a new key to distribute.

Comparison to certificates and certificate revocation lists: The Internet community is familiar with X.509 certificates that bind subjects to keys, with Certificate Authorities (CA's) that issue and distribute certificates, and with Certificate Revocation Lists, also distributed by CA's, that publish notification of revoked certificates. The mechanism we propose is similar in effect.

The X.509 certificates contain the subject's identification, the issuer's identification, the validity period, a serial number and the key, all signed by the issuer. The validity period is used to determine when the key should be used. This capability is needed when the certificate and the material it protects arrive asynchronously and when the need to validate persists past the time of active use of the key. The Router Public Key LSA contains the subject identification, serial number ("key identifier") and key, but not the issuer's identification or validity period. The issuer's identification is not needed, as we have designed only one. Because the key arrives synchronously with the information it is meant to protect and is not needed to validate old information past its period of active use, we do not believe a validity period was needed. Also, employing a validity period would introduce the need for network time synchronization, a requirement we did not wish to introduce.

The CA and the Trusted Entity we propose both certify the binding between subject and key. However, our Trusted Entity is not presumed to be online and so does not play an active part in key distribution. If it were online, it could distribute the keys by flooding each router's Router Public Key LSA itself, just as a CA can actively distribute certificates. However, this would lead to asynchronous arrival of the keys and the signed information that uses the keys. This is tolerable in common uses of certificates (e.g., protecting electronic mail), but not in protecting routing information.

The CRL revokes certificates by explicitly listing them. Our Router Public Key LSA revokes old keys implicitly, by superseding the old key identifier. Note that the Trusted Entity must have the cooperation of a router (usually the router to which the key belongs) to distribute a new Router Public Key LSA and thereby revoke the old. A CA need not, because it actively distributes the revocation (the CRL) itself. This should not be a problem if a key is revoked because it is old and suspect or known to be compromised. In the case where the router itself is misbehaving, the Trusted Entity would need the cooperation of some other router in the network to distribute a new Router Public Key LSA and thereby revoke the old.

4.3 Using authenticated routing information

It is likely, particularly in the period when authentication is first being implemented in a network, that not all routers in an autonomous system will be capable of dealing with digital signatures on routing information. It might be possible to isolate the non-authenticated routers into their own separate areas. Initially, this might result in an awkward network configuration. Therefore, we suggest features in the design that allow OSPF with digital signatures to operate in an area containing both authenticated and non-authenticated routers.

Authenticated OSPF routers can send out signed and unsigned versions of each LSA. This requires that there be a new LSA type defined for each existing type. The unsigned version, the standard OSPF V2 [6] LSA, provides backward compatibility with non-authenticated routers. The signed LSA's contain the same routing information, and are flooded, aged, and used in routing calculations like unsigned LSA's. Each router is configured to know whether to send signed LSA's, unsigned LSA's, or both. If all routers in an AS are authenticated then only signed LSA's need to be sent. If authentication is not available, then only unsigned LSA's are sent. If authenticated and non-authenticated routers are mixed in an area, then signed and unsigned versions of the same LSA's must be sent out. This design works best if all the routers in an AS are authenticated, but it can still be useful in a mixed environment.

The type of the signed LSA's will indicate the presence of a signature. Standard OSPF routers will discard the unfamiliar LSA's containing key and signature data, so, in a mixed environment there will be "islands" of authenticated routers that receive each other's authenticated LSA's, but do not receive the authenticated LSA's from other "islands" of authenticated routers. In order for the computation of routes to be consistent across the network, the authenticated routers must compute the same routes as the unauthenticated routers. To use the authenticated routing information, they must compute routes specifically with that information. We accomplish this by providing for an Authenticated Routing type of service (TOS). The signed LSA's will include metrics for TOS = Authenticated Routing. Routes will be computed in the SPF tree computation for TOS 0 (normal routing) with the unsigned data, for compatibility with non-authenticated routers. A separate SPF tree will be computed with the TOS = Authenticated Routing metrics, i.e., with the authenticated routing information. Both sets of routes will be stored in the routing table. To take advantage of the authenticated routes, an application must set the requested TOS in the IP header to Authenticated Routing, and the IP forwarding code must use the TOS routes from the routing table. IP packets not requesting this special type of service will be routed by the TOS 0, non-authenticated routes.

Another method of employing the authenticated routing information would be for the source host to compute or retrieve an authenticated route for use with the source route feature of IP.

In either case, authenticated routes will only be used on an "island" of authenticated routers that includes the destination host. The signed LSA's cannot propagate past the border of the "island", so authenticated information is not available even on another "island" of authenticated routers. The source route computation method would be of greater usefulness if the authenticated information was available outside each "island". In that case, the source could compute routes that passed through authenticated islands as much as possible.

5 Remaining vulnerabilities

Note that with this mechanism, one router can still distribute incorrect data in the information for which it itself is responsible. Consequently, an autonomous system employing digital signatures with this mechanism will not be completely invulnerable to routing disruptions from a single router. For example, the ABR's and ASBR's will still be able to inject incorrect routing information. Also, any single internal router can be incorrect in the routing information that it originates about its own links.

5.1 Area Border Routers

Even with the design proposed here, the ABR's can inject incorrect routing information into their attached areas about the backbone and the other areas in Summary LSA's (Type 3 and 4). They can also inject incorrect routing information into the backbone about their attached area.

Because all the ABR's in one area work from the same database of LSA's received in their common area, it would be possible for the ABR's to corroborate each other. Any ABR for an area could double check the Type 3 and 4 LSA's received over the backbone from other ABR's from the area, and could double check the Type 3 and 4 LSA's flooded through the area from the other ABR's. The other routers in the area or backbone would have to be warned of any check failure. The warning would be a signed message from the ABR detecting the failure, flooded in the usual mechanism.

Another possible solution would be for the ABR's in an area to originate multiple sets of Type 3 and Type 4 LSA's. One set would be generated for itself and would contain its own routing information. One set would be generated for each of the other ABR's in the area, containing the information each of them *should* originate. Each router in the area or backbone could then determine for itself whether the ABR's agreed. This distribution of information but coordination of processing is in keeping with the paradigm of link state protocols, where information and processing is duplicated in each router.

Both alternatives mean much additional processing and additional traffic, over and above the additional processing required for signature generation and verification. Because the vulnerability is isolated to a few points in each area, because the source of incorrect information is detectable (in those situations where the incorrect information is spotted) and because the protection is costly, we have not added this protection to this design.

5.2 Internal Routers

The internal routers can be incorrect about information they themselves originate.

A router could announce an incorrect metric for a valid link. There is no way to guard against this, but the damage would be small and localized even if the router is announcing that the link is up when it is down or vice versa.

A router could announce a connection that does not in fact exist. If a router announces a non-existent connection to a transit network, the OSPF Dijkstra computation will not consider the connection without a similar announcement from another router at the other "end". Therefore, no damage would result without the cooperation of another router (above network impact to transmit and store the incorrect information). A router could also announce a connection to a stub network or a host route that does not exist. In this case, the Dijkstra computation can not perform the same check for a similar announcement from the other "end", because no other end exists. This is a vulnerability.

A faulty router announcing a nonexistent connection to a stub network or host would result in the faulty router receiving IP packets bound for that network or host. Unless the faulty router then forwarded the packets to the correct destination by source routing, the failure of packet delivery would expose the incorrect routing. To exploit the vulnerability deliberately, the faulty router would have to be able to handle and pass on the received traffic for the incorrectly announced destination. Furthermore, if the incorrect routing were discovered, the signatures on the routing information would identify the faulty router as the source of the incorrect information.

Even so, there may be reason to protect against one faulty router disrupting routing by announcing these unsubstantiated connections. In the worst case, a faulty router could announce nonexistent host routes to a large number of addresses in the area or autonomous system. (Note that announcing a large number of incorrect routes would raise the probability that the incorrect routing would be detected, leading to detection of the faulty router as the source of the error.) To guard against this vulnerability would require that there be some corroboration of the connections a router could announce. One way to do this would be to have an authority in the autonomous system produce signed authorizations of the networks that a router would be allowed to announce. This would mean that before a router could be part of the OSPF exchanges it would need to communicate, either on-line or off-line, with the authority. When an existing connection disappeared permanently or a new connection came into being, a new authorization from the authority would be needed. As the existence of connections a router has with networks, hosts, and other routers is not as dynamic as the state of those connections, this might not be too great a hardship for network management for one router.

This announced networks authorization could be made part of the Router Public Key LSA and therefore distributed as part of the normal OSPF flooding mechanism, by including in the Router Public Key LSA the number of allowed network ranges and, for each allowed network, the network address and mask. This information would have to be signed by the authority. If the authority and the Trusted Entity were the same, then this information could be included in the certified information field and be covered by the Trusted Entity's certification signature.

(Note that the internal router vulnerability applies only to one-sided connections, but the protection could be applied to all connections a router may announce.)

As the connections that would require authorization should not change frequently, distributing the authorization with the speed of the OSPF flooding mechanism may be unnecessary. Some other authorization distribution mechanism could be employed.

5.3 Autonomous System Boundary Routers

The ASBR's can produce incorrect routing information in the external routes information they originate. There is no way to double check or corroborate this information, as there is with ABR's. No authority within an autonomous system exists to authorize the networks an ASBR could announce, as is the case for the internal networks an internal router could announce. Consequently, the ASBR's remain a unprotected vulnerability. With this in mind, special care should be taken to protect the ASBR's with other means.

5.4 MaxAge

Despite the mechanism described in Section 4.1, there is still a vulnerability arising from the use of the MaxAge age to purge LSA's from the database. Because each router can modify the age field of an LSA, only the distinguished value MaxAge can be protected. A faulty router might modify the age field so that it was not exactly MaxAge, but close to MaxAge. In the normal aging process, the age would become MaxAge prematurely, causing the LSA to be purged from the distributed database prematurely.

The effect of this vulnerability is limited. OSPF uses the age field in deciding which of two LSA's is the more recent. It will neither store nor propagate the older LSA. An LSA with an age close to MaxAge would therefore have little chance of being accepted and propagated. However, OSPF will judge two ages to be the same if they are within a window called MaxAgeDiff. An LSA whose age field has been modified to be close to MaxAge will be accepted if the true age of the LSA is within MaxAgeDiff of MaxAge. That is, the bogus LSA could replace the true LSA and cause premature purging, but only if the true LSA is already itself within MaxAgeDiff of being purged.

One might think that a misbehaving router could accelerate the aging rate of any LSA by advancing the age field several times, each time by an amount within the acceptable window. This would not be effective, however. Any time that the age of an LSA exceeds the true age by more than the acceptable window, it will be rejected (and actively replaced) in favor of the LSA with the true age. On a graph of age vs. time, with two lines representing the normal aging of LSA's and the premature aging of LSA's, the line representing the premature aging has a steeper slope than the line representing the normal aging. Any time that the two lines are separated by more than the acceptable window, the true aged LSA will replace the prematurely aged LSA. So the premature age can not exceed the true age by more than the acceptable window.

The extent of database corruption would depend on how much of the network was "immunized" with the true aged LSA when the prematurely aged LSA is produced by the misbehaving router. That would depend on where the bad router was situated in the network with respect to the originating router. No damage would result from the database corruption unless the premature age got within the window of the MAXAGE value, at which point normal aging could cause the age to reach MAXAGE and the LSA would begin to be purged, somewhat before its time, from databases.

The vulnerability, then, is only as large as the MaxAgeDiff window. MaxAgeDiff is currently defined to be one quarter of the MaxAge and one half of the normal refresh interval. This would in normal circumstances mean that an LSA would not age to within MaxAgeDiff of the MaxAge before it was replaced by a new instance.

6 Performance concerns

Some features of the OSPF protocol, e.g., the flooding mechanism, are specifically designed to permit routing information to be communicated throughout the network as quickly as possible. It is undeniable that introducing signature verifications in this process will affect performance. Public key cryptography, in particular, could slow performance.

Routers will be using cryptography in two ways:

- signing their own LSA's - periodically at the refresh interval and when changes to their routing state occur, and
- verifying the LSA's they receive.

Because the number of the router's own LSA's that must be signed is small compared to the number of verifications, because signatures are produced at relatively infrequent intervals (baring changes in the routing state), and because LSA signatures can be computed at any time during the interval, we do not believe that signing will significantly damper performance. It will be the need to verify incoming LSA's that will be the stressing factor.

The time it takes to verify a signature can vary widely, depending on algorithm, software implementation, key length, platform, etc. Results for RSA have been seen as low as 270 microseconds for a Sparc 20 with a 512 bit key size using the GNU MP library.

The number of verifications needed depends on the size of the topology database. Networks of 40-50 routers are common, with some networks as large as 1000 routers. The size of one router's database will depend on the number of announced subnets as well as the configuration of the autonomous system into areas. But an even larger affect will come from the external routes announced by the ASBR's. It is not unusual to see databases with tens of thousands of entries due to external routes. The problem presented by the external routes is exacerbated by the OSPF feature of announcing each external route in a separate LSA, which means that one verification is needed for each external route, instead of one (or a few) per ASBR.

Several methods could be used to reduce the impact of the required verifications. First, the verification effort could be offloaded from the processor performing the normal router functions. One offload method would be to use PCMCIA cards (e.g., from National Semiconductor or Telequip or the government's Fortezza card). PCMCIA cards have an added benefit in that they provide increased security for the crucial private key. Another offload method would be to employ a separate processor in a multiprocessor router architecture. Multi-processor architectures are available among many router vendors already, as mechanisms to separate routing computations from forwarding functions. Second, the verification effort could be lessened if the OSPF protocol was changed so that the external routes from one ASBR were packaged in larger aggregates reducing the number of verifications needed. Third, the verifications could be scheduled periodically or on demand, instead of in real-time upon arrival. This last option will reduce the chances of preventing damage due to misbehavior but will retain the ability to detect the router that is misbehaving.

7 Conclusion

It is possible with digital signature techniques to protect the source authenticity and integrity of the routing information distributed by the OSPF routing protocol. This paper discusses a design in which digital signatures are added to LSA data. This protects both against external intruders and against internal sources of error, i.e., participants in the protocol that misbehave. A Router Public Key LSA is introduced which distributes the router's public key to other routers in the area via the usual OSPF flooding mechanism. A Trusted Entity provides a certification of the binding between the router ID and the public key; the certification is included in the new LSA.

Even with this protection of the routing information, there are attacks on the routing information to which the protocol is still vulnerable. These remaining vulnerabilities, the attendant risks, and possible solutions are discussed.

References

[1] Gregory G. Finn. Reducing the vulnerability of dynamic computer networks. Technical report, University of Southern California Information Sciences Institute, Marina del Rey, California, June 1988.

[2] Joint Technical Committee ISO/IEC JTC 1 *Information Technology.* Information Technology - Telecommunications and Information Exchange between Systems - Intermediate System to Intermediate System Intra-domain Routeing Information Exchange Protocol for Use in Conjunction with the Connectionless-mode Network Service (ISO 8473). ISO/IEC 10589, International Organization for Standardization, April 1992.

[3] Joint Technical Committee ISO/IEC JTC 1 *Information Technology.* Information Processing Systems - Telecommunications and Information Exchange between Systems - Protocol for Exchange of Inter-domain Routeing Information among Intermediate Systems to Support Forwarding of ISO 8473 PDUs. ISO/IEC 10747, International Organization for Standardization, October 1993.

[4] B. Kumar and J. Crowcroft. Integrating security in inter-domain routing protocols. *Computer Communications Review*, 23(5), October 1993.

[5] Gary Malkin. Rip Version 2 Carrying Additional Information. Internet RFC 1723, Xylogics, Inc., November 1994.

[6] John Moy. Ospf Version 2. Internet RFC 1583, Proteon, Inc., March 1994.

[7] Radia Perlman. *Network Layer Protocols with Byzantine Robustness.* PhD thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, August 1988.

[8] Radia Perlman. *Interconnections: Bridges and Routers.* Addison-Wesley, Reading, Mass., 1992.

[9] Bruce Schneier. *Applied Cryptography: Protocols, Algorithms, and Source Code in C.* John Wiley & Sons, Inc., New York, 1994.

[10] Martha Steenstrup. Inter-domain Policy Routing Protocol Specification: Version 1. Internet RFC 1479, BBN Systems and Technologies, July 1993.