Secure Routing Protocols: Theory and Practice*

Feiyi Wang, Brian Vetter, Shyhtsun Felix Wu

{fwang2,bmvetter,wu}@eos.ncsu.edu

North Carolina State University

Abstract Most routing protocols are only designed to deal with simple network failures (e.g., links going up and down, nodes crashing and restarting etc.), and thus can have many vulnerabilities when facing a strategically placed intruder. The paper identifies these threats to routing protocols based on a unified routing framework, with focus on the information flow inside/outside a router. Real world attacking examples are given to illustrate the argument. Further, we review a few common security techniques and their pros and cons. Finally, we explore the rationales behind securing routing protocols and conclude with our remarks and summary.

1 Introduction

The spectacular growth of Internet has spawned an increased awareness of and interest in network security. Four top-level requirements for internet security have been identified: end-system security, end-to-end security, secure Quality of Service (QoS), and secure network infrastructure [9]. End system security is usually achieved by erecting a security wall for a single host, but recently firewall technology has been used to enlarge "defense perimeter" for whole organizations by protecting intranet with a small number of firewall systems. End-to-end security is well suited to provide confidentiality, authentication and integrity. Secure QoS poses a set of new security problems: authentication and authorization of users requesting expensive network resources, both to prevent theft of resources and to prevent denial of service due to unauthorized traffic etc. The above three areas are being actively studied and many of them are being deployed on today's Internet. There is another important class of security problems which concerns the network infrastructure. Network operation depends upon management and control protocols to configure and operate the network infrastructure, including routers, DNS servers etc. An attack on the network infrastructure can cause denial of service from the viewpoint of user, but for network manager or operator, the attacker is taking advantage of the lack of authenticity, integrity and possibly privacy.

The security importance of routing protocols, the heart of network operation, has only been addressed recently. Most routing protocols are only designed to deal with simple network failures (e.g., links going up and down, nodes crashing and restarting etc.), and thus can have many vulnerabilities when facing a strategically placed intruder. An earlier vulnerability analysis of routing protocol was given by Rosen[24]: the hardware fault of source node caused the halt of whole ARPANET. A recent accident occurs on April 27, 1997, a router from MAI Network services in Virginia absorbed about 50,000 network addresses which caused much of the Internet to be disconnected from 20 minutes to 3 hours. A technical bug was blamed to the MAI's Bay Network router, but the same attack is very feasible from an evil insider. Current research work focuses on security enhancement or design for a single specific routing protocol or a class of routing protocols: Kumar[18] discusses security threats to routing protocols from point of view of distance vector based and link stated based; Smith [1][2] discuss secure BGP and secure distance vector-based routing protocol; Murphy [3] discusses secure OSPF, a link state-based algorithm; Sirois [4] discusses the Nimrod routing architecture. One exception is Perlman's work [10][11], which is not based on any existing network protocol but on a realistic network model. Perlman presents two network layer routing protocol designs which are robust in the presence of Byzantine failures. The first is a design for robust routing based on robust flooding of packets, and the second is a link state routing protocol which uses robust flooding for distribution of control messages.

This paper is organized as follows: Section 2 defines a framework of routing protocols, which includes three parts: a topology model, an information model inside an intermediate system (IS) and an operation model, which emphasizes IS-IS interactions. Based on this framework, Section 3 will discuss the threats faced by routing protocols. We classify

* This research is supported in part by the U.S. Department of Defense Advanced Research Projects Agency and the U.S. Air Force Rome Laboratory under contract F30602-96-C0325, and in part by the Center for Advanced Communication and Computing under grant FAS #5-30183.

two sources of threats, external and internal. External threats come from outside intruders, whose objective is attacking the operation model by disrupting the normal operation of the routing protocol. Internal threats come from protocol participants (e.g. subverted routers), whose objective is attacking the information model within an IS by abusing the routing information in exposure. Later, we discuss the impact of a few security-related routing protocol features. Section 4 provides four working examples of attacks to current routing protocols (RIP and OSPF) to illustrate real threats. Section 5 and Section 6 discuss security requirements and security mechanisms currently in use or potentially available to protect routing protocols respectively. We highlight the challenges and solutions to applying security techniques to protect routing protocols. Section 7 is a preliminary summary of the rationales behind securing routing protocols. Finally, we conclude with summary and remarks.

2 Routing Protocol Framework

We propose a routing framework which derived from the ISO/OSI routing infrastructure[23] and is the basis of routing protocol security discussions in later sections. This framework includes three parts: The *topology model* defines the topology relationship of different routing protocols; The *information model* defines the information flow inside an intermediate system; And the *operation model,* which defines the general procedures involved in a routing protocol's operation, and reflects the information flow among ISs. Based on this framework, we will explore the vulnerabilities of routing protocol and security threats in the next section. The term internet, with lower case *i*, is used to refer to an arbitrary collection of connected networks, while Internet, with capitalized *I*, refers to the global network widely used today. Whenever possible, we use ISO/OSI terminology to describe network concepts.

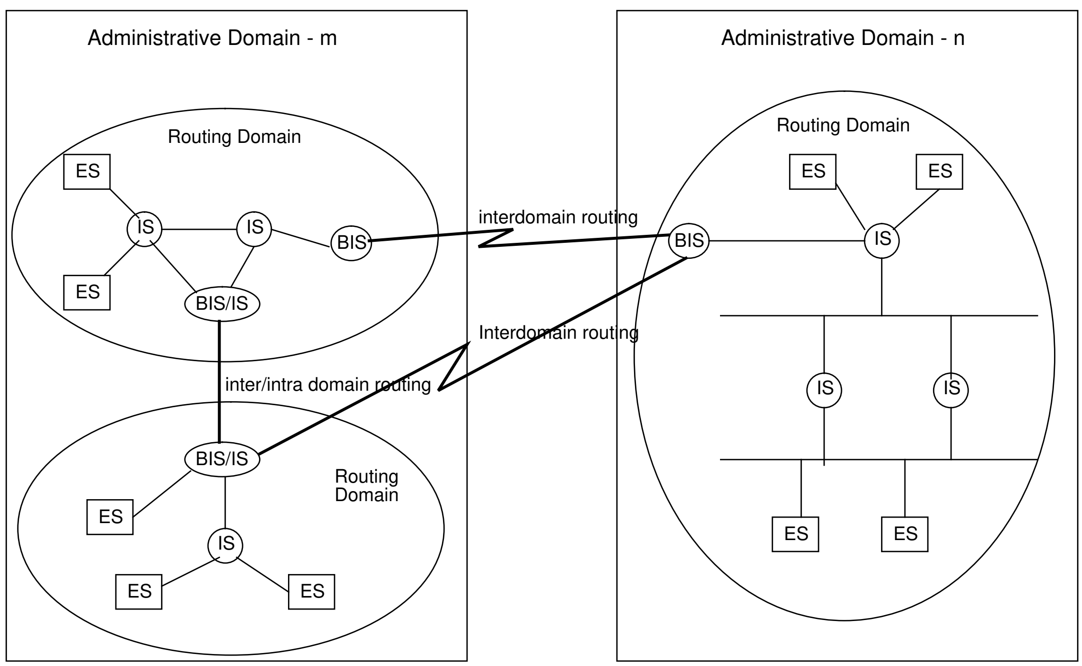

Figure 1. Illustrated topology relationship of routing protocols

Topology Model

Routing protocols need to deal with diverse network topologies. A common way to model network topology is to partition the network into two levels: intra-domain and inter-domain. Intra-domain routing handles routing procedures within a single provider/subscriber, where a *service provider* (or just provider) could be defined as organizations that share their resources with other organizations, and *service subscriber* (or just subscriber) could be defined as organizations that use other organization's resources. Inter-domain routing handles routing procedures that span multiple providers/subscribers. A single provider/subscriber that spans a contiguous segment of an internet topology forms an *administrative domain* (AD)*. The concept of an AD provides inter-domain routing a convenient model with which organizations that contribute their resources to the outside world can establish boundaries (e.g. firewalls) to protect and control access to their resources. A connected set of ISs that participate in a single instance of a particular intra-domain routing protocol forms a *routing domain* (RD). End system (ES) is a host system which usually doesn't participate routing. They are usually connected to intermediate systems (IS) which participate only in intra-domain routing. ISs which participate both in intra-domain and inter-domain routing are called *boundary intermediate systems* (BISs). In certain cases, a single administration may employ several intra-domain routing protocols, and thus an AD may consists of one or more RDs. Also, depending on the specific intra-domain routing protocol, a RD may further be divided into a hierarchy of several small routing areas. Later, we discuss the effect of hierarchy division with respect to routing security.

Figure 1 illustrates the topology model. There are two ADs in the figure, *m* and *n*. AD-*m* includes two routing domain, the boundary ISs in both routing domain can run either inter-domain routing protocols or intra-domain routing protocols, other IS can only talk with other ISs in the same routing domain using intra-domain routing. Also, BISs in AD-*m* and AD-*n* usually talk each other by running inter-domain routing protocol. A BIS must belong to one and only one AD, i.e. this topology model disallows overlapping ADs.

Information model inside an IS

The goal of connection-less network-layer protocol is to forward network-layer protocol data units (PDUs). When an IS receives a PDU, the IS makes its forwarding decision (determine the next hop) using two sources of information: PDU header (e.g. destination address) and a forwarding table, called the *forwarding information base* (FIB). Conceptually, each FIB entry consists of a destination; the subnetwork over which packets should be forwarded to reach that destination (also known as next hop); and some form of routing metrics, which expresses one or more characteristics of route (its delay property, expected error rates etc.) in terms that can be used to evaluate the suitability of this route to convey one particular packet or a class of packets. The forwarding mechanism is implemented in network layer, but the policy is governed by routing protocol. An IS constructs its FIB using routing information it receives as a result of participating one or more routing protocols. Every routing protocol maintains its own *routing information base* (RIB).

We have made two important distinctions, which are often neglected. One is the distinction between forwarding and routing. Forwarding consists of taking a packet, looking at its destination address, consulting the FIB table, and sending the packet in a direction determined by that table. Routing is the process by which routing protocol determines what goes into RIB (by policy) from which the FIB is built. Correspondingly, we need to distinguish between RIB and FIB: one is maintained by an individual routing protocol entity, one is maintained by network layer, usually in kernel. This information flow model is illustrated in Figure 2.

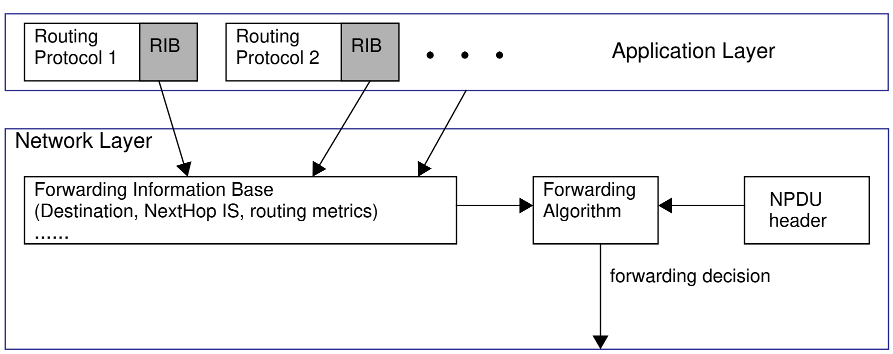

Figure 2. Routing information flow model inside an IS

Operation Model

---

* In Internet context, it is called *Autonomous System* (AS).

Routing protocols are complex concurrent systems. Any working routing protocol needs to deal with many engineering details. We now define a simplified model which reflects the core procedures involved with separation of a routing protocol. It includes five components:

(1) Neighbor Acquisition - defines how a IS/BIS acquires neighbor information. This can usually be realized by sending out `HELLO` messages on all outgoing interfaces.

(2) Neighbor Reachability - defines how an IS/BIS maintains the neighbor relationship with previously acquired neighbors. This can be usually realized by periodically broadcasting `KEEPALIVE` or `POLLING` messages.

(3) Routing Information Exchange - defines how and what routing information is exchanged among ISs. We further break the information into three pieces centered around the RIB.

- Neighbor-In-RIB, which stores information an IS/BIS receives from its neighbor.
- Neighbor-Out-RIB, which stores information an IS/BIS sends to its neighbors.
- Local RIB, which stores the routing information this particular IS/BIS needs, excluding whole transit traffic.

The relevance of maintaining the RIB is updating strategy. As the network changes dynamically (links and nodes go up and down), so must the routing information. We therefore need to define a strategy to propagate these changes. This can be done either by periodical update or event-drive polling or both. Also, the information updates could be complete or incremental, raw (complete) or aggregate (summarized). The separation of Neighbor-In-RIB and Neighbor-Out-RIB models any such routing information transformations at this stage.

(4) Route Generation and Selection - Based on the Local RIB, a route selection algorithm will determine what goes into the FIB. Previous choices about what kind of information used to propagate is crucial in choosing route-selection algorithm. Two popular algorithms are the distributed Bellman-Ford algorithm and Dijkstra's algorithm. Routing protocols are traditionally classified as either distance vector based or link state. The difference between them can be summarized as follows. In a distance vector routing algorithm, each IS communicates only with its directly connected neighbors, and describes to them everything it has learned about the network. In a link state routing algorithm, each node communicates the states of its directly connected links to all the nodes. Some routing protocols (e.g. BGP-4) adopt a *path vector* routing algorithm, which some researchers do not classify as distance vector. In practice, if a node possesses complete topology information, Dijkstra's Algorithm is preferred, if partial information, Bellman-Ford Algorithm is preferred. In Nimrod [21], a new experimental routing architecture, the authors claim that different route selection algorithms can co-exist in the same routing architecture, depending on the routing information that an IS has. In that case, it is difficult to say exactly if the protocol is distance vector-based or link-state based. For this model, we try to avoid this restriction by emphasizing the exchanged information instead of the route generation and selection algorithm. We believe this information oriented model will facilitate our analysis of the routing protocol security problem.

(5) Neighbor Relation Termination - defines how to terminate a neighbor relationship. There could be several causes for termination a neighbor relationship: Link layer notification of a change of underlying link status, a BIS/IS changes the neighbor's `STATUS` and piggybacks it in routing update, or additional control message explicitly causing the change.

In summary, we define a routing protocol framework which characterizes the essential routing components. In terms of security, we are more interested in the latter two parts: the information model inside an IS and the operation model. The following section will discuss threats to routing protocols.

3 Threats to Routing Protocols

Based on the preceding routing protocol framework, we now explore vulnerabilities and possible attacks. Broadly speaking, there are two kinds of attacks. One is the *active attack*, which is an attempt to improperly modify data, gain authentication, or gain authorization by inserting false packets into the data stream or by modifying packets transiting within the data stream. Second is the *passive attack*, which is an attack on the authorization system which inserts nothing into the data stream, but instead passively monitors information being sent between other parties. For attacks to routing protocols, our main concern is active attack.

Threats to routing protocols come mainly from two sources, external and internal. External threats come from outside intruders, i.e. non-participants in the protocol, whose objective is the disruption of the normal operation of routing protocol by attacking the operation model. Internal threats come from subverted protocol participants. E.g. if an IS is compromised, the information inside the IS which is in exposure can be abused. We will detail the analysis in the following. Further, we observe that certain characteristics of routing protocols are related to the vulnerability of those protocols to specific threats, so we conclude this section with a few favored design features with respect to security.

Threats From Outside Intruder

An outsider intruder could break into a routing domain in various ways. Such an intruder is a threat to the operation model which has been discussed in Section 2. Specific threats include:

- Breaking the neighbor relationship - An intelligent filter placed by an intruder on a communication link between two ISs could modify or change information in the routing updates or even intercept traffic belonging to any data session. For example. if `KEEPALIVE` messages are filtered out, then the neighbor relationship is terminated.
- Replay attack - An intruder could passively collect routing information. Later, the intruder could retransmit "obsolete" routing information messages. If obsolete information is accepted and disseminated, a normal IS could make incorrect routing decision. Even if rejected, this is also a kind of denial of service.
- Masquerading - During the neighbor acquisition process, a outside intruder could masquerade an nonexistent or existing IS by attaching itself to communication link and illegally joining in the routing protocol domain by compromising authentication system. The threat of masquerading is almost the same as that of a compromised IS.
- Passive Listening and traffic analysis - The intruder could passively gather exposed routing information. Such an attack can not effect the operation of routing protocol, but it is a breach of user trust to routing the protocol. Thus, sensitive routing information should be protected. However, the confidentiality of user data is not the responsibility of a routing protocol.

Threats From A Compromised IS

In this section we assume the threat is from an IS which has been subverted. Thus all information inside the IS (e.g. FIB and RIB) is in exposure and at risk. The attacks from Section also apply to the case of compromised IS, but there are additional threats in this case.

First, the FIB could be directly manipulated system commands or kernel interfaces and disrupt the network layer decisions. One possible thing you can do is misroute or reroute. By seizing the control of an IS, you could add an route entry to FIB which will reroute data traffic to a particular destination. It is a breach of information confidentiality and integrity. You could randomly modify FIB to make router misroute, which is a kind of denial of service attack.

Second, you can abuse the RIB information for your purpose. Depending on how much Neighbor-In-RIB, Neighbor-Out-RIB, Local RIB are stored, you may able to forge other ISs' routing information or supply the wrong routing information. For example: you can claim a link exist or not, down or up, modify a particular control field. The damages of this type of attack are variable, such as crashing the network, partitioning the network, non-optimal route, congestion etc. It really depend on how clever the attacking scheme is and how robust the attacked routing protocol is.

The compromised IS problem has not received much attention to date, for several reasons. First, there are usually much fewer ISs in a routing domain than hosts, and they are usually under the tight control and monitoring of network administrators. Therefore, the ISs have a larger defense perimeter than the ordinary hosts. In other words, the trust of them from humans is high. Second, routing protocols are distributed and cooperative in nature (i.e. ISs in a AD or RD must coordinate or cooperate to meet their protocol requirement) and thus there is a tradition of trust in routing protocol design. If a IS was compromised, the trust relationship would be broken. Most routing protocol don't have preparation to deal with that. One the other hand, IS security is not impregnable with growing size of internetwork. We believe compromised IS should play no small part in the future routing protocol design.

Security-related Characteristics of Routing Protocols

In face of such threats, different routing protocols have different degrees of immunity. There are three characteristics we think are favorable to secure routing protocols.

- Self-stabilization - this concept was first come up with by E.W.Dijkstra [22] in distributed control context. In routing context, it means after some unforeseen disruption to routing, the network will return to normal operation without human intervention within a reasonable time, provided that the faulty hardware is disconnected from the network or repaired. We will see this character is important to security and fault detection, though it doesn't guarantee the network will operate properly when a piece of malicious code attached to the network.
- Byzantine Robustness - the routing protocol should perform correctly under unusual or unforeseen circumstances such as hardware failure, extreme condition, incorrect implementation. Especially when we consider security, the ideal robustness is Byzantine Robustness[10]. The term Byzantine failure is taken from a famous problem in computer science known as Byzantine generals problem. A Byzantine failure is the one in which a node fails not by simply ceasing operation but instead by performing arbitrarily. A network with Byzantine robustness is the one which would be able to continue working properly even if some portion of nodes had Byzantine failures.
- Fault detection - Most of today's network will not operate properly in the face of actively malfunctioning node, though they have some ability to detect "natural" faults. One favorable design feature would be: if some IS goes wrong, we can detect it quickly or better, identify the source and isolate the problem. Combine with Byzantine robustness, we can even recover from the failure without human intervention. Byzantine robustness alone can't provide this feature. One thing to note is that this design feature couldn't be accomplished only by routing protocol, help from management layer is a must. How to integrate routing security with fault detection to provide an integrated security environment remains a open issue.

4 Example Attack Analysis

Routing protocol threats are usually specific to particular security weakness (not necessary design weakness) of the protocol. In this section, we will discuss several well known and several not-so-well known examples, which are the subject of current research within our group. The first is the *Black Hole Attack*, a well-know attack which is peculiar to distance vector routing algorithms. The rest of the attacks, namely the *Table Overflow Attack*, the *Age Field Attack*, and the *Sequence Number Attack*, are all based on the OSPF routing protocol[14][15][16], a very robust link state routing protocol, which aims to be the future intradomain routing protocol of choice and may replace the current widely used RIP protocol. We examine OSPF with respect to above scenarios and some preliminary conclusion are given.

Black Hole Attack

In a distance vector routing protocol like RIP, a router, say *X*, will periodically receive link updates from its neighbor, say *Y*. Conceptually, this update information will include (*Destination ID, Cost/distance to the destination, and Next Hop ID*). When *X* receives update information regarding a change in link status around *X*, e.g. a change in cost, it will use the distributed Bellman-Ford algorithm to recompute the distance to the destination. If necessary, it will propagate this result to its neighbors. The attack is as follows: assume *X* is compromised and claims he has the shortest path to a particular network address. *X* sends an update to its neighbors, none of which can confirm the validity of *X*'s claim because in a distance vector algorithm, *X* only propagates the *results* of distance computations. After recomputation each neighbor will propagate some new shortest path to their neighbors. This scenario continues, expanding outward from *X*. Eventually, all the traffic to that address will be attracted to *X*, hence the name Black Hole.

Table Overflow Attack

Historically, there have been examples of the rapid growth of the Internet resulting in routing table overflows in core routers [13]. A faulty router can also cause overflow by launching a strategically placed attack. An example from OSPF works as following: AS external link (Type 5) advertisements are originated by AS boundary routers. A separate advertisement is made for each destination which is external to the AS. One feature unique to this type of LSA makes it an attractive prospect for such an attack: It is the only type of LSA that is flooded throughout the entire Autonomous System (except for Stub areas); all other types of link state advertisement are specific to a single area. A compromised AS boundary router can generate a deluge of external LSAs, and each of them will be flooded to every router in the AS. The is no mechanism in specification to verify that these external LSAs are valid. Eventually, every router will be filled in these junk LSAs, preventing the routing protocol from successfully installing any new network entries. This could also be disastrous if the implementation does not have proper overflow protection mechanisms, as the routers will likely crash.

Age Field Attack

In the OSPF protocol, an age field is used to keep track of how long a given LSA has been in the system. The age of each LSA is increased by every router when propagating the LSA and while the LSA is installed in the link state database, thus ensuring that LSAs have a finite lifetime of one hour. The agefield of an LSA, along with the LSA's sequence number and checksum, allow a router to determine which is the most recent when multiple instances of an LSA are encountered. When the age of an LSA stored in any router's database reaches MaxAge, the router will purge the LSA from its own database, and it then floods the MaxAge LSA to the rest of the network, so that other routers also purge the obsolete LSA. The intention of this design is to purge obsolete LSAs from the routing domain as soon as possible and make the link state database converge rapidly. A malicious router could take advantage of this feature and launch a MaxAge attack by modifying LSA age fields to be MaxAge, therefore causing unnecessary flooding and refreshment. Such an attack not only consumes network bandwidth, but also makes the routing information database inconsistent, disrupting correct routing.

Sequence Number Attack

Implementation bugs are also serious threats. We have discovered such a threat in our experimental work with the OSPF protocol. In the OSPF specification[14], the LSA sequence number is a signed 32-bit integer. It is used to detect old and duplicate link state advertisements, i.e. successive instances of an LSA have successive sequence numbers. The space of sequence numbers is linearly ordered, and the larger the sequence number the more recent LSA is. With `0x80000000` reserved and unused, a router uses `0x80000001` to number its first originated LSA. When a new instance of a particular LSA is generated, the sequence number is set to the sequence number of the last (now obsolete) instance plus one. Thus when flooded, each router can determine which is the newer instance by examining the sequence numbers. It is also possible for a router to receive an obsolete instance of its own LSA with a sequence number greater than the current sequence number. This could happen if the router loses state (crashes) or if the network has been partitioned. The router must then generate a new instance of that LSA with a greater sequence number, or in some cases, flush the LSA by reflooding with the age set to Maxage (called premature aging). When an attempt is made to increment the sequence number past maximum value (`0x7fffffff`), any current instance of the LSA being generated must first be flushed from the routing domain before the new instance of the LSA is generated with sequence number `0x80000001`. If the implementation does not implement this sequence wrap-around process correctly (i.e. does not first flush the existing LSA), a newly generated instance of the LSA will be rejected by other routers because this obsolete LSA has the larger sequence number and is thus, by definition, *newer*. The result is that the obsolete LSA will stay in the RIB until the obsolete LSA ages out, and during this time the link state database will remain inconsistent, Figure 3 illustrate the above attack.

One thing needs to point out is that In normal operation, the chance of a router reaching the maximum sequence number is extremely small, as the protocol mandates that a router not generate instances of a particular LSA more frequently than every MinLSInterval = 5 seconds. Thus it should take a minimum of 5×2^{31} seconds = 340 years for an LSA to reach `0x7fffffff`. For the sake of completeness, a simple computation can tell us maximum period for a sequence number reach `0x7fffffff.`Suppose the network is extremely stable and once initialized, there is no change at all. Since the OSPF specification mandates every 30 minutes the originator will refresh the LSA with the sequence number of current instance plus one even no change to the LSA at all, so it needs about $(30 \times 60) / 5 \times 340 =$ 122400 years. However, a malicious or faulty router could certainly speed up this process, as we have done in our experiments.

5 Requirements for Secure Routing Protocols

In spite of the diversity of the above attacks, they actually take advantage of the lack of authenticity, integrity or possibly, confidentiality. We now define these services in the context of secure routing protocols.

- *Authentication services* are primarily concerned with the providing assurances about the identity of an entity. In a routing protocol context, when a router sends out a routing message, the identity of the originator of the information should be able to be validated.
- *Integrity services* ensure that the data being transmitted is consistent with the data being received. We do not require complete data integrity. For example, we need not prevent data copying, as along as it does not jeopardize routing.

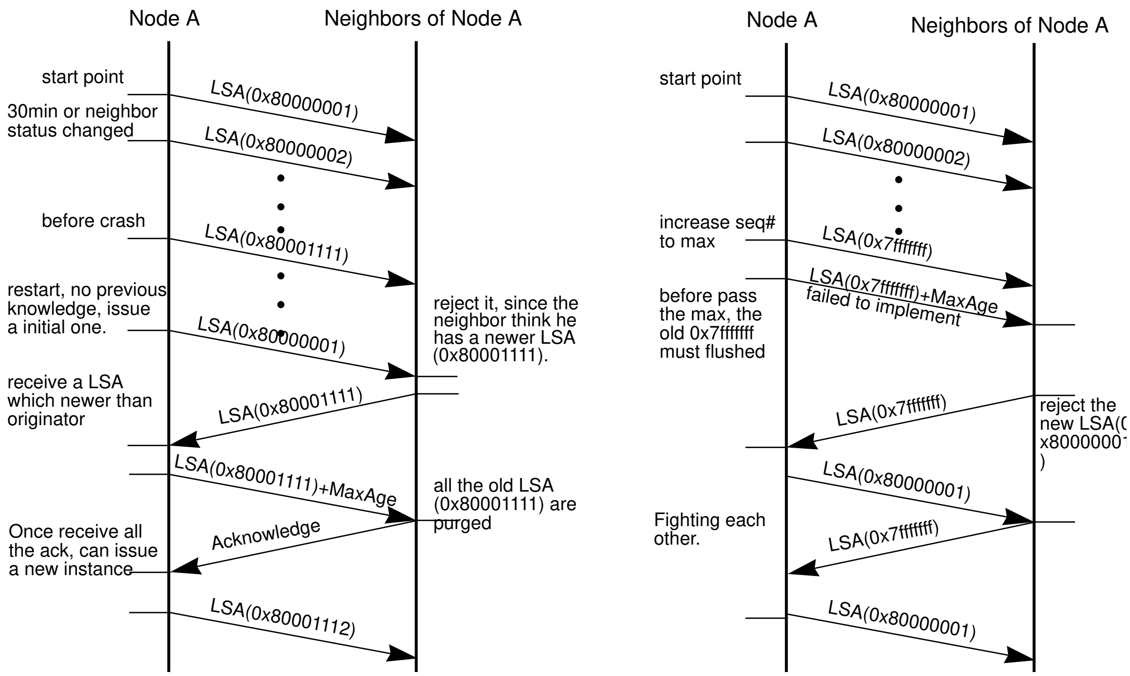

Figure 3. Illustration of Sequence Number Attack

- *Non-repudiation services* provide irrefutable evidence that a certain event took place. An common example of its use is to prevent a sender from denying that it sent a message when it actually did, or from a sender claiming that it sent information that it actually did not. Note that integrity is a prerequisite for non-repudiation.
- *Confidentiality service* provides privacy of routing message, which use encryption to prevent others from knowing what the routing message is. No current routing protocol supports it, OSPF for IPv6 relies on IPSEC[6][7][8] to provide authentication and confidentiality, you can also combine kerberos and it could be used in cases where routing information is so sensitive that its privacy is a serious requirement.

In general, to secure a routing protocol requires that important if not all routing information be authenticated between neighboring routers. The source of all route information should also be authenticated. The methodology adopted is quite straightforward. First, analyze the protocol to identify its vulnerabilities and possible threats. Second, the security requirements, based on the running environment and those envisioned for the future, must be determined. Finally, existing cryptographic techniques and authentication mechanisms can be incorporated into the design to address these issues. Usually, in terms of intended use, accessibility, and network connectivity, different security mechanisms may be needed to address the security requirements. In the following section, we discuss the security mechanisms currently available to help secure routing protocols.

6 Mechanisms for Secure Routing Protocols

Cryptography is the building block in providing privacy, integrity, authentication. For routing protocol security, we are concerned with integrating different authentication schemes with routing to provide secure routing services and satisfy security requirements. First we briefly introduce the concept of cryptographic algorithms, then we concentrate on various authentication techniques in the context of secure routing protocols. Further, we discusses the key distribution and management problem. Finally special care for the routing information header are considered.

Cryptography

Cryptography is the basis for secure anything. Cryptography can provide confidentiality and integrity of either data or traffic flow information. It can be used alone or in combination with a number of other security mechanisms, as described in the following sections. A cryptographic algorithm must be reversible and there are two general classifications of reversible encryption algorithms:

- symmetric (i.e. secret key) cryptographic algorithms, in which knowledge of the encryption key implies knowledge of the decryption key and vice-versa. Data Encryption Standard (DES) is the best-known example of a secret key encryption algorithm.
- asymmetric (e.g. public key) encryption, in which knowledge of the encryption key does not imply knowledge of the decryption key or vice-versa. The two keys of such a system are sometimes referred to as the public key and the private key. RSA is best-known public key encryption algorithm.

There is a third type of cryptographic algorithm, called a hash or message digest function, which map a potentially large message into a small fixed length number, analogous to the way a regular hash function maps values from a large space into a small space. In routing security context, we are interested a variant of it, keyed Message Digest Algorithm which ca provide both authentication and integrity. The most widely used of such algorithms is keyed Message Digest version 5 (MD5).

To support authentication, routing protocol designers usually reserve certain fields in the routing packet header. Incorporating security mechanism into routing protocol will pretty much to do with how to use these fields. Many protocols define `NOAUTH` options which provide none of security protection for the operation of routing protocol. An obvious reason to do so is that any incorporation of security mechanism will inevitably complicate the system design and when security conflicts with efficiency and performance goal, they drop the security or adopt a weak and simple authentication scheme. However, there is trend that stronger authentication technique should be used to provide protection of routing protocol. We will first discuss the techniques currently in use to secure routing protocols, then we will further identify other possible security techniques used to secure routing protocols.

Current Techniques Used To Secure Routing Protocols

We now describe three authentication techniques which are currently used or proposed in routing protocol design.

(1) Simple password scheme - the simple password check is by far the most common form of authentication. The secret password is carried in the packet header, without benefit of further protection. The scope of simple passwords can be entire networks or on a per- interface (i.e. per-subnet) basis. For per-interface scheme, each router must have knowledge of its neighbors' passwords to authenticate them. The simple password authentication system is said to be "disclosing" because the key is transmitted *in the clear* with the routing information over a network, and thus can be disclosed to eavesdroppers. Recent sniffer attack make the plaintext password less useful and attractive.

(2) Message Digest Signature - Message digest schemes use a one-way hash function to generate a signature based on the message information in conjunction with a secret key. The hash value is thus unique to the particular message and key, and thus only an authentic party with knowledge of the key could have generated it. This scheme is usually implemented using Keyed-MD5.

(3) Public Key Based Digital Signature - These schemes consist of two independent procedures: (a). signing a message; and (b). verifying a signed message. The first process uses information (cryptographic key) which is private (i.e. unique and confidential) to the signer. The second process uses procedures and information which are publicly available, but from which the signer's private information cannot be deduced. The signing process involves either an encryption of entire data unit or the production of a check-value of the message and then signing that, using the signer's private information (private key). The verifying process involves using the public procedures and information to determine whether the signature was produced with the signer's private information. Therefore, A digital signature attached to a piece of routing information by the source router not only provides assurance of authentication, but also and ensures message integrity and uniquely identifies the information's originator. The RSA algorithm usually used to implement such schemes. In practice, the RSA public exponent is usually much smaller than the RSA private exponent, and thus the verification of a signature involves less work than the message signing process. This is desirable in some routing protocol contexts (e.g. OSPF) in which a message will be signed by an individual only once, but the signature must be verified many times.

Comparison

The major advantage of simple password scheme is its simplicity and performance: the password can be checked very quickly through a simple comparison. Its usefulness is limited, however. Both message digests and digital signatures can provide hop-by-hop authentication and integrity, but only digital signatures can provide the same functionality to end to end. Also, public key based digital signatures can be the basis for non-repudiation. Although public key based digital signature schemes have significant advantages, they suffer from two problems: (1) performance: the cryptographic transformation required in the RSA algorithm is much slower than the incremental hash calculation (including appended secret key) used in Key-MD5. (2) Patent exportation problem, famous RSA algorithm is under U.S. patent and export of RSA falls under the same U.S. laws as all other cryptographic products. RSA used for authentication is more easily exported than when it is used for privacy. In the former case, export is allowed regardless of key (modulus) size, although the exporter must demonstrate that the product cannot be easily converted to use for encryption. In the case of RSA used for privacy (encryption), the U.S. government generally does not allow export if the key size exceeds 512 bits. The first problem could be solved by technique invention, one way or another. but the second one is beyond the technical scope.

Providing Privacy for Routing Protocols

We identify another security techniques which can be used to enhance routing protocol security - *using digital envelopes to provide privacy.* Providing privacy to routing information is a subject which is usually ignored since many people either think there is no sense to provide privacy for routing information or it costs too much to pay for it. If there is a requirement for privacy in addition to authentication, *digital envelopes* might become a good candidate. Public key cryptography has the well-known significant speed disadvantage compared to many symmetric encryption methods. Digital envelopes combine public key and secret-key cryptography together to achieve both the speed advantage of a secret-key system and the security advantages of a public-key system. Specifically, a public-key system is used to encrypt a secret key which is then used to encrypt the message with a fast symmetric algorithm. Both the encrypted message and the encrypted message-key form the Digital Envelope, which is sent to the receiver. The receiver uses the public key system to decrypt the message-key, which is in turn used with the symmetric algorithm to decrypt the message itself.

For example, suppose Alice wishes to send an encrypted routing message to Bob. She first encrypts the message with DES, using a randomly chosen DES key. Then she looks up Bob's public key and uses it to encrypt the DES key. The DES-encrypted message and the RSA-encrypted DES key together form the RSA digital envelope and are sent to Bob. Upon receiving the digital envelope, Bob decrypts the DES key with his private key, then uses the DES key to decrypt to message itself. This combines the high speed of DES with the key-management convenience of RSA, Figure 4 is an illustration of above algorithm.

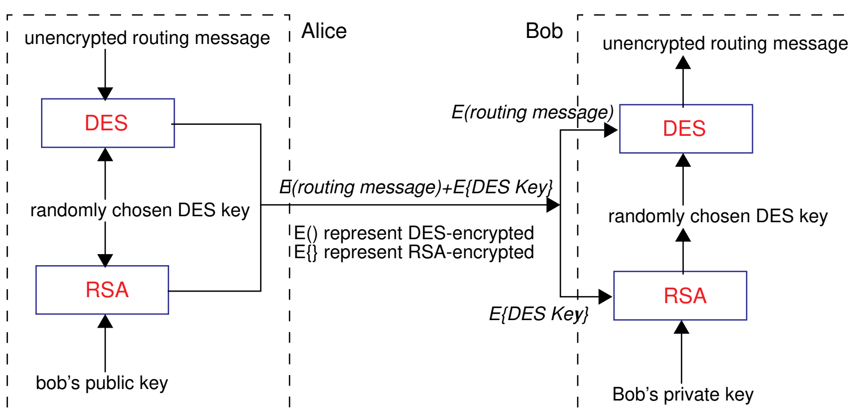

Figure 4. Illustrating using digital envelope to provide privacy

Key Management & Key Distribution

The discussion thus far has periodically mentioned keys, either for encryption or for authentication (e.g. as input to a digital signature function). Suppose we are considering a Message Digest Signature scheme using pair-wise keys, i.e. a unique secret shared by each pair of routers (source, destination). For N routers, this would require $\binom{N}{2}$ unique keys, which is an order of $\Theta(N^2)$. Whenever a router is added or removed, keys in every router must be changed. Obviously, a secure and scalable key distribution and management scheme is very important, and must be chosen carefully to fit into the security solution of the whole system. We identify three basic key management techniques:

(1) Manual configuration - this is the simplest and secure, but it is tedious and error-prone. It may only suitable for small environments.

(2) Automatic key distribution and management - this can be embedded key distribution mechanism into routing protocol and make the work automatically done.

(3) Using existing key management scheme in layer, such as using ISKAMP deployed in IP layer or use secure DNS to do key management work.

In summary, we have discussed a few common techniques used in secure protocol design. In a real environment, things can get quite complicated, and more subtle techniques may be needed to meet specific security requirements. In the next section we examine some principle problems related to secure routing protocol more closely.

7 Remarks on Secure Routing Protocols

In this section, we explore the underlying nature of routing protocols which help one is better than another one in terms of security. We discuss a few preliminary observations.

Hierarchical Routing

Hierarchical Routing is the primary way to deal with scalability issue, though it bring us other disadvantages such as suboptimal routing and processing overhead. Here we argue that hierarchical routing structure is also favorable with respect to security, The security advantage is that a well designed algorithm can and should be able to contain certain problems to a small portion of the hierarchy, leaving other portions mostly unaffected. For example, in OSPF an Autonomous System is divided into areas. A router has a separate topological database for each area it is connected to. Routing takes place on two levels, depending on whether the source and destination of a packet reside in the same area (intra-area routing) or different areas (inter-area routing). Since intra-area routing relies only on information from within that area, it is not vulnerable to problems in another area. Likewise, problems in one area would not affect inter-area routing among other areas.

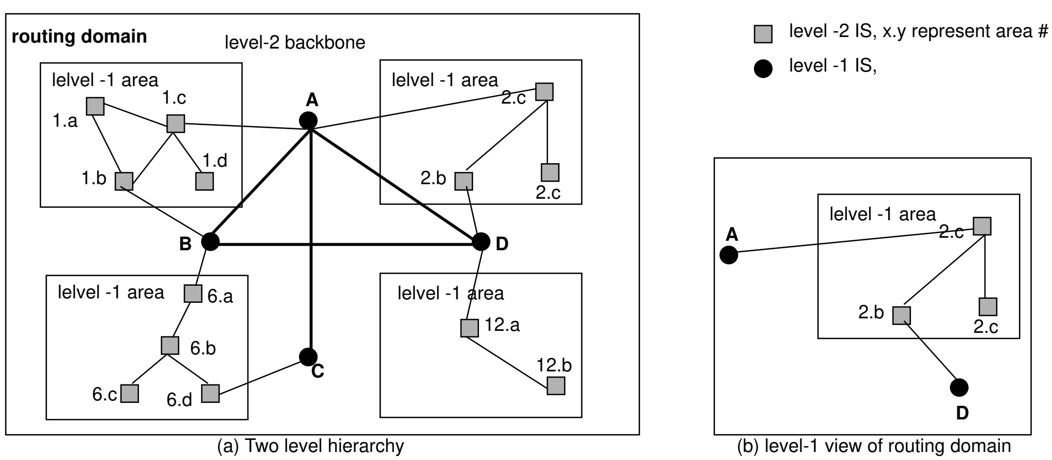

Figure 5. Two-level Hierarchy Routing and level-1 view of RD

As another example, we consider IS-IS [12], an OSI intradomain routing protocol. Two level hierarchy are defined, ISs in different levels are named “level-1 IS” and “level-2 IS”. Compared with OSPF, IS-IS has more strict hierarchy requirement: level-1 IS need to know only about the ESs and other level-1 ISs in its own level-1 area and about the nearest level-2 IS which it can use for traffic destined for other routing domain. while OSPF level-1 ISs have knowledge about other level-2 ISs so that it can choose which level-2

IS offers the best path to destinations, in some sense, with the trade-off against security, see Figure 5 (a) for a illustration of two-level of routing hierarchy.

Least information and privilege

It states that a system is most robust when it is structured to demand the least privilege from its component. A simple reasoning behind the principle is "*never give one more than what he needs*". For example, message digest could be widely used to validate the routing control message, this usually require a share secret key. If you issue the whole domain a single key, it is a violation of above principle because we only trust the router with respect to the routes involving it. If a router is compromised, a single domain key would make the router generate any bogus traffic he wants. On the other hand, if we use pair-wise key, a compromised router could only affect the route he has direct connection, which is the part he could control/damage anyway. As another example, we consider hierarchy routing in above Figure 5 (b), level-2 ISs frees level-1 ISs from know anything more than how to route to the nearest level-2 IS. Once a packet reaches a level-2 IS, it can reach its destination level-1 area (or another RD entirely) via level-2 IS routing exclusively. If one of level-2 IS was compromised, it should be less harmful if this level-1 doesn't have much knowledge about outside word (level-2 ISs) and the damage can possibly be limited inside level-1 area.

Least information dependency

Routing is a distributed process, as every router needs to exchange information to determine its best path to destination. We claim that a secure protocol design should strive to make acquired routing information as independent as possible. An example will illustrate our idea.In a pure distance vector algorithm (like RIP) routers distribute computational results rather than *raw* topological information. In other words, each router sends only *summarized* information, and thus each router actually only possesses aggregate information from its neighbors. This has two results. First, it is very hard for a router to validate the information it receives Second, even if a router detects incorrect information, it is also difficult to determine the source of that information.

By comparison, in a link state routing algorithm, each router generates information about its local topology (e.g. its neighbors), and also forwards such information from other routers via flooding. This has several good consequences. As such, every router independently possesses the entire topology information for the network. Since each router is responsible only for their own local portion of the topology, As long as one of your neighbor is honest, you can get *raw independent information* from the whole world. Obviously, it is easier (compared to distance vector) to find which router is lying. Recently, a few researchers have proposed predecessor-based distance vector algorithms prove it from another side. A predecessor is essentially a piece of information provided by source to alleviate the total blindness of router. By going through the predecessor, a router can sort of reconstruct the shortest path tree back to the source. It is predecessor make secure a distance-vector based algorithm possible.

Self-stabilization

Self-stabilization is an important characteristic which we believe routing protocols should strive for. This property guarantees that once any malfunctioning nodes are removed from the network, the network will resume normal operation in a finite period of time without human intervention. This makes network much less vulnerable to sabotage. An important result from this property is if the intruder want to make the network disruptive, *he must keep his bad behavior persistent*. Jump out the control plane, any persistent bad behavior can make network management work a lot easier. Consider such an scenario: an intruder breaks in, injects a few bad packets and lets the network remain disruptive, after erase/cover the trail, he escape. Now, the network manager will face the trouble to figure out where the source is.

Engineering Trade-off

Any security mechanism will complicate the routing protocol itself. Engineering trade-off are inevitable. They generally fall into two classes.

(1) Performance - public key (PK) based digital signatures provide most nice things we need, but PK is notorious for its intensive computational burden which makes performance a great concern. Also, export limitation is a concern with any cryptographic solution.

(2) Convenience - we saw in the OSPF table overflow attack discussion that every external LSA is flooded to all non-stub areas. This provides an efficient facility to propagate external routing information, but it also weaken the pro-

tection provided by the hierarchical routing scheme. Although the attack exploited the inability of verifying whether an external LSA is valid. the fact that the external LSAs are forwarded throughout the AS increases the scope of attack damage. Similarly, it is most convenient from an ease of key management perspective to use a single key throughout an AS. But form a security perspective, we prefer pair-wise keys because it follows least and consistent privilege rules

In summary, a well designed secure protocol is a well-balanced integrated system of security, performance, and convenience.

8 Summary & Remarks

In summary, we propose a routing framework and identify various threats to routing protocols based on the framework. Common security techniques used to secure routing protocols are discussed and further, their pros and cons. A few principles are presented here based on our understanding and experience to secure routing protocols. We observe that public key based digital signature is a promising way to thwart most of the threats, but before it can be widely deployed and accepted, we have to conquer the performance issue associated with the expensive public key algorithm.

Reference

[1] Bradley R. Smith, J.J Garcia-Luna-Aceves, "Securing the Border Gateway Routing Protocol", *Proc. Global Internet'96*, London, UK, 20-21 November 1996.

[2] Bradley R. Smith, Shree Murthy, J.J. Garcia-Luna-Aceves, "Securing Distance-Vector Routing Protocols", *IEEE/ISOC Symposiums on Network and Distributed System Security*, 1996.

[3] S.L. Murphy, "Digital Signature Protection of the OSPF Routing Protocol", *IEEE/ISOC Symposiums on Network and Distributed System Security*, 1996.

[4] Karen E. Sirois, Stephen T. Kent, "Securing the Nimrod Routing Architecture", *IEEE/ISOC Symposiums on Network and Distributed System Security*, 1997.

[5] Angus Mackinnon, Tony McGregor and James Breen, "Overview of Internet Protocol Security", Technical Report 94-10, Monash University.

[6] R. Atkinson, "Security Architecture for the Internet Protocol", RFC 1825, August, 1995.

[7] R. Atkinson, "IP Authentication Header", RFC 1826, August, 1995.

[8] R. Atkinson, "IP Encapsulating Security Payload (ESP)", RFC 1827, August, 1995.

[9] R. Braden, D. Clark, S. Crocker, C. Huitema, "Report of IAB Workshop on Security in the internet Architecture", RFC 1636.

[10] Radia Perlman, "Network Layer Protocols With Byzantine Robustness," MIT/LCS/TR-429, October, 1988.

[11] Radia Perlman, "Fault-Tolerant Broadcast of Routing Information", *Computer Networks*, no. 7, p. 395-405, 1983.

[12] Radia Perlman, "Interconnections: Bridges and Routers", Addison-Wesley, 1992.

[13] Douglas Comer, "Internetworking with TCP/IP", vol. 1, third edition, Prentice Hall, 1995.

[14] J. Moy, "OSPF Version 2", RFC 1583, March, 1994.

[15] J. Moy, "OSPF protocol analysis", RFC 1245, July, 1991.

[16] J. Moy, "Experience with the OSPF protocol", RFC 1246, July, 1991.

[17] Y. Rekhter & T. Li, "A Border Gateway Protocol 4 (BGP-4)", RFC 1771, March, 1995

[18] Brijesh Kumar, "Integration of Security in Network Routing Protocols", *SIGSAC Review*, vol. 11, no. 2, Spring, 1993.

[19] Felix Wu, F. Wang, B. Vetter, "Intrusion Detection for Link-State Routing Protocols", draft, January, 1997.

[20] RSA Labs FAQ, http://www.rsa.com/rsalabs/newfaq

[21] I. Castineyra, N. Chiappa, M. Steenstrup, “The Nimrod Routing Architecture”, RFC 1992, August, 1996.

[22] E.W.Dijkstra, “Self-stabilizing Systems in Spite of Distributed Control”, Communications of the ACM, vol. 1, no. 11, p 643-644, November, 1974.

[23] David M. Piscitello, A. Lyman Chapin, “Open Systems Networking”, Addison Wesley, 1993

[24] Eric C. Rosen, “Vulnerabilities of Network Control Protocols: An Example”, Computer Communication Review, p. 10-16, July, 1981.