Network Working Group
Request for Comments: 3756
Category: Informational

P. Nikander, Ed.
Ericsson Research Nomadic Lab
J. Kempf
DoCoMo USA Labs
E. Nordmark
Sun Microsystems Laboratories

May 2004

IPv6 Neighbor Discovery (ND) Trust Models and Threats

Status of this Memo

This memo provides information for the Internet community. It does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Copyright Notice

Abstract

The existing IETF standards specify that IPv6 Neighbor Discovery (ND) and Address Autoconfiguration mechanisms may be protected with IPsec Authentication Header (AH). However, the current specifications limit the security solutions to manual keying due to practical problems faced with automatic key management. This document specifies three different trust models and discusses the threats pertinent to IPv6 Neighbor Discovery. The purpose of this discussion is to define the requirements for Securing IPv6 Neighbor Discovery.

Table of Contents

1. Introduction 2
   1.1. Remarks 3
2. Previous Work. 4
3. Trust Models 4
   3.1. Corporate Intranet Model. 5
   3.2. Public Wireless Network with an Operator. 6
   3.3. Ad Hoc Network. 7
4. Threats on a (Public) Multi-Access Link. 8
   4.1. Non router/routing related threats. 9
        4.1.1. Neighbor Solicitation/Advertisement Spoofing . . . 9
        4.1.2. Neighbor Unreachability Detection (NUD) failure. . 10
        4.1.3. Duplicate Address Detection DoS Attack 11
   4.2. Router/routing involving threats. 12
        4.2.1. Malicious Last Hop Router. 12

4.2.2. Default router is 'killed' 13
4.2.3. Good Router Goes Bad 14
4.2.4. Spoofed Redirect Message 14
4.2.5. Bogus On-Link Prefix 14
4.2.6. Bogus Address Configuration Prefix 15
4.2.7. Parameter Spoofing 16
4.3. Replay attacks and remotely exploitable attacks 17
4.3.1. Replay attacks 17
4.3.2. Neighbor Discovery DoS Attack. 18
4.4. Summary of the attacks. 19
5. Security Considerations. 20
6. Acknowledgements 21
7. Informative References 21
Authors' Addresses 22
Full Copyright Statement 23

1. Introduction

The IPv6 Neighbor Discovery (ND) RFC 2461 [2] and Address Autoconfiguration RFC 2462 [3] mechanisms are used by nodes in an IPv6 network to learn the local topology, including the IP to MAC address mappings for the local nodes, the IP and MAC addresses of the routers present in the local network, and the routing prefixes served by the local routers. The current specifications suggest that IPsec AH RFC 2402 [1] may be used to secure the mechanisms, but does not specify how. It appears that using current AH mechanisms is problematic due to key management problems [8].

To solve the problem, the Secure Neighbor Discovery (SEND) working group was chartered in Fall 2002. The goal of the working group is to define protocol support for securing IPv6 Neighbor Discovery without requiring excessive manual keying.

The purpose of this document is to define the types of networks in which the Secure IPv6 Neighbor Discovery mechanisms are expected to work, and the threats that the security protocol(s) must address. To fulfill this purpose, this document first defines three different trust models, roughly corresponding to secured corporate intranets, public wireless access networks, and pure ad hoc networks. After that, a number of threats are discussed in the light of these trust models. The threat catalog is aimed to be exhaustive, but it is likely that some threats are still missing. Thus, ideas for new threats to consider are solicited.

1.1. Remarks

Note that the SEND WG charter limits the scope of the working group to secure Neighbor Discovery functions. Furthermore, the charter explicitly mentions zero configuration as a fundamental goal behind Neighbor Discovery. Network access authentication and access control are outside the scope of this work.

During the discussions while preparing this document, the following aspects that may help to evaluate the eventual solutions were mentioned.

Zero configuration

Interaction with access control solutions

Scalability

Efficiency

However, the main evaluation criteria are formed by the trust models and threat lists. In other words, the solutions are primarily evaluated by seeing how well they secure the networks against the identified threats, and only secondarily from the configuration, access control, scalability, and efficiently point of view.

IMPORTANT. This document occasionally discusses solution proposals, such as Cryptographically Generated Addresses (CGA) [7] and Address Based Keys (ABK) [6]. However, such discussion is solely for illustrative purposes. Its purpose is to give the readers a more concrete idea of *some* possible solutions. Such discussion does NOT indicate any preference on solutions on the behalf of the authors or the working group.

It should be noted that the term "trust" is used in this document in a rather non-technical manner. The most appropriate interpretation is to consider it as an expression of an organizational or collective belief, i.e., an expression of commonly shared beliefs about the future behavior of the other involved parties. Conversely, the term "trust relationship" denotes a mutual a priori relationship between the involved organizations or parties where the parties believe that the other parties will behave correctly even in the future. A trust relationship makes it possible to configure authentication and authorization information between the parties, while the lack of such a relationship makes it impossible to pre-configure such information.

2. Previous Work

The RFCs that specify the IPv6 Neighbor Discovery and Address Autoconfiguration protocols [2] [3] contain the required discussion of security in a Security Considerations section. Some of the threats identified in this document were raised in the original RFCs. The recommended remedy was to secure the involved packets with an IPsec AH [1] header. However, that recommendation oversimplifies the problem by leaving the AH key management for future work. For example, a host attempting to gain access to a Public Access network may or may not have the required IPsec security associations set up with the network. In a roaming (but not necessarily mobile) situation, where a user is currently accessing the network through a service provider different from the home provider, it is not likely that the host will have been preconfigured with the proper mutual trust relationship for the foreign provider's network, allowing it to directly authenticate the network and get itself authenticated.

As of today, any IPsec security association between the host and the last hop routers or other hosts on the link would need to be completely manually preconfigured, since the Neighbor Discovery and Address Autoconfiguration protocols deal to some extent with how a host obtains initial access to a link. Thus, if a security association is required for initial access and the host does not have that association, there is currently no standard way that the host can dynamically configure itself with that association, even if it has the necessary minimum prerequisite keying material. This situation could induce administration hardships when events such as re-keying occur.

In addition, Neighbor Discovery and Address Autoconfiguration use a few fixed multicast addresses plus a range of 16 million "solicited node" multicast addresses. A naive application of pre-configured SAs would require pre-configuring an unmanageable number of SAs on each host and router just in case a given solicited node multicast address is used. Preconfigured SAs are impractical for securing such a large potential address range.

3. Trust Models

When considering various security solutions for the IPv6 Neighbor Discovery (ND) [2], it is important to keep in mind the underlying trust models. The trust models defined in this section are used later in this document, when discussing specific threats.

In the following, the RFC 2461/RFC 2462 mechanisms are loosely divided into two categories: Neighbor Discovery (ND) and Router Discovery (RD). The former denotes operations that do not primarily involve routers while the operations in the latter category do.

Three different trust models are specified:

1. A model where all authenticated nodes trust each other to behave correctly at the IP layer and not to send any ND or RD messages that contain false information. This model is thought to represent a situation where the nodes are under a single administration and form a closed or semi-closed group. A corporate intranet is a good example.

2. A model where there is a router trusted by the other nodes in the network to be a legitimate router that faithfully routes packets between the local network and any connected external networks. Furthermore, the router is trusted to behave correctly at the IP layer and not to send any ND or RD messages that contain false information.

   This model is thought to represent a public network run by an operator. The clients pay to the operator, have its credentials, and trust it to provide the IP forwarding service. The clients do not trust each other to behave correctly; any other client node must be considered able to send falsified ND and RD messages.

3. A model where the nodes do not directly trust each other at the IP layer. This model is considered suitable for e.g., ad hoc networks.

Note that even though the nodes are assumed to trust each other in the first trust model (corporate intranet), it is still desirable to limit the extent of damage a node is able to inflict to the local network if it becomes compromised.

3.1. Corporate Intranet Model

In a corporate intranet or other network where all nodes are under one administrative domain, the nodes may be considered to be reliable at the IP layer. Thus, once a node has been accepted to be a member of the network, it is assumed to behave in a trustworthy manner.

Under this model, if the network is physically secured or if the link layer is cryptographically secured to the extent needed, no other protection is needed for IPv6 ND, as long as none of the nodes become compromised. For example, a wired LAN with 802.1x access control or

a WLAN with 802.11i Robust Security Network (RSN) with AES encryption may be considered secure enough, requiring no further protection under this trust model. On the other hand, ND security would add protection depth even under this model (see below). Furthermore, one should not overestimate the level of security any L2 mechanism is able to provide.

If the network is not physically secured and the link layer does not have cryptographic protection, or if the cryptographic protection is not secure enough (e.g., just 802.1x and not 802.11i in a WLAN), the nodes in the network may be vulnerable to some or all of the threats outlined in Section 4. In such a case some protection is desirable to secure ND. Providing such protection falls within the main initial focus of the SEND working group.

Furthermore, it is desirable to limit the amount of potential damage in the case a node becomes compromised. For example, it might still be acceptable that a compromised node is able to launch a denial-of-service attack, but it is undesirable if it is able to hijack existing connections or establish man-in-the-middle attacks on new connections.

As mentioned in Section 2, one possibility to secure ND would be to use IPsec AH with symmetric shared keys, known by all trusted nodes and by no outsiders. However, none of the currently standardized automatic key distribution mechanisms work right out-of-the-box. For further details, see [8]. Furthermore, using a shared key would not protect against a compromised node.

More specifically, the currently used key agreement protocol, IKE, suffers from a chicken-and-egg problem [8]: one needs an IP address to run IKE, IKE is needed to establish IPsec SAs, and IPsec SAs are required to configure an IP address. Furthermore, there does not seem to be any easy and efficient ways of securing ND with symmetric key cryptography. The required number of security associations would be very large [9].

As an example, one possible approach to overcome this limitation is to use public key cryptography, and to secure ND packets directly with public key signatures.

3.2. Public Wireless Network with an Operator

A scenario where an operator runs a public wireless (or wireline) network, e.g., a WLAN in a hotel, airport, or cafe, has a different trust model. Here the nodes may be assumed to trust the operator to provide the IP forwarding service in a trustworthy manner, and not to disrupt or misdirect the clients' traffic. However, the clients do

not usually trust each other. Typically the router (or routers) fall under one administrative domain, and the client nodes each fall under their own administrative domain.

It is assumed that under this scenario the operator authenticates all the client nodes, or at least requires authorization in the form of a payment. At the same time, the clients must be able to authenticate the router and make sure that it belongs to the trusted operator. Depending on the link-layer authentication protocol and its deployment, the link layer may take care of the mutual authentication. The link-layer authentication protocol may allow the client nodes and the access router to create a security association. Note that there exist authentication protocols, e.g., particular EAP methods, that do not create secure keying material and/or do not allow the client to authenticate the network.

In this scenario, cryptographically securing the link layer does not necessarily block all the threats outlined in Section 4; see the individual threat descriptions. Specifically, even in 802.11i RSN with AES encryption the broadcast and multicast keys are shared between all nodes. Even if the underlying link layer was aware of all the nodes' link-layer addresses, and were able to check that no source addresses were falsified, there would still be vulnerabilities.

One should also note that link-layer security and IP topology do not necessarily match. For example, the wireless access point may not be visible at the IP layer at all. In such a case cryptographic security at the link layer does not provide any security with regard to IP Neighbor Discovery.

There seems to be at least two ways to bring in security into this scenario. One possibility seems to be to enforce strong security between the clients and the access router, and make the access router aware of the IP and link-layer protocol details. That is, the router would check ICMPv6 packet contents, and filter packets that contain information which does not match the network topology. The other possibly acceptable way is to add cryptographic protection to the ICMPv6 packets carrying ND messages.

3.3. Ad Hoc Network

In an ad hoc network, or any network without a trusted operator, none of the nodes trust each other. In a generic case, the nodes meet each other for the first time, and there are no guarantees that the other nodes would behave correctly at the IP layer. They must be considered suspicious to send falsified ND and RD messages.

Since there are no a priori trust relationships, the nodes cannot rely on traditional authentication. That is, the traditional authentication protocols rely on some existing relationship between the parties. The relationship may be direct or indirect. The indirect case relies on one or more trusted third parties, thereby creating a chain of trust relationships between the parties.

In the generic ad hoc network case, there are no trusted third parties, nor do the parties trust each other directly. Thus, the traditional means of first authenticating and then authorizing the users (to use their addresses) do not work.

It is still possible to use self-identifying mechanisms, such as Cryptographically Generated Addresses (CGA) [7]. These allow the nodes to ensure that they are talking to the same nodes (as before) at all times, and that each of the nodes indeed have generated their IP address themselves and not "stolen" someone else's address. It may also be possible to learn the identities of any routers using various kinds of heuristics, such as testing the node's ability to convey cryptographically protected traffic towards a known and trusted node somewhere in the Internet. Methods like these seem to mitigate (but not completely block) some of the attacks outlined in the next section.

4. Threats on a (Public) Multi-Access Link

In this section we discuss threats against the current IPv6 Neighbor Discovery mechanisms, when used in multi-access links. The threats are discussed in the light of the trust models defined in the previous section.

There are three general types of threats:

1. Redirect attacks in which a malicious node redirects packets away from the last hop router or other legitimate receiver to another node on the link.

2. Denial-of-Service (DoS) attacks, in which a malicious node prevents communication between the node under attack and all other nodes, or a specific destination address.

3. Flooding Denial-of-Service (DoS) attacks, in which a malicious node redirects other hosts' traffic to a victim node, and thereby creates a flood of bogus traffic at the victim host.

A redirect attack can be used for DoS purposes by having the node to which the packets were redirected drop the packets, either completely or by selectively forwarding some of them and not others.

The subsections below identify specific threats for IPv6 network access. The threat descriptions are organized in three subsections. We first consider threats that do not involve routers or routing information. We next consider threats that do involve routers or routing information. Finally, we consider replay attacks and threats that are remotely exploitable. All threats are discussed in the light of the trust models.

4.1. Non router/routing related threats

In this section we discuss attacks against "pure" Neighbor Discovery functions, i.e., Neighbor Discovery (ND), Neighbor Unreachability Detection (NUD), and Duplicate Address Detection (DAD) in Address Autoconfiguration.

4.1.1. Neighbor Solicitation/Advertisement Spoofing

Nodes on the link use Neighbor Solicitation and Advertisement messages to create bindings between IP addresses and MAC addresses. More specifically, there are two cases when a node creates neighbor cache entries upon receiving Solicitations:

1. A node receives a Neighbor Solicitation that contains a node's address. The node can use that to populate its neighbor cache. This is basically a performance optimization, and a SHOULD in the base documents.

2. During Duplicate Address Detection (DAD), if a node receives a Neighbor Solicitation for the same address it is soliciting for, the situation is considered a collision, and the node must cease to solicit for the said address.

In contrast to solicitation messages that create or modify state only in these specific occasions, state is usually modified whenever a node receives a solicited-for advertisement message.

An attacking node can cause packets for legitimate nodes, both hosts and routers, to be sent to some other link-layer address. This can be done by either sending a Neighbor Solicitation with a different source link-layer address option, or sending a Neighbor Advertisement with a different target link-layer address option.

The attacks succeed because the Neighbor Cache entry with the new link-layer address overwrites the old. If the spoofed link-layer address is a valid one, as long as the attacker responds to the unicast Neighbor Solicitation messages sent as part of the Neighbor Unreachability Detection, packets will continue to be redirected. This is a redirect/DoS attack.

This mechanism can be used for a DoS attack by specifying an unused link-layer address; however, this DoS attack is of limited duration since after 30-50 seconds (with default timer values) the Neighbor Unreachability Detection mechanism will discard the bad link-layer address and multicast anew to discover the link-layer address. As a consequence, the attacker will need to keep responding with fabricated link-layer addresses if it wants to maintain the attack beyond the timeout.

The threat discussed in this subsection involves Neighbor Solicitation and Neighbor Advertisement messages.

This attack is not a concern if access to the link is restricted to trusted nodes; if a trusted node is compromised, the other nodes are exposed to this threat. In the case where just the operator is trusted, the nodes may rely on the operator to certify the address bindings for other local nodes. From the security point of view, the router may act as a trusted proxy for the other nodes. This assumes that the router can be trusted to represent correctly the other nodes on the link. In the ad hoc network case, and optionally in the other two cases, the nodes may use self certifying techniques (e.g., CGA) to authorize address bindings.

Additionally, some implementations log an error and refuse to accept ND overwrites, instead requiring the old entry to time out first.

4.1.2. Neighbor Unreachability Detection (NUD) failure

Nodes on the link monitor the reachability of local destinations and routers with the Neighbor Unreachability Detection procedure [2]. Normally the nodes rely on upper-layer information to determine whether peer nodes are still reachable. However, if there is a sufficiently long delay on upper-layer traffic, or if the node stops receiving replies from a peer node, the NUD procedure is invoked. The node sends a targeted NS to the peer node. If the peer is still reachable, it will reply with a NA. However, if the soliciting node receives no reply, it tries a few more times, eventually deleting the neighbor cache entry. If needed, this triggers the standard address resolution protocol to learn the new MAC address. No higher level traffic can proceed if this procedure flushes out neighbor cache entries after determining (perhaps incorrectly) that the peer is not reachable.

A malicious node may keep sending fabricated NAs in response to NUD NS messages. Unless the NA messages are somehow protected, the attacker may be able to extend the attack for a long time using this technique. The actual consequences depend on why the node become

unreachable for the first place, and how the target node would behave if it knew that the node has become unreachable. This is a DoS attack.

The threat discussed in this subsection involves Neighbor Solicitation/Advertisement messages.

This attack is not a concern if access to the link is restricted to trusted nodes; if a trusted node is compromised, the other nodes are exposed to this DoS threat. Under the two other trust models, a solution requires that the node performing NUD is able to make a distinction between genuine and fabricated NA responses.

4.1.3. Duplicate Address Detection DoS Attack

In networks where the entering hosts obtain their addresses using stateless address autoconfiguration [3], an attacking node could launch a DoS attack by responding to every duplicate address detection attempt made by an entering host. If the attacker claims the address, then the host will never be able to obtain an address. The attacker can claim the address in two ways: it can either reply with an NS, simulating that it is performing DAD, too, or it can reply with an NA, simulating that it has already taken the address into use. This threat was identified in RFC 2462 [3]. The issue may also be present when other types of address configuration is used, i.e., whenever DAD is invoked prior to actually configuring the suggested address. This is a DoS attack.

The threat discussed in this subsection involves Neighbor Solicitation/Advertisement messages.

This attack is not a concern if access to the link is restricted to trusted nodes; if a trusted node is compromised, the other nodes become exposed to this DoS threat. Under the two other trust models, a solution requires that the node performing DAD is able to verify whether the sender of the NA response is authorized to use the given IP address or not. In the trusted operator case, the operator may act as an authorizer, keeping track of allocated addresses and making sure that no node has allocated more than a few (hundreds of) addresses. On the other hand, it may be detrimental to adopt such a practice, since there may be situations where it is desirable for one node to have a large number of addresses, e.g., creating a separate address per TCP connection, or when running an ND proxy. Thus, it may be inappropriate to suggest that ISPs could control how many addresses a legitimate host can have; the discussion above must be considered only as examples, as stated in the beginning of this document.

In the ad hoc network case one may want to structure the addresses in such a way that self authorization is possible.

4.2. Router/routing involving threats

In this section we consider threats pertinent to router discovery or other router assisted/related mechanisms.

4.2.1. Malicious Last Hop Router

This threat was identified in [5] but was classified as a general IPv6 threat and not specific to Mobile IPv6. It is also identified in RFC 2461 [2]. This threat is a redirect/DoS attack.

An attacking node on the same subnet as a host attempting to discover a legitimate last hop router could masquerade as an IPv6 last hop router by multicasting legitimate-looking IPv6 Router Advertisements or unicasting Router Advertisements in response to multicast Router Advertisement Solicitations from the entering host. If the entering host selects the attacker as its default router, the attacker has the opportunity to siphon off traffic from the host, or mount a man-in-the-middle attack. The attacker could ensure that the entering host selected itself as the default router by multicasting periodic Router Advertisements for the real last hop router having a lifetime of zero. This may spoof the entering host into believing that the real access router is not willing to take any traffic. Once accepted as a legitimate router, the attacker could send Redirect messages to hosts, then disappear, thus covering its tracks.

This threat is partially mitigated in RFC 2462; in Section 5.5.3 of RFC 2462 it is required that if the advertised prefix lifetime is less than 2 hours and less than the stored lifetime, the stored lifetime is not reduced unless the packet was authenticated. However, the default router selection procedure, as defined in Section 6.3.6. of RFC 2461, does not contain such a rule.

The threat discussed in this subsection involves Router Advertisement and Router Advertisement Solicitation messages.

This attack is not a concern if access to the link is restricted to trusted nodes; if a trusted node is compromised, the other nodes are exposed to this threat. However, the threat can be partially mitigated through a number of means, for example, by configuring the nodes to prefer existing routers over new ones. Note that this approach does not necessarily prevent one from introducing new routers into the network, depending on the details of implementation. At minimum, it just makes the existing nodes to prefer the existing routers over the new ones.

In the case of a trusted operator, there must be a means for the nodes to make a distinction between trustworthy routers, run by the operator, and other nodes. There are currently no widely accepted solutions for the ad hoc network case, and the issue remains as a research question.

4.2.2. Default router is 'killed'

In this attack, an attacker 'kills' the default router(s), thereby making the nodes on the link to assume that all nodes are local. In Section 5.2 of RFC 2461 [2] it is stated that "[if] the Default Router List is empty, the sender assumes that the destination is on-link." Thus, if the attacker is able to make a node to believe that there are no default routers on the link, the node will try to send the packets directly, using Neighbor Discovery. After that the attacker can use NS/NA spoofing even against off-link destinations.

There are a few identified ways how an attacker can 'kill' the default router(s). One is to launch a classic DoS attack against the router so that it does not appear responsive any more. The other is to send a spoofed Router Advertisement with a zero Router Lifetime (see Section 6.3.4 of RFC 2461 [2]). However, see also the discussion in Section 4.2.1, above.

This attack is mainly a DoS attack, but it could also be used to redirect traffic to the next better router, which may be the attacker.

The threat discussed in this subsection involves Router Advertisement messages. One variant of this threat may be possible by overloading the router, without using any ND/RD messages.

This attack is not a concern if access to the link is restricted to trusted nodes; if a trusted node is compromised, the other nodes are exposed to this threat. In the case of a trusted operator, there must be a means for the nodes to make a distinction between trustworthy routers, run by the operator, and other nodes. That protects against spoofed Router Advertisements, but it does not protect against router overloading. There are currently no widely accepted solutions for the ad hoc network case, and the issue remains as a research question.

Thanks to Alain Durand for identifying this threat.

4.2.3. Good Router Goes Bad

In this attack, a router that previously was trusted is compromised. The attacks available are the same as those discussed in Section 4.2.1. This is a redirect/DoS attack.

There are currently no known solutions for any of the presented three trust models. On the other hand, on a multi-router link one could imagine a solution involving revocation of router rights. The situation remains as a research question.

4.2.4. Spoofed Redirect Message

The Redirect message can be used to send packets for a given destination to any link-layer address on the link. The attacker uses the link-local address of the current first-hop router in order to send a Redirect message to a legitimate host. Since the host identifies the message by the link-local address as coming from its first hop router, it accepts the Redirect. As long as the attacker responds to Neighbor Unreachability Detection probes to the link-layer address, the Redirect will remain in effect. This is a redirect/DoS attack.

The threat discussed in this subsection involves Redirect messages.

This attack is not a concern if access to the link is restricted to trusted nodes; if a trusted node is compromised, the other nodes are exposed to this threat. In the case of a trusted operator, there must be a means for the nodes to make a distinction between trustworthy routers, run by the operator, and other nodes. There are currently no widely accepted solutions for the ad hoc network case, and the issue remains as a research question.

4.2.5. Bogus On-Link Prefix

An attacking node can send a Router Advertisement message specifying that some prefix of arbitrary length is on-link. If a sending host thinks the prefix is on-link, it will never send a packet for that prefix to the router. Instead, the host will try to perform address resolution by sending Neighbor Solicitations, but the Neighbor Solicitations will not result in a response, denying service to the attacked host. This is a DoS attack.

The attacker can use an arbitrary lifetime on the bogus prefix advertisement. If the lifetime is infinity, the sending host will be denied service until it loses the state in its prefix list e.g., by rebooting, or after the same prefix is advertised with a zero

lifetime. The attack could also be perpetrated selectively for packets destined to a particular prefix by using 128 bit prefixes, i.e., full addresses.

Additionally, the attack may cause a denial-of-service by flooding the routing table of the node. The node would not be able to differentiate between legitimate on-link prefixes and bogus ones when making decisions as to which ones are kept and which are dropped. Inherently, any finite system must have some point at which new received prefixes must be dropped rather than accepted.

This attack can be extended into a redirect attack if the attacker replies to the Neighbor Solicitations with spoofed Neighbor Advertisements, thereby luring the nodes on the link to send the traffic to it or to some other node.

This threat involves Router Advertisement message. The extended attack combines the attack defined in Section 4.1.1 and in this section, and involves Neighbor Solicitation, Neighbor Advertisement, and Router Advertisement messages.

This attack is not a concern if access to the link is restricted to trusted nodes; if a trusted node is compromised, the other nodes are exposed to this threat. In the case of a trusted operator, there must be a means for the nodes to make a distinction between trustworthy routers, run by the operator, and other nodes. There are currently no known solutions for the ad hoc network case, and the issue remains as a research question.

As an example, one possible approach to limiting the damage of this attack is to require advertised on-link prefixes be /64s (otherwise it's easy to advertise something short like 0/0 and this attack is very easy).

4.2.6. Bogus Address Configuration Prefix

An attacking node can send a Router Advertisement message specifying an invalid subnet prefix to be used by a host for address autoconfiguration. A host executing the address autoconfiguration algorithm uses the advertised prefix to construct an address [3], even though that address is not valid for the subnet. As a result, return packets never reach the host because the host's source address is invalid. This is a DoS attack.

This attack has the potential to propagate beyond the immediate attacked host if the attacked host performs a dynamic update to the DNS based on the bogus constructed address. DNS update [4] causes the bogus address to be added to the host's address record in the

DNS. Should this occur, applications performing name resolution through the DNS obtain the bogus address and an attempt to contact the host fails. However, well-written applications will fall back and try the other addresses registered in DNS, which may be correct.

A distributed attacker can make the attack more severe by creating a falsified reverse DNS entry that matches with the dynamic DNS entry created by the target. Consider an attacker who has legitimate access to a prefix <ATTACK_PRFX>, and a target who has an interface ID <TARGET_IID>. The attacker creates a reverse DNS entry for <ATTACK_PRFX>:<TARGET_IID>, pointing to the real domain name of the target, e.g., "secure.target.com". Next the attacker advertises the <ATTACK_PRFX> prefix at the target's link. The target will create an address <ATTACK_PRFX>:<TARGET_IID>, and update its DNS entry so that "secure.target.com" points to <ATTACK_PRFX>:<TARGET_IID>.

At this point "secure.target.com" points to <ATTACK_PRFX>:<TARGET_IID>, and <ATTACK_PRFX>:<TARGET_IID> points to "secure.target.com". This threat is mitigated by the fact that the attacker can be traced since the owner of the <ATTACK_PRFX> is available at the registries.

There is also a related possibility of advertising a target prefix as an autoconfiguration prefix on a busy link, and then have all nodes on this link try to communicate to the external world with this address. If the local router doesn't have ingress filtering on, then the target link may get a large number of replies for those initial communication attempts.

The basic threat discussed in this subsection involves Router Advertisement messages. The extended attack scenarios involve the DNS, too.

This attack is not a concern if access to the link is restricted to trusted nodes; if a trusted node is compromised the other nodes are exposed to this threat. In the case of a trusted operator, there must be a means for the nodes to make a distinction between trustworthy routers, run by the operator, and other nodes. There are currently no known solutions for the ad hoc network case, and the issue remains as a research question.

4.2.7. Parameter Spoofing

IPv6 Router Advertisements contain a few parameters used by hosts when they send packets and to tell hosts whether or not they should perform stateful address configuration [2]. An attacking node could send out a valid-seeming Router Advertisement that duplicates the

Router Advertisement from the legitimate default router, except the included parameters are designed to disrupt legitimate traffic. This is a DoS attack.

Specific attacks include:

1. The attacker includes a Current Hop Limit of one or another small number which the attacker knows will cause legitimate packets to be dropped before they reach their destination.

2. The attacker implements a bogus DHCPv6 server or relay and the 'M' and/or 'O' flag is set, indicating that stateful address configuration and/or stateful configuration of other parameters should be done. The attacker is then in a position to answer the stateful configuration queries of a legitimate host with its own bogus replies.

The threat discussed in this subsection involves Router Advertisement messages.

Note that securing DHCP alone does not resolve this problem. There are two reasons for this. First, the attacker may prevent the node from using DHCP in the first place. Second, depending on the node's local configuration, the attacker may spoof the node to use a less trusted DHCP server. (The latter is a variant of the so called "bidding down" or "down grading" attacks.)

As an example, one possible approach to mitigate this threat is to ignore very small hop limits. The nodes could implement a configurable minimum hop limit, and ignore attempts to set it below said limit.

This attack is not a concern if access to the link is restricted to trusted nodes; if a trusted node is compromised the other nodes are exposed to this treat. In the case of a trusted operator, there must be a means for the nodes to make a distinction between trustworthy routers, run by the operator, and other nodes. There are currently no known solutions for the ad hoc network case, and the issue remains a research question.

4.3. Replay attacks and remotely exploitable attacks

4.3.1. Replay attacks

All Neighbor Discovery and Router Discovery messages are prone to replay attacks. That is, even if they were cryptographically protected so that their contents cannot be forged, an attacker would

be able to capture valid messages and replay them later. Thus, independent on what mechanism is selected to secure the messages, that mechanism must be protected against replay attacks.

Fortunately it is fairly easy to defeat most replay attacks. In request-reply exchanges, such as Solicitation-Advertisement, the request may contain a nonce that must appear also in the reply. Thus, old replies are not valid since they do not contain the right nonce. Correspondingly, stand-alone messages, such as unsolicited Advertisements or Redirect messages, may be protected with timestamps or counters. In practise, roughly synchronized clocks and timestamps seem to work well, since the recipients may keep track of the difference between the clocks of different nodes, and make sure that all new messages are newer than the last seen message.

4.3.2. Neighbor Discovery DoS Attack

In this attack, the attacking node begins fabricating addresses with the subnet prefix and continuously sending packets to them. The last hop router is obligated to resolve these addresses by sending neighbor solicitation packets. A legitimate host attempting to enter the network may not be able to obtain Neighbor Discovery service from the last hop router as it will be already busy with sending other solicitations. This DoS attack is different from the others in that the attacker may be off-link. The resource being attacked in this case is the conceptual neighbor cache, which will be filled with attempts to resolve IPv6 addresses having a valid prefix but invalid suffix. This is a DoS attack.

The threat discussed in this subsection involves Neighbor Solicitation messages.

This attack does not directly involve the trust models presented. However, if access to the link is restricted to registered nodes, and the access router keeps track of nodes that have registered for access on the link, the attack may be trivially plugged. However, no such mechanisms are currently standardized.

In a way, this problem is fairly similar to the TCP SYN flooding problem. For example, rate limiting Neighbor Solicitations, restricting the amount of state reserved for unresolved solicitations, and clever cache management may be applied.

It should be noted that both hosts and routers need to worry about this problem. The router case was discussed above. Hosts are also vulnerable since the neighbor discovery process can potentially be abused by an application that is tricked into sending packets to arbitrary on-link destinations.

4.4. Summary of the attacks

Columns:

N/R	Neighbor Discovery (ND) or Router Discovery (RD) attack
R/D	Redirect/DoS (Redir) or just DoS attack
Msgs	Messages involved in the attack: NA, NS, RA, RS, Redir
1	Present in trust model 1 (corporate intranet)
2	Present in trust model 2 (public operator run network)
3	Present in trust model 3 (ad hoc network)

Symbols in trust model columns:

-	The threat is not present or not a concern.
+	The threat is present and at least one solution is known.
R	The threat is present but solving it is a research problem.

Note that the plus sign '+' in the table does not mean that there is a ready-to-be-applied, standardized solution. If solutions existed, this document would be unnecessary. Instead, it denotes that in the authors' opinion the problem has been solved in principle, and there exists a publication that describes some approach to solve the problem, or a solution may be produced by straightforward application of known research and/or engineering results.

In the other hand, and 'R' indicates that the authors' are not aware of any publication describing a solution to the problem, and cannot at the time of writing think about any simple and easy extension of known research and/or engineering results to solve the problem.

Sec	Attack name	N/R	R/D	Msgs	1	2	3	
4.1.1	NS/NA spoofing	ND	Redir	NA NS	+	+	+	
4.1.2	NUD failure	ND	DoS	NA NS	-	+	+	
4.1.3	DAD DoS	ND	DoS	NA NS	-	+	+	
4.2.1	Malicious router	RD	Redir	RA RS	+	+	R	
4.2.2	Default router killed	RD	Redir	RA	+/R	+/R	R	1)
4.2.3	Good router goes bad	RD	Redir	RA RS	R	R	R	
4.2.4	Spoofed redirect	RD	Redir	Redir	+	+	R	
4.2.5	Bogus on-link prefix	RD	DoS	RA	-	+	R	2)
4.2.6	Bogus address config	RD	DoS	RA	-	+	R	3)
4.2.7	Parameter spoofing	RD	DoS	RA	-	+	R	
4.3.1	Replay attacks	All	Redir	All	+	+	+	
4.3.2	Remote ND DoS	ND	DoS	NS	+	+	+	

Figure 1

1. It is possible to protect the Router Advertisements, thereby closing one variant of this attack. However, closing the other variant (overloading the router) does not seem to be plausible within the scope of this working group.

2. Note that the extended attack defined in Section 4.2.5 combines sending a bogus on-link prefix and performing NS/NA spoofing as per Section 4.1.1. Thus, if the NA/NS exchange is secured, the ability to use Section 4.2.5 for redirect is most probably blocked, too.

3. The bogus DNS registration resulting from blindly registering the new address via DNS update [4] is not considered an ND security issue here. However, it should be noted as a possible vulnerability in implementations.

For a slightly different approach, see also Section 7 in [9]. Especially the table in Section 7.7 of [9] is very good.

5. Security Considerations

This document discusses security threats to network access in IPv6. As such, it is concerned entirely with security.

6. Acknowledgements

Thanks to Alper Yegin of DoCoMo Communications Laboratories USA for identifying the Neighbor Discovery DoS attack. We would also like to thank Tuomas Aura and Michael Roe of Microsoft Research Cambridge as well as Jari Arkko and Vesa-Matti Mantyla of Ericsson Research Nomadiclab for discussing some of the threats with us.

Thanks to Alper Yegin, Pekka Savola, Bill Sommerfeld, Vijay Devaparalli, Dave Thaler, and Alain Durand for their constructive comments.

Thanks to Craig Metz for his numerous very good comments, and especially for more material of implementations that refuse to accept ND overrides, for the bogus on-link prefix threat, and for reminding us about replay attacks.

7. Informative References

[1] Kent, S. and R. Atkinson, "IP Authentication Header", RFC 2402, November 1998.

[2] Narten, T., Nordmark, E. and W. Simpson, "Neighbor Discovery for IP Version 6 (IPv6)", RFC 2461, December 1998.

[3] Thomson, S. and T. Narten, "IPv6 Stateless Address Autoconfiguration", RFC 2462, December 1998.

[4] Wellington, B., "Secure Domain Name System (DNS) Dynamic Update", RFC 3007, November 2000.

[5] Mankin, A., "Threat Models introduced by Mobile IPv6 and Requirements for Security in Mobile IPv6", Work in Progress.

[6] Kempf, J., Gentry, C. and A. Silverberg, "Securing IPv6 Neighbor Discovery Using Address Based Keys (ABKs)", Work in Progress, June 2002.

[7] Roe, M., "Authentication of Mobile IPv6 Binding Updates and Acknowledgments", Work in Progress, March 2002.

[8] Arkko, J., "Effects of ICMPv6 on IKE", Work in Progress, March 2003.

[9] Arkko, J., "Manual Configuration of Security Associations for IPv6 Neighbor Discovery", Work in Progress, March 2003.

Authors' Addresses

Pekka Nikander (editor)
Ericsson Research Nomadic Lab
JORVAS FIN-02420
FINLAND

Phone: +358 9 299 1
EMail: pekka.nikander@nomadiclab.com

James Kempf
DoCoMo USA Labs
181 Metro Drive, Suite 300
San Jose, CA 95110
USA

Phone: +1 408 451 4711
EMail: kempf@docomolabs-usa.com

Erik Nordmark
Sun Microsystems
17 Network Circle
Menlo Park, CA 94043
USA

Phone: +1 650 786 2921
EMail: erik.nordmark@sun.com

Full Copyright Statement

Intellectual Property

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.