Mobile IPv6 Route Optimization Protocol

Aumdevi K. Barbudhe
Assistant Professor & ME Coordinator
Dr.B.N.C.P.E
Dept.of Comp. Sci. & Tech. Yavatmal.

Vishwajit K. Barbudhe
Assistant Professor & ME Coordinator
Jagdambha college of Engineering & Tech.
Dept. of Electronics and Telecomm, Yavatmal

Abstract-

Mobile IPv6 (MIPv6) allows a Mobile Node to talk directly to its peers while retaining the ability to move around and change the currently used IP address. This mode of operation is called Route Optimization (RO), as it allows the packets to traverse a shorter route than the default one through the Home Agent. In Route Optimization, the peer node learns a binding between the Mobile Node’s permanent Home Address and its current temporary Care-of-Address. Once such a binding is in place, the peer node will send all packets whose destination is the Home Address to the Care-of-Address. This is potentially dangerous, since a malicious host might be able to establish false bindings, thereby preventing some packets from reaching their intended destination, diverting some traffic to the attacker, or flooding third parties with unwanted traffic. In this paper we discuss the design rationale behind the MIPv6 Route Optimization Security Design.

Introduction

Mobile IP is based on the idea of providing mobility support on the top of existing IP infrastructure, without requiring any modifications to the routers, the applications or the stationary end hosts. However, in Mobile IPv6 (as opposed to Mobile IPv4) also the stationary end hosts can provide additional support for mobility, i.e., to support route optimization. In route optimization a correspondent node(CN), i.e. a peer for a mobile node, learns a binding between the mobile node’s stationary home address and its current temporary care-of-address. This binding is then used to modify the handling of outgoing packets, leading to security risks. To fully understand the security implications of the design constraints it is necessary to briefly explore the nature of the existing IP infrastructure, the problems Mobile IP aims to solve, and the design principles applied. One of the design goals in the Mobile IP design was to make mobility possible without changing too much. This was especially important for IPv4, with its large installed base, but the same design goal was inherited by Mobile IPv6.

Existing Protocols in Mobile IPV6

1. Triangle Routing

In the basic Mobile IP protocol, IP packets destined to a mobile node that is outside its home network are routed through the home agent. However packets from the mobile node to the correspondent nodes are routed directly. This is known as triangle routing. Figure 1 illustrates triangle routing. This method is inefficient in many cases. Consider the case when the correspondent node and the mobile node are in the same network, but not in the home network of the mobile node. In this case the messages will experience unnecessary delay since they have to be first routed to the home agent that resides in the home network. Triangle routing suffers from a long trip time that affects real time traffic. While forwarding data packets from source to destination triangular path a triangular path has to establish. This causes loss of data packets. One way to improve this is Route
Optimization

Fig 1: Triangle Routing

2. Route Optimization

Route Optimization is an extension proposed to the basic Mobile IP protocol. Route optimization is about routing packets between a mobile node and a correspondent node, using the shortest possible path (as it is normally done between two communicating hosts relying on normal routing). The mobile node is aware when packets are routed through the home agent when it receives tunneled packets addressed to its home address.

Fig 2: Route Optimization

Mobile IPv6 Security Threats:-

This section describes some of the major threats against Mobile IPv6 Route Optimization. A more thorough threat analysis is available in [4]. The goal of an attacker can be to corrupt the correspondent node's binding cache and to cause packets to be delivered to a wrong address. This can compromise secrecy and integrity of communication and cause denial-of-service (DoS) both at the communicating parties and at the address that receives the unwanted packets. The attacker may also exploit features of the Binding Update (BU) protocol to exhaust the resources of the mobile node, the home agent, or the correspondent nodes. It is essential to understand that some of the threats are more serious than others, some can be mitigated but not removed, some threats may represent acceptable risk, and some threats may be considered too expensive to be prevented. Here we consider only active attackers. The rationale behind this is that in order to corrupt the binding cache, the attacker must sooner or later send one or more messages. Thus, it makes little sense to consider attackers that only observe messages but do not send any. In fact, some active attacks are easier, for the average attacker, to launch than a passive one would be. In many active attacks the attacker can initiate the BU protocol execution at any time, while most passive attacks require the attacker to wait for suitable messages to be sent by the targets nodes.

False Binding Update Attacks

Spoofed Binding Updates may be sent to home agents and correspondent nodes. As every IPv6 node is expected to be deployed as a MIPv6 node as well, and every MIPv6 node is to be a Correspondent Node (CN), BU security threats can be seen as applicable to the whole Internet. By spoofing Binding Updates, an attacker can redirect traffic to itself or another node and prevent the original node from receiving traffic destined to it. For example, let us say nodes A and B have been communicating with each other, then, an attacker, node C, sends a spoofed Binding Update packet to node B, claiming to be node A with a care-of-address of node C. This would cause node B to create a binding for node A's CoA and subsequent
further traffic to node C, believing it to be node A’s new care-of-address. Node A would not receive the data it was intended to receive, and, if the data in the packets is not protected cryptographically, node C will be able to see all of node A’s sensitive information.

**Man-in-the-Middle Attack**

It is a form of active eavesdropping in which the attacker makes independent connections with the victims and relays messages between them, making them believe that they are talking directly to each other over a private connection, when in fact the entire conversation is controlled by the attacker. An attacker may also spoof BUs to two corresponding nodes in order to set itself as a Man-in-the-Middle between a MN and a CN. For example, if node A and node B are communicating, the attacker could send both nodes a spoofed Binding Update with the care-of-address set to its own address. This would cause both nodes A and B to send all packets to node C rather than to each other.

**Denial-of-Service Attacks**

Another category of attacks are formed by denial-of-service attacks. The target of a denial-of-service attack can be the mobile node itself, or any other node. In the latter case, a mobile node (or rather a large group of them) is used only as a vehicle of the attack, the actual target being elsewhere. Again, here we consider only some basic attack versions.

1) **Basic Denial-of-Service Attacks:**

By sending spoofed BUs, the attacker could redirect all packets sent between two IP nodes to a random or nonexistent address(es). This way, it might be able to stop or disrupt communication between the nodes. This attack is serious because any Internet node could be targeted, also fixed nodes belonging to the infrastructure (e.g. DNS servers) are vulnerable.

2) **Flooding:**

By sending spoofed BUs, an attacker could redirect traffic to an arbitrary IP address. This could be used to bomb an arbitrary Internet address with excessive amounts of packets. The attacker could also target a network by redirecting data to one or more IP addresses within the network. In the simplest flooding attack, the attacker knows that there is a heavy data stream from node A to B and redirects this to the target address C. However, A would soon stop sending the data because it is not receiving acknowledgments from B. A more sophisticated attacker would act itself as B. It would first subscribe to a data stream (e.g. a video stream) and then redirects this stream to the target address C. The attacker would even be able to spoof the acknowledgements.

**Securing Route Optimization**

The current Mobile IPv6 route optimization security has been carefully designed to prevent or mitigate a number of known threats, some of which were described above. The goal has been to produce a design whose security is close to that of a static IPv4-based Internet, and whose cost in terms of packets, delay and processing is not excessive. The result is not what one would expect; the result is definitely not a traditional cryptographic protocol. Instead, the result relies heavily on the assumption of an uncorrupted routing infrastructure, and builds upon the idea of checking that an alleged mobile node is indeed reachable both through its home address and its care-of-address. Furthermore, the lifetime of the state created at the corresponded nodes is deliberately restricted to a few minutes, in order to limit the potential ability of time shifting.

**Return Routability Procedure**

Return Routability (RR) [1] method was developed to provide adequate authentication between a MN and a CN. First, it ensures that the MN is able to receive messages with its HoA and CoA, after that it protects the binding messages between the MN and the CN. The MN can receive messages with the HoA only if the MN has created a valid binding to the HA in advance. A CN has a private secret key, kcn and a random number, Nj, which it renews at regular intervals [Table 1] (e.g. every few minutes). The CN uses the same kcn and Nj with all the mobiles it is in communication with, so it doesn’t need to generate and store a new Nj
when a new mobile contacts it. Each value of \( N_j \) is identified by the subscript \( j \), which is communicated in the protocol, so when \( N_j \) is replaced by \( N_{j+1} \), the CN can distinguish messages that should be checked against the old random number from messages that should be checked against the new random number. CNs keeps \( N_j \) and a small set of previous values \( N_{j+1}, N_{j-2}, \ldots \) in memory. Older values can be discarded, and messages using them can be rejected as replays. The key \( k_{cn} \) can be either a fixed value or regularly updated. An update of \( k_{cn} \) can be done at the same time as an update of \( N_j \), so that \( j \) identifies both the random number and the key. A correspondent node can generate a fresh \( k_{cn} \) each time that it boots to avoid the need for secure persistent storage for \( k_{cn} \). The RR signaling happens as follows [Figure 2]:

1. \( MN(HoA) \to CN: \text{“HoA”} \)
2. \( MN(CoA) \to CN: \text{“CoA”} \)
3. \( CN \to MN(HoA): \ K_0, j \)
4. \( CN \to MN(CoA): \ K_1, i \)
5. \( MN(CoA) \to CN: BU, K_{bu}, j, i \)
6. \( CN \to MN(CoA): BA, K_{ba} \)
7. \( CN \to MN(HoA): BR, K_{br} \)

The first and the second message are sent concurrently by the MN to the CN to initiate the RR method and they contain only the MN’s HoA and CoA respectively. The first message is sent from the HoA and it is sent via a HA by reverse tunneling the packet first to the HA and then forwarding it to the CN. The second message is sent from the CoA to the CN directly. The third and the fourth messages are sent as responses to the first and the second address respectively. They contain the keys \( K_0 \) and \( K_1 \), which are used for authentication of the binding messages, and also the indices of the used random numbers and private keys. The keys \( K_0 \) and \( K_1 \) are calculated as follows:

\[
K_0 = H_{\text{ecn}}(HoA, N_j, 0) \\
K_1 = H_{\text{ecn}}(CoA, N_i, 1) \\
\]

HMAC SHA1 function is used to calculate the keys by using the CN’s private key \( k_{ecn} \) from the addresses and the current random number. The final ‘0’ or ‘1’ is used to distinguish keys that are calculated from HoA and CoA. The fifth message is the binding update message that is sent by the MN to the CN. It is authenticated by using a secret \( k_{bu} \), which is calculated with the HMAC SHA1 function by using \( km \) as a key from the binding message content. The key \( km \) is calculated with SHA1 function from the keys \( K_0 \) and \( K_1 \). The indexes \( i \) and \( j \) are used to identify the used random numbers and keys.

\[
km = H(K_0, K_1) \\
K_{bu} = H_{km}(CoA, CN, BU, s) \\
\]

The BU contains HoA, sequence number, which is used to prevent replay attacks, and lifetime, which indicates the preferred lifetime for the binding. The lifetime indicates the preferred time before the binding must be refreshed and thus authenticated again. The last number \( s \) is a sequence number that is 0 for the first BU using the key \( km \) and it is incremented in every subsequent BU messages using the same key. It is used to prevent other nodes from learning the CN’s private key \( k_{ecn} \) from the BUs sent by a MN. The sixth and the seventh messages are optional and they are authenticated basically in the same way as the fifth message. The BA contains status of the BU processing, sequence number, which is again used to prevent replay attacks, lifetime and refresh time. Lifetime indicates the granted binding lifetime and the refresh time the recommended time to refresh the binding. BR contains the home address to which the request is issued.

\[
K_{ba} = H_{km}(CoA, CN, BA, s) \\
\]
Kbr = HKm(CoA, BR, s)

The binding lifetime should be adjusted with the renewal period of the CN’s random number Nj and the RR authentication period to prevent other nodes from finding out the private key kcn of the CN. The RR authentication period means the interval in which the CN requires MN to perform RR authentication again.

Creating state safely

The correspondent node may remain stateless until it receives the first Binding Update. That is, it does not need to record receiving and replying to the HoTI and CoTI messages. This helps in potential Denial-of-Service situations: no memory needs to be reserved when processing HoTI and CoTI messages. Furthermore, HoTI and CoTI processing is designed to be lightweight, and it can be rate limited if necessary. When receiving a first binding update, the correspondent node goes through a rather complicated procedure. The purpose of this procedure is to ensure that there is indeed a mobile node that has recently received a HoT and a CoT that were sent to the claimed home and care-of-addresses, respectively, and to make sure that the correspondent node does not unnecessarily spend CPU or other resources while performing this check. Since the correspondent node does not have any state when the BU arrives, the BU itself must contain enough information so that relevant state can be created. Given IP addresses, nonce indices, and the key Kcn, the correspondent node can re-create the home and care-of tokens at the cost of a few memory lookups and two applications of the hash function. Once the correspondent node has re-created the tokens, it hashes the tokens together, giving the key Kbm. This key is then used to verify the MAC that protects integrity and origin of the actual Binding Update. Note that the same Kbm may be used for a while, until either the mobile node moves (and needs to get a new care-of-address token), the care-of token expires, or the home token expires.

Quick expiration of Bindings

A Binding Cache Entry, along the key Kbm, represents the return routability state of the network at the time when the HoT and CoT messages were sent out. Now, it is possible that a specific attacker is able to eavesdrop a HoT message at some point of time but not later. If the HoT had an infinite or a long lifetime, that would allow the attacker to perform a time shifting attack. That is, in the current IPv4 architecture an attacker at the path between the correspondent node and the home agent is able to perform attacks only as long as the attacker is able to eavesdrop (and possibly disrupt) communications on that particular path. A long living HoT, and consequently the ability to send valid binding updates for a long time, would allow the attacker to continue its attack even after the attacker is not any more able to eavesdrop the path. To limit the seriousness of this and other similar timeshifting threats, the validity of the tokens is limited to a few minutes. This effectively limits the validity of the key Kbm and the lifetime of the resulting binding updates and binding cache entries. While short life times are necessary given the other aspect of the security design and the goals, they are clearly detrimental for efficiency and robustness. That is, a HoTI / HoT message pair must be exchanged through the home agent every few minutes. These messages are unnecessary from a pure functional point of view, thereby representing overhead. What is worse, though, is that they make the home agent a single point of failure. That is, if the HoT / HoTI messages were not needed, the existing connections from a mobile node to other nodes could continue even when the home agent fails, but the current design forces the bindings to expire after a few minutes. This concludes our brief walkthrough of the selected security design. The cornerstones of the design were the employment of the return routability idea in the HoT, CoT and binding update messages, the ability to remain stateless until a valid binding update is received, and the limiting of the life times to a few minutes.

Conclusion

The increasing demand for wireless services in recent years is driving the need for a new version of IP that addresses the limitations of the current IP protocol. This new version, called...
IPv6, with its many advantages, including increased address space, address auto-configuration, and integrated IP mobility, is a promising technology to enable the mobile IP world of tomorrow. The transition to IPv6 is now the obvious solution to a growing problem and this transition process has already begun. And, although Mobile IPv6 has recently been slowed down in standardization due to security issues, these issues will have to continue to get attention, get resolved and integrated into the protocol itself, making every device in tomorrow’s Internet, a Mobile IPv6 device, and the Mobile Internet, more efficient, robust, secure.

References:
9. Perkins, C. and Johnson, D. Internet Draft - Route Optimization in Mobile IP.