INTERNET-DRAFT
July 2001
Expires: January 2002

Matthias Frank
Rolf Goepffarth
Wolfgang Hansmann
Ulrich Mueller
University of Bonn

Transmission of native IPv6 over Bluetooth (6overBT)
<draft-hansmann-6overbt-00.txt>

Status of this Memo

Abstract

The Bluetooth technology is intended for a wide range of different user devices and will be employed in various devices such as mobile phones, PDAs and notebooks. To enable communication between these devices and the existing IP world, the transmission of IPv4 and IPv6 packets over Bluetooth has already been addressed in the Bluetooth LAN access profile and the Bluetooth PAN profile.

However, these approaches introduce a considerable amount of protocol overhead. To abolish this deficiency, this document specifies the transmission of "native IPv6 over Bluetooth" (6overBT). The term "native IPv6" refers to the transmission of plain IPv6 packets in L2CAP packets. No additional link layer headers are used.

Table of Contents

1.0 Introduction

The Bluetooth technology is intended for a wide range of different user devices and will be employed in various devices such as mobile phones, PDAs and notebooks. To enable communication between these devices and the existing IP world, the transmission of IPv4 [1] packets over Bluetooth has already been addressed in the Bluetooth LAN access profile [2] and extending this approach to support IPv6 [3] over Bluetooth seems straightforward. However, this approach introduces a considerable amount of protocol overhead and only permits point-point connections between Bluetooth devices.

Alternatively, the Bluetooth PAN profile provides a method for the transmission of IPv4 as well as IPv6 over Bluetooth using the Bluetooth Network Encapsulation Protocol (BNEP). This approach too, results in an unnecessary amount of protocol overhead as it introduces an additional link layer protocol above L2CAP and Baseband.

The purpose of this document is to specify an efficient method of transmitting IPv6 over Bluetooth, which we refer to as "native IPv6 over Bluetooth" (6overBT). The term "native IPv6" refers to the transmission of plain IPv6 packets in L2CAP packets. No additional link layer or encapsulation protocol headers are used.

2.0 6overBT Architecture

In the following the 6overBT architecture is described, which is maintained throughout the present document. It is differentiated between three 6overBT entities, namely the 6overBT mobile device (MD), the 6overBT access router (AR), and the 6overBT IPv6 switch (SW).

The usage scenario with these three entities is depicted in Figure 1. As the use (topology, scheduling, routing) of scatternets is still a field of current research, the 6overBT protocol concepts concentrate only on single piconet Bluetooth scenario. Support for scatternet routing may be addressed in a future specification. Two different kind of scenarios are possible. The scenario where the functionality of the access router and the IPv6 switch is separated into two individual devices, or the scenario where a single device must perform the functionality of the access router and the IPv6 switch.

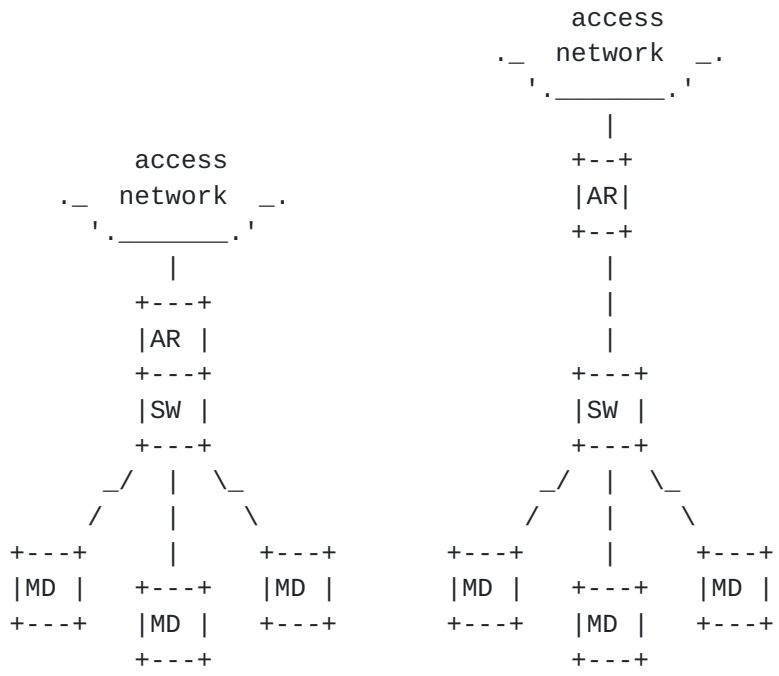

Figure 1: 6overBT usage scenario

2.1 6overBT Mobile Device

In 6overBT, mobile devices participate in a 6overBT scenario by maintaining a single Bluetooth connection to a 6overBT IPv6 switch. The following 6overBT specification enables these mobile devices to communicate with peer IPv6 hosts using native IPv6 over Bluetooth.

2.2 6overBT IPv6 Switch

An IPv6 switch manages IPv6 packet communication for a single piconet. Being a piconet master, the 6overBT IPv6 switch is able to receive and forward data from/to each of its attached slaves, which may be 6overBT mobile devices or 6overBT access routers. Packet switching decisions are based on the destination address of received IPv6 packets. A 6overBT IPv6 switch and a 6overBT access router may be implemented on the same physical entity. It is possible for a switch to function as an IPv6 node, just like a mobile device.

The specification of a switch covers two different usage modes: single-user mode and multi-user mode. In single-user mode, only one mobile device at a time can be connected to the switch. This usage mode requires less Bluetooth functionality and should be implemented, if simultaneous connections to multiple mobile devices are not supported or, a master-slave switch is not available.

2.3 6overBT Access Router

The 6overBT access router is the edge device between an existing IPv6 network and the ad-hoc Bluetooth piconet. Thus, it is typically equipped with at least two network interfaces: a Bluetooth transceiver and a non-Bluetooth interface (e.g. Ethernet). This non-

Bluetooth interface connects to an external IPv6 capable network.

3.0 6overBT Specification

This section presents the specification of the 6overBT protocol concepts of native IPv6 over Bluetooth. After a description of the 6overBT protocol stack, each 6overBT entity is described in an own subsection.

3.1 6overBT Protocol Stack

No additional encapsulation protocol between L2CAP and IPv6 will be used. Figure 2 depicts the necessary protocol stack for the 6overBT specification. 6overBT protocol concepts reside between the Bluetooth protocol stack and IPv6, and handle IPv6 based intra-piconet communication. Therefore, 6overBT appears to IPv6 as a multiple access broadcast layer-2 protocol similar to Ethernet.

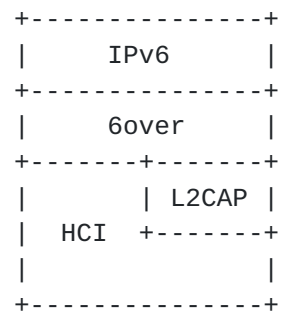

Figure 2: 6overBT protocol stack

3.2 Specification of a 6overBT Mobile Device

3.2.1 Required Bluetooth Functionality

3.2.1.1 Baseband/Link Control/Link Manager

The Baseband and Link Control capabilities required by a 6overBT mobile device are shown in Table 1. The requirement levels used in this and the following tables comply with the requirement levels used in Bluetooth profile documents [4], where 'M' denotes a mandatory feature, 'O' an optional feature, and 'X' an excluded feature. Excluded features are features never to be used in the present or future specifications.

The 6overBT mobile device must be able to establish an ACL link to a 6overBT IPv6 switch. In order to do this, the mobile device must support the inquiry procedure to receive the IPv6 switch's Bluetooth

device address, and the page procedure to establish the definite connection.

In order to detect the loss of a connection, the mobile device must support the link supervision timer. The support for multi-slot packets is recommended, but not necessary for 6overBT.

Capability		Support for a 6overBT mobile device
Link types	ACL links	M
	SCO links	X
Inquiry	Inquiry	M
	Inquiry Scan	O
Paging	Paging	M
	Page scan	O
	Link supervision	M
	Multi-slot packets	O
	Baseband Broadcast	O

Table 1: Baseband and Link Control capabilities needed by a 6overBT mobile device.

Table 2 shows the necessary Link Manager capabilities a 6overBT mobile device has to support. If the 6overBT device connects to an IPv6 switch in multi-user mode it must support the master-slave-switch procedure. However, when connecting to an IPv6 switch in single-user mode, the master-slave-switch procedure is not mandatory anymore.

Other Link Manager procedures such as the power saving modes, authentication, and name requesting are not necessary for the 6overBT specification.

Procedure	Support for a 6overBT mobile device
Power saving modes	O
Authentication / Pairing / Encryption	O
Name request	O
Detach	O
Initiate master-slave switch	X
Perform master-slave switch	M (Switch in multi-user mode) O (Switch in single-user mode)

Table 2: Link manager procedures required by a 6overBT mobile device

3.2.1.2 L2CAP

The L2CAP requirements for a 6overBT mobile device are shown in Table 3. IPv6 packets should be encapsulated in L2CAP packets. For this, a connection-oriented L2CAP channel between the mobile device and the IPv6 switch is required. We strongly recommend reserving a fixed, well known Protocol Switching Multiplexer (PSM) for 6overBT to be used to set up the data connection. This "well-known" PSM should then be used by the mobile device for initiating the channel between the two entities. If a fixed PSM is not available, then an IPv6 switch dependent, dynamic PSM can be used instead. To discover this dynamic PSM, a mobile device has to query the 6overBT SDP service record provided by the IPv6 switch.

Procedure		Support for a 6overBT mobile device
Channel type	Connection-oriented channel	M
	Connectionless channel	O
Signalling	Connection establishment	M
	Connection termination	M
Channel configuration	MTU Configuration	M

	QoS	O

Table 3: L2CAP requirements of a 6overBT mobile device

For the negotiation of the MTU size used between the mobile device and the IPv6 switch, the mobile device must support the L2CAP MTU configuration procedure. Features such as Quality of Service negotiations or the use of a connectionless channel, are not considered in the current 6overBT specification. However, they may be used in order to enhance the functionality in later 6overBT specifications.

3.2.1.3 SDP

As shown in Table 4, a 6overBT mobile device must have SDP functionality in order to communicate with an IPv6 switch acting as a SDP server.

SDP feature	Support for a 6overBT mobile device
SDP client	M
SDP server	M

Table 4: SDP feature requirements for a 6overBT mobile device

3.2.2 Interface Identifier

IPv6 addresses consist of two parts, a prefix and an interface identifier. The interface identifier for 6overBT entities is based on the EUI-64 identifier as in [5] and derives from the 48-bit Bluetooth device address. The interface identifier in 6overBT may be formed in analogy to an EUI-64 identifier deriving from an 48-bit IEEE 802 address (cf. [6]). Alternative interface identifiers should work within 6overBT as well.

3.2.3 Connection State Machine

The state machine for a 6overBT mobile device is fairly simple and consists of two states, the disconnected state and the connected state. In the following two subsections these two states and their task within 6overBT are described in detail.

3.2.4 Disconnected State

3.2.4.1 Baseband / Link Manager Operation

When activating the 6overBT functionality of a mobile device, the device should initially startup in the disconnect state. In this state the mobile device should attempt to establish a baseband connection to a nearby 6overBT IPv6 switch. This should be done by performing an inquiry procedure using the general inquiry access code

(GIAC). After a successful inquiry procedure, the mobile device knows of Bluetooth device addresses of nearby IPv6 switches. This address should then be used in the page procedure performed by the mobile device to create a definite baseband connection to the IPv6 switch.

After a new Baseband connection has been established, the slave should be prepared to perform a master-slave-switch initiated by the IPv6 switch, if the IPv6 switch operates in multi-user-mode.

3.2.4.2 L2CAP Operation

After a baseband connection has been established, the mobile device must initiate a connection-oriented L2CAP channel to the IPv6 switch in order to retrieve the 6overBT IPv6 switch's SDP 6overBT service record. Following this, the L2CAP channel is terminated and a new connection-oriented L2CAP channel for the PSM, as indicated in the retrieved service record, is established. During the channel configuration phase, the mobile device and the IPv6 switch should negotiate the L2CAP specific parameters such as MTU size. For more detailed information on the L2CAP operation refer to section 3.3.5.2.

3.2.5 Connected State

After a baseband and L2CAP connection have successfully been established, the mobile device transits from the disconnected state to the connected state. Now, the mobile device has to assign an IPv6 address to its Bluetooth transceiver. For this the mobile should use the information previously retrieved in the 6overBT SDP service record.

Alternatively, the mobile device may wait for the next ICMPv6 router advertisement which is sent periodically by the 6overBT access router, or may prompt the access router to send an advertisement by sending an ICMPv6 router solicitation message. When sending a solicitation message the 6overBT mobile device may use either its link-local address, or the unspecified address :: as the source address.

If no router is available, that is to say, if the mobile device does not receive any router advertisement, it may use its link local address for further communication. If the mobile device receives a router advertisement, the advertisement should contain the prefix information option containing the valid network prefixes that a mobile device may use. Such a prefix should be used by the mobile device when forming the IPv6 address it wants to assign to its Bluetooth interface. We strongly recommend that the IPv6 address is assembled by concatenating the received network prefix with the Bluetooth devices EUI-64 interface identifier. However, the concepts presented in this specification support also other methods of generating the interface identifier, e.g. for security purposes, if it is desired to hide the Bluetooth device address to the outside world. To prevent duplicate address assignment, neighbor discovery's duplicate address detection procedure must be used before assigning an IPv6 address to an interface.

For this, the mobile device must send an ICMPv6 neighbor solicitation

message to the solicited-node-multicast address, asking whether some other host already owns its link-local address. The solicitation message in the duplicate address detection procedure should use the source address of ::, to distinguish these solicitation messages from normal neighbor solicitations. If the mobile device does not hear a

response to its solicitation message, then no duplicate address exists and the device may use the corresponding IPv6 address in further operation. To detect the loss of connectivity to the IPv6 switch, link supervision should be used. The link supervision timer is reset with every received baseband packet. If the supervision timer expires, then the link is regarded as broken and the mobile device should change to the disconnected state again.

3.3 Specification of a 6overBT IPv6 Switch

The following section specifies the operation of a 6overBT IPv6 switch. A 6overBT IPv6 switch is an IPv6 interconnection entity for 6overBT mobile devices. It is the core entity in 6overBT, which must be always present for IPv6 over Bluetooth communication.

3.3.1 Required Bluetooth Functionality

As required, a 6overBT switch must also be able to act as a 6overBT mobile device. The Bluetooth requirements are more restrictive and form a superset of the functionality required by 6overBT mobile devices. The following section specifies Bluetooth requirements for a 6overBT switch.

3.3.1.1 Baseband/Link Control/Link Manager

A 6overBT IPv6 switch must be able to accept baseband ACL links from mobile devices. The ability to actively page other Bluetooth devices is only required when acting as 6overBT mobile device. If running in multi-user mode, a 6overBT switch must be able to maintain several simultaneous connections to slave devices. In order to permit the connection of mobile devices, inquiry scans must be performed regularly. However, initiating an inquiry procedure is only required when acting as 6overBT mobile device. Link supervision must be implemented to detect the loss of a connection. Support for multi-slot packets is not required, but strongly recommended. The current 6overBT specification does not require native baseband broadcast. However, in order to increase multicast performance, later enhancements to the 6overBT specification, as proposed in section 4.1, could benefit from broadcast support.

Table 5 summarizes the required Bluetooth baseband and Link Control functionality for a 6overBT IPv6 switch.

Capability		Support for a 6overBT switch
Link types	ACL links	M
	SCO links	X
Inquiry	Inquiry	M
	Inquiry Scan	M
Paging	Paging	M
	Page scan	M
	Link supervision	M
	Multi-slot packets	O
	Baseband Broadcast	O

Table 5: Bluetooth link control / Baseband capabilities needed by a 6overBT switch.

Table 6 lists necessary link controller procedures for a 6overBT switch. The support of power saving modes, as SNIFF, PARK or HOLD mode is not required by 6overBT. A crucial requirement for 6overBT in multi-user mode is the support of master-slave switch. A 6overBT switch running in multi-user mode must be able to initiate and perform a master-slave switch.

Procedure	Support for a 6overBT switch
Power saving modes	O
Authentication / Pairing / Encryption	O
Name request	O
Detach	O
Initiate master-slave switch	M (multi-user mode) O (single-user mode)
Perform master-slave switch	M (multi-user mode) O (single-user mode)

+------------------------------+------------------------------+

Table 6: Link manager procedures required by a 6overBT switch

The 6overBT specification does not cover link manager based configuration for authentication and link encryption. However, later specifications could benefit from the presence of this functionality.

3.3.1.2 L2CAP

L2CAP connection-oriented channels will be used for service discovery [7] and IPv6 packet transfer. Table 7 summarizes 6overBT requirements to L2CAP. A 6overBT IPv6 switch must be able to accept L2CAP channel requests by mobile devices. The L2CAP implementation should be able to open L2CAP channels on already established HCI connection handles or provide HCI connection handles from already opened L2CAP channels

For 6overBT, the reservation of a "well known" PSM is recommended to be used by mobile devices to open L2CAP channels for 6overBT controlled IPv6 communication. However, in absence of a global PSM, a dynamical PSM can be configured individually by the SDP 6overBT service record provided by the 6overBT switch.

Procedure		Support for a 6overBT switch
Channel type	Connection-oriented channel	M
	Connectionless channel	O
Signalling	Connection establishment	M
	Connection termination	M
Channel configuration	MTU Configuration	M
	QoS	O

Table 7: L2CAP requirements of a 6overBT switch

3.3.1.3 SDP

An IPv6 switch must have both SDP client and SDP server functionality. However, if used as a 6overBT mobile device, only SDP client functionality will be used.

SDP feature	Support for a 6overBT switch
SDP client	M
SDP server	M

Table 8: SDP feature requirements for a 6over switch

3.3.2 General Operation

3.3.2.1 Advertising

A switch must advertise its presence by regular inquiry scans for the GIAC. In the advertised device class field sent to devices performing inquiry a 6overBT switch should report itself as Networking/LAN access router. As specified in [8] the minor device field should represent the current switch load, specifying the current number of connected slaves.

3.3.2.2 Service Discovery

Apart from the information given in the device class field passed during inquiry, a switch should unambiguously denote its service by providing a SDP service record derived from the 6overBT service class. The information provided by this service record is described in section 3.5.

3.3.3 Binding Records

Fundamental data structures operated by the switch are unicast binding records. A binding record stores all known IPv6 addresses, including the link-local address, which can be associated with a single mobile device. For each connection to a mobile device, a binding record is constructed and updated on each ICMPv6 neighbor advertisement message received by this mobile device. Binding records represent soft state, which must be regularly updated in order to prevent it from timeout.

Once the connection to the mobile device is lost, the associated binding record still persists until the lifetime of all of its advertised IPv6 addresses expire. Keeping the address state for a short while after connection loss speeds up the time in which a mobile device is able to fully participate again on reconnection to the same IPv6 switch (e.g. after recovering from a previous established link break down).

Figure 3 shows a unicast binding record. A single Bluetooth Device

Address (BD_ADDR) is associated with n > 0 IPv6 unicast addresses.
Each IPv6 unicast address has its own timer.

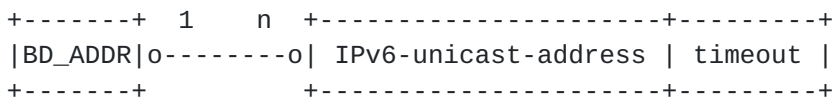

Figure 3: unicast binding record

Binding records only contain information on unicast address mappings. To store information about multicast address bindings, a different data structure is used, the multicast binding record. In a multicast binding record, an IPv6 multicast address is mapped to a list of BD_ADDRs with mobile devices subscribed to a particular multicast group. Information on multicast group memberships are learned from ICMPv6 Multicast Listener Discovery (MLD) messages received by mobile devices. Unlike unicast binding records, information on multicast group membership of a particular mobile device is discarded on connection loss.

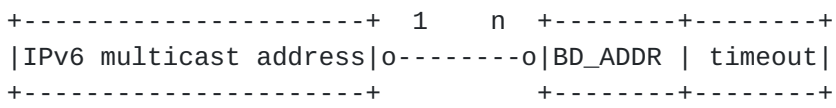

Figure 4: multicast binding record

A multicast binding record as shown in figure 4 is associated with n, n > 0 BD_ADDRs. Each BD_ADDR has its individual timeout after which its association to the multicast address will expire.

3.3.4 Connection State Machine

Figure 5 shows the state diagram of the connection to a single mobile device. There are four different states of a connection specified: unbound/disconnected, unbound/connected, bound/ connected and bound/disconnected. Except for the unbound/disconnected state, a binding record is always associated with a connection.

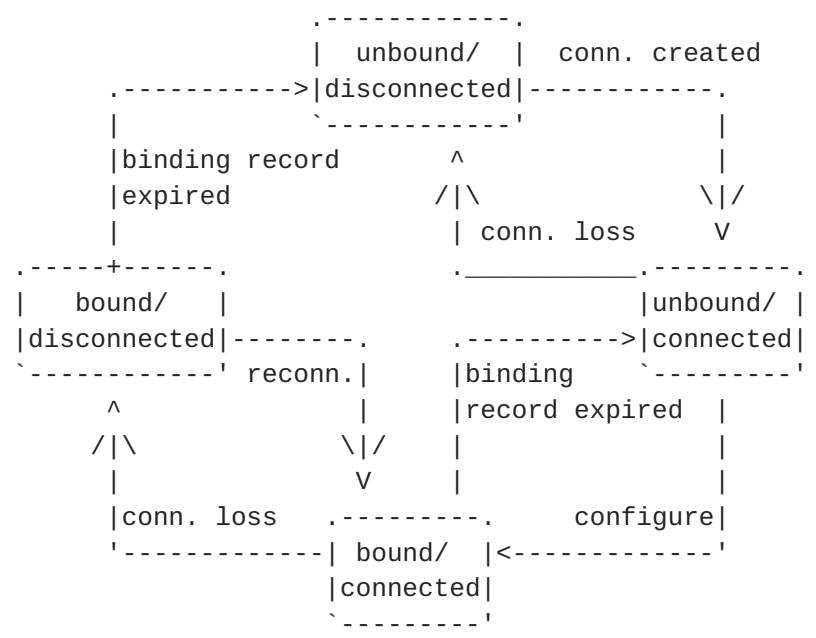

Figure 5: 6overBT switch binding record state machine

3.3.5 Disconnected State

3.3.5.1 Baseband / Link Manager Operation

A switch should perform regular page scans in order to accept baseband connection attempts by mobile devices. After a new connection has been established, the switch must initialize a master-slave-switch in order to join the mobile device as slave into its piconet. This applies only, if the switch is configured to run in multi-user mode. Running in single-user mode does not require a master-slave-switch.

3.3.5.2 L2CAP Operation

After baseband link establishment, a 6overBT switch expects a L2CAP channel request initiated by the mobile device for the PSM, which is advertised in the 6overBT service record. During L2CAP connection configuration, the switch should notify a link MTU of at least 1280 octets. The switch must be able to handle L2CAP datagrams with a payload of at least 1280 octets. However, if information on the actual external link MTU is known (e.g. as given in the MTU option of router advertisements sent by access routers), this MTU value should be used instead to configure the L2CAP link. Both peers must use the default Flush timeout value in order to provide a reliable L2CAP channel. On successful channel configuration, the switch should create an empty unicast binding record for the mobile device and move the connection into connected/unbound state. If the IPv6 switch does

not receive a L2CAP connection attempt to the PSM used by 6overBT within a TBD amount of time, the baseband connection should be released.

3.3.6 Unbound/Connected State

Connections in unbound/connected state represent links to mobile devices to which no BD_ADDR to IPv6 interface address mapping already exists. Though mobile devices are able to send packets, the switch is only able to broadcast packets destined to an unknown IPv6 address to all connected mobile devices. After at least one single IPv6 address has been associated with a mobile device, the connection changes to bound/connected state. If the connection is lost, the empty unicast binding record is discarded.

3.3.7 Bound/Connected State

For connections in bound/connected state at least one IPv6 address of the mobile device is known. This is the default state for active connections. Packets determined to mobile devices, which connections are in bound/connected state can be forwarded without broadcasting. If all known IPv6 addresses of a connected mobile device have expired, the connection changes to unbound/connected state.

If the physical loss of the connection to a mobile device has been signalled, its unicast binding record changes to bound/disconnect state.

3.3.8 Bound/Disconnected State

In the bound/disconnected state, the switch has lost the physical Bluetooth connection to a mobile device. However, information of associated IPv6 addresses is still maintained in a binding record until the addresses expire.

If a mobile device reconnects to the switch and a binding record is still maintained, the baseband link setup step as described in section 3.3.5 is performed and the connection changes back into bound/connected state.

IPv6 packets destined to mobile devices which are not connected to the switch but still have a binding record should be silently discarded. If all IPv6 addresses in a binding record have expired, the binding record is released.

3.3.9 IPv6 Packet Switching

The following packet switching algorithm is performed each time the switch receives an IPv6 packet from any connection to a mobile device either in unbound/connected or bound/connected state.

Switching decision is solely based on the IPv6 destination address of the received packet. A different switching strategy is used for unicast and multicast destination addresses.

3.3.9.1 Evaluation of ICMPv6 Messages

A switch must evaluate IPv6 neighbor advertisement messages, router

advertisements and MLD messages intercepted from mobile devices. After evaluation, these packets should be forwarded as any other packet. In the following the special treatment of these messages is depicted.

Router advertisements

A switch must intercept and evaluate router advertisements received by access routers. For each access router, it should store the information given in the latest router advertisement. This information should be deleted after its lifetime expires, which is given in the router advertisement.

One of the connected access routers should be elected as default access router. How to determine this access router is still to be decided. A possibility is to use the access router with the lowest BD_ADDR. Other solutions may take access router capacity into account. The appropriate solution is still TBD.

Neighbor advertisements

Neighbor advertisements update binding records. If a new address is learned from a mobile device, this address should be added to its binding record, otherwise, the lifetime of the address should be updated. Expired IPv6 unicast addresses should be removed from unicast binding records.

The lifetime of an IPv6 address in a binding record should be according to the "Reachable Time" value given in the router advertisement of the access router, which advertised network prefix was used to construct the address. If no router advertisements are available, or an unspecified value is given in the 'Reachable Time' field of the router advertisement, a timeout value of 45 (MAX_RANDOM_FACTOR * REACHABLE_TIME) seconds should be assumed. According to [9], this is the longest time a node can use information from the neighbor cache of an unresponsive target node without performing neighbor unreachability detection (sending neighbor solicitation and expecting a neighbor advertisement).

> [FIXME: according to our spec the Binding Record of a host times out 45 seconds after the last NA message has passed the switch. However, if a host receives reachability confirmation by other means than NUD (e.g. by progress detection) our Binding Record times out. Should actual usage of an address reset the Binding Cache timeout, too?]

Multicast Listener Discovery messages

An IPv6 switch should evaluate Multicast Listener Discovery messages sent by mobile devices in order to learn about multicast group mem-

berships. Joining a multicast group, a mobile device sends an unsolicited Multicast Listener Report to the all router multicast group of the link. Proper knowledge of group memberships reduces unnecessary forwarding of multicast packets to unsubscribed mobile devices.

On reception of a Multicast Listener Report message, a switch must check, if an according multicast binding record for the advertised multicast address already exists. If yes, the BD_ADDR of the sending device is added to the multicast binding record. Otherwise, a new multicast binding record is created with the BD_ADDR of the mobile device as single listener. The multicast binding information should time out after 260 seconds as suggested by MLD (cf. [10]).

If a Multicast Listener Done message is received from a mobile device, its binding must be immediately removed from the multicast address' multicast binding record. If no mobile devices are registered listeners on a multicast group, the multicast binding record must be discarded.

After evaluation, Multicast Listener Discovery messages must be forwarded like normal multicast packets.

3.3.9.2 Unicast Switching

Receiving a packet, the switch has to decide, if the destination IPv6 address is on-link, or not. A destination is considered "on-link", if the network prefix matches any of the network prefixes advertised in the intercepted router advertisements. If no router advertisements are available, the destination should be regarded on-link. If a destination is not on-link, the packet should be forwarded to the access router, which has advertised the network prefix, from which the source address in the packet has been generated. If several access routers have advertised the same prefix, an arbitrary access router should be chosen.

If the destination address has an EUI-64 identifier (contains the BD_ADDR of the destination device), the datagram should be directly forwarded to the connected mobile device. No binding record lookup is necessary for such destination addresses.

Otherwise, if a destination is on-link, the associated unicast binding record is looked up and the packet should be forwarded to the appropriate mobile device. If no binding record is found for the destination, the packet should be forwarded to all links maintained to mobile devices.

3.3.9.3 Multicast Switching

For forwarding of packets destined to a multicast group the following algorithm applies:

> If the destination multicast group has either global, site-local or organizational local scope AND the destination multicast address matches an existing Multicast Binding Record, THEN a copy

of the packet should be forwarded through each L2CAP channel given in the Multicast Binding Record.

If the IPv6 destination address is the solicited node multicast

address (FF02::1:FFxx:yyzz), the packet should be forwarded to all mobile devices with matching interface identifier.

If according to these rules no destination has already been found, the multicast packet must be forwarded to all mobile devices.

3.3.10 IPv6 Host Capability

An IPv6 switch may be configured to operate as an IPv6 capable node just like mobile devices. In this case, a binding record containing the IPv6 switch's own BD_ADDR address must be maintained permanently. As information on the used IPv6 addresses is locally available, the switch may employ means other than intercepting its own neighbor advertisements and MLD messages to learn about its IPv6 addresses and multicast group memberships.

For the local IPv6 stack the 6overBT IPv6 switch should appear as a network interface, to which packets can be forwarded. Packets intercepted from the local IPv6 stack should be routed, as if a mobile device would have received them.

3.4 Specification of a 6overBT Access Router

6overBT access routers are entities equipped with Bluetooth transceivers and at least another non-Bluetooth link-layer technology over which IPv6 based communication can take place. Access routers may be either fixed or mobile. The 6overBT specification for access routers is quite similar to the specification of a 6overBT mobile device - an access router is a 6overBT mobile device that operates an IPv6 stack configured as router. Optionally, an IPv6 access router can be implemented along with a 6overBT IPv6 switch on the same physical entity.

3.4.1 Required Bluetooth Functionality

Implemented as a single entity, a 6overBT access router has the same Bluetooth requirements as a 6overBT mobile device. Combined with a 6overBT IPv6 switch, an access router has the same Bluetooth requirements as a 6overBT IPv6 switch.

3.4.2 General Operation

3.4.2.1 Inquiry

If implemented as a separate entity, an access router could be restricted to connect only to an administratively determined set of known 6overBT IPv6 switches. The list of IPv6 switches allowed to connect to should be configurable. For this, Bluetooth specified means for authentication and security could be incorporated. Details for appropriate means are not within the scope of the current docu-

ment.

3.4.3 IPv6 Requirements

The IPv6 stack on 6overBT access routers should be properly configured as router. The Bluetooth link presented by 6overBT should appear as a network interface for a broadcast multiple access link (e.g. Ethernet). At least one global scope or site local network prefix should be manually configured to the 6overBT link. The interface identifier for this link must be constructed according to the EUI-64 identifier based on the 48-bit BD_ADDR, as described in section 3.2.2. Other methods for building an interface identifier for access routers are not covered by this specification.

The IPv6 stack must advertise the network prefix in router advertisements. In router advertisements, the 'M' and 'O' flags should be set to zero in order to denote, that no stateful address configuration will take place for configuring a mobile device's IPv6 address. The advertised network prefixes must have the 'L' and 'A' flags set to one, indicating that the network prefixes should be used for on-link address determination and autonomous address configuration.

Each router advertisement should contain a MTU Option, in which the recommended MTU of the 6overBT link is advertised. The link MTU should be the same MTU used on the access router's non-Bluetooth link, or 1280 octets, which ever is larger.

3.5 Service Records

A 6overBT switch must provide a SDP service record of the 6overBT service class to advertise its service. The information items of that service record are summarized in Table 9. In the Protocol descriptor list 6overBT services are denoted to be built on top on L2CAP. Which PSM actually should be used for L2CAP channel establishment is given as "SpecificParameter0" of the L2CAP protocol description.

The service record also provides information that could be obtained by router advertisements, once a 6overBT mobile device is connected to the switch. Providing information on IPv6 network prefixes at this early stage permits a 6overBT mobile device to configure its IPv6 address(es) without sending a router solicitation and expecting subsequent router advertisement by 6overBT access routers.

The 6overBT_SwitchLoad information item reveals the load situation on the switch in terms of the number of currently connected mobile devices. The 6overBT_SessionDescriptionString information item provides an identification useful to differentiate between 6overBT sessions offered by other 6overBT switches.

Item	Type	Description
ServiceRecordHandle		
ServiceClassIDList		
ServiceClass0	UUID	UUID of 6overBT service class
ProtocolDescriptorList		
Protocol0	UUID	UUID of L2CAP
SpecificParameter0	UInt16	L2CAP PSM for 6overBT
Protocol1	UInt16	UUID for 6overBT
SpecificParameter0	UInt16	Version 1.0 (0x100)
6OVER_NetworkPrefixList		List of network prefixes for the 6overBT piconet spanned by this switch. The list of prefixes is learned from Router Advertisements sent by access routers
Prefix0	UInt128	Network prefix 0
PrefixLength0	UInt16	Network prefix length
...		
6OVER_SwitchLoad	UInt16	Number of 6overBT mobile devices connected to this switch
6OVER_SessionDescr	String	Human readable string describing the switch's service. Useful to distinguish between the services of several available 6overBT switches at the same place

Table 9: Information elements in a 6overBT service record

3.6 Multiple Switches in a Piconet

It is likely that several devices with 6overBT switch capability attempt to participate in a 6overBT piconet. In this case, one device must be selected as 6overBT switch while the other devices take over the role as 6overBT mobile device. The choice of which device to use as 6overBT switch in this situation should be user configurable.

4.0 Security Considerations

6overBT makes excessive use of Neighbor Discovery and thus shares ND's security problems in case of unauthenticated ND messages.

The current version of the draft does not address the setup of Bluetooth link encryption. A future version of the draft should reflect that.

No access control has been specified so far. Mobile devices can connect at their will to a 6overBT switch. This issue must be addressed in a future version of this draft.

5.0 Summary

With 6overBT we have specified a mechanism for efficient transport of IPv6 over Bluetooth. IPv6 packets are transported in the payload of L2CAP with no additional protocol overhead. Being a Bluetooth master device a 6overBT switch is able to provide connectivity for up to 7 slave devices. 6overBT emulates a broadcast access environment using the 6overBT switch as the basic means to organize communication. IP switching is performed at the switch and is based on information gained from intercepted ND messages by the slave devices. This way, the Bluetooth device appears to the IP stack at the switch as single network interface providing capabilities similar to Ethernet.

6.0 References

[1] J. Postel, "Internet Protocol", STD 5, RFC 791, IETF, September 1981.

[2] Specification of the Bluetooth System, Version 1.0 B, Profiles, Part K: 9, "LAN Access Profile", December 1999.

[3] S. Deering, R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, IETF, December 1998.

[4] Specification of the Bluetooth System, Version 1.0 B, Profiles, December 1999.

[5] R. Hinden, S. Deering, "IP Version 6 Addressing Architecture", RFC 2373, IETF, July 1998.

[6] M. Crawford, "Transmission of IPv6 Packets over Ethernet Network", RFC 2464, December 1998

[7] Specification of the Bluetooth System, Version 1.0 B, Profiles, Part K:2, "Service Discovery Application Profile", December 1999.

[8] Specification of the Bluetooth System, Version 1.0 B, Core, December 1999.

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

[10] S. Deering, W. Fenner, B. Haberman, "Multicast Listener Discovery (MLD) for IPv6", RFC 2710, IETF, October 1999.

7.0 Author's Addresses

Matthias Frank
University of Bonn
Institute of Computer Science IV
Roemerstr. 164 Phone: +49 228 73 4550
D-53117 Bonn, Germany Email: matthew@cs.uni-bonn.de

Rolf G. Goepffarth
University of Bonn
Institute of Computer Science IV
Roemerstr. 164 Phone: +49 228 73 4549
D-53117 Bonn, Germany Email: rgg@cs.uni-bonn.de

Wolfgang Hansmann
University of Bonn
Institute of Computer Science IV
Roemerstr. 164 Phone: +49 228 73 4549
D-53117 Bonn, Germany Email: hansmann@cs.uni-bonn.de

Ulrich Mueller
University of Bonn
Institute of Computer Science IV
Roemerstr. 164
D-53117 Bonn, Germany Email: muelleru@cs.uni-bonn.de