Università degli Studi di Lecce

FACOLTÀ DI INGEGNERIA
CORSO DI LAUREA IN INGEGNERIA INFORMATICA

TESI DI LAUREA
IN
SISTEMI OPERATIVI

PERFORMANCE DEL PROTOCOLLO MOBILE IPV6

RELATORE:
CHIAR.MO PROF. FRANCO TOMMASI

CORRELATORE:
ING. SIMONE MOLENDINI

CONTRORELATORE:
PROF. MARIO DE BLASI

LAUREANDO:
PAOLO CAVONE

ANNO ACCADEMICO 2004-2005

ai miei Genitori

Sommario

1 - Introduzione

Il *wireless*, ovvero la connettività senza fili, sta vivendo uno sviluppo sorprendente: ovunque se ne parla e il grande successo è senza dubbio dovuto alla facilità con cui è possibile realizzare una Wireless Local Area Network (WLAN) e al costo molto contenuto di schede e stazioni base.

Di conseguenza i dispositivi (portatili, palmari, telefoni cellulari, automobili, etc) e le applicazioni (VoIP fra tutte) "mobili" si stanno sempre più diversificando e diffondendo. Come spesso accade, quando una tecnologia (IEEE 802.11) investe un numero sempre crescente di utenti, nuove esigenze aprono nuovi scenari di ricerca.

La mobilità su Internet sarà il fenomeno che caratterizzerà il networking del prossimo futuro: un utente può essere collegato alla Rete usando WLAN ma può nel contempo muoversi, cambiare il suo punto di collegamento e mantenere eventuali connessioni pre-esistenti.

Per questo motivo, l'Internet Engineering Task Force ha progettato il protocollo Mobile-IP, che, insieme con WLAN, fornisce la *mobilità* ad Internet.

In questa tesi analizzeremo le performance di Mobile-IP utilizzando la nuova versione del protocollo su cui si fonda la grande rete: IPv6.

Internet protocol Version 6 nasce già di per sè orientato alla mobilità infatti, grazie all' enorme spazio di indirizzamento ed all'autoconfigurazione di rete *state-less* (ovvero senza DHCP) è possibile che un nodo wireless aquisisca le informazioni, a livello di rete, necessarie per "adattarsi" all'attuale WLAN a cui è connesso a livello 2 (link layer) .

La versione attuale (ovvero quella più utilizzata) di IP (Ipv4) assume implicitamente che un nodo abbia sempre lo stesso punto di connessione ad Internet, in altri termini l'indirizzo IP identifica univocamente il punto di "attracco" su cui il nodo risiede.

Nel momento in cui si "stacca" un dispositivo mobile da Internet per collegarlo altrove questo, non potrebbe continuare la comunicazione fino a che non si riconfiguri il sistema con un nuovo indirizzo IP, subnet mask e default gateway.

Se un nodo mobile (MN) si muove senza cambiare la sua configurazione di rete, eventuali connessioni TCP/IP in status "*established*" andranno perse. I protocolli attuali di routing non possono perciò inoltrare correttamente i datagrammi IP.

Gli indirizzi IP definiscono, cioè, il rapporto topologico fra gli host collegati in rete.

Per sostenere i dispositivi mobili, che cambiano dinamicamente i loro punti di accesso al Internet, l'Internet Engineering Task Force (IETF) ha sviluppato un protocollo ad-hoc: Mobile IP (MIP).

A seconda della versione del protocollo IP sono state sviluppate due variazioni di Mobile-IP:

- Mobile IPv4: basato su IPv4
- Mobile IPv6: basato su IPv6

1.1 Obiettivo della Tesi

Gli **obiettivi** principali di questa Tesi sono:

- L'analisi delle **performance del protocollo Mobile-IPv6 (MIPv6),** in particolare della implementazione per Linux sviluppata presso l'Helsinky University of Tecnology, basandosi sulle pubblicazioni (*Request For Comments*) dei seguenti gruppi di lavoro IETF:
  - MIP6WG Mobility for IPv6 Working Group:
    - *RFC 3775, "Mobility Support in IPv6"*
  - BMWG Benchmarking Methodology Working Group:
    - *RFC 2544, "Benchmarking Methodology for Network Interconnect Devices"*
- L' analisi dell'***handoff a livello 3***, ovvero il processo di aggiornamento della configurazione di rete (IPv6) di un nodo mobile (MN) a seguito dell'intercettazione di una nuova rete wireless *MIPv6 enabled*, sulla base delle messaggistiche e relative temporizzazioni.

1.2 Struttura della Tesi

- Nel Cap. sequente vengono esposte tutte le caratteristiche native di IPv6 e che interoperano con Mobile IPv6, le principali sono: lo spazio e le tipologie di indirizzamento IPv6, l'autoconfigurazione di rete stateless, la procedura di rilevamento di infirizzi duplicati (DAD), le estensioni dell'Header IPv6, Ipsec.
- Nel Cap.3 viene esposto nel dettaglio Mobile IPv6: l'attenzione è stata posta sugli algoritmi (Binding De/Registration, Return Routabilty Procedure, Dynamic Home Agent Discovery, Correspondent Registration, etcc) e le strutture dati (Binding Cache, Mobility Header) del protocollo. Alla fine è riportato una tabella per il confronto delle implementazioni attualmente disponibili.
- L'analisi svolta è stata condotta secondo due direttive: da un lato (Cap.4) le indicazioni riportate nel RFC2544 (*Benchmarking Methodology for Network Interconnect Devices)* hanno permesso di testare il protocollo attivamente ed analizzare le performance in termini di Throughput, Frame Loss Rate, System Recovey Time, e Burst size.
- Una seconda metodologia seguita, di seguito indicata come *analisi passiva* (Cap.5) si basa sulla misura dei tempi delle messaggistiche utilizzate da Mobile IPv6 col fine di individuare eventuali colli di bottiglia durante l'handoff. A tal proposito è stata sviluppata una applicazione web che consente di rilevare le tempistiche a partire dai Log di Tcpdump sulla base delle indicazioni riportate nel *TAHI conformance test suite for Mobile IPv6* (www.tahi.org).

- Nei Cap.6 e 7 sono riportati i risultati di queste due metodologie di analisi da cui si evincono i benefici di Mobile IPv6, ma allo stesso tempo la criticità di alcune fasi che incidono notevolmente sul tempo totale dell'Handoff Latency.
- Nel Cap.8 le conclusioni.

2 Il supporto di IPv6 alla mobilità

In questo capitolo ricordiamo le principali caratteristiche di IPv6 necessarie per comprendere al meglio i meccanismi che sono alla base di Mobile IPv6 e le relative relazioni di interoperabilità fra i due protocolli.

Prerequisiti per Mobile IPv6 sono sicuramente: lo spazio di l'indirizzamento e tipologie di indirizzi IPv6, il formato dell'Header IPv6 e le sue estensioni, i sistemi autoconfigurazione di IPv6, il supporto alla Sicurezza a livello di rete (IPSec).

2.1 Lo spazio di indirizzamento:

Una dei principali limiti di Ipv4 è il formato degli indirizzi a 32-bit, sicuramente incapace, nel prossimo futuro, di soddisfare l'incremento esponenziale del numero di utenti e dispositivi connessi ad Internet. Per ovviare a tali inconvenienti la nuova versione dell'Internt Protocol introduce un nuovo formato di indirizzi a 128-bit.

Un indirizzo IPv6 è composto da 8 campi rappresentati in altrettanti valori esadecimali di 16-bit separati da (:). Ad esempio:

`2001:0000:0000:0200:002D:D0FF:FE48:4672`

Per brevità di notazione è possibile sostituire sequenze contigue di valori nulli con un unico campo vuoto ed omettere eventuali zeri in testa di ciascun campo, in questo caso possiamo anche scrivere:

`2001::200:2D:D0FF:FE48:4672`

Un indirizzo IPv6 è generalmente suddivisibile in due porzioni: il *network-*

prefix (prefisso di sottorete) costituito dai primi k bit più significativi, e un suffisso (*inteface ID*) assegnato una particolare interfaccia di rete. Come per IPv4 è possibile specificare una sottorete tramite la notazione: <prefix>/<prefix-length>, ad esempio:

```
2001:FF08:49EA:D088::/64
```

2.2 Tipologie di indirizzi IPv6

Come per Ipv4 è possibile assegnare differenti indirizzi ad una stessa interfaccia. IPv6 però introduce il concetto di visibilità o *portata degli indirizzi* (scope) e ne definisce tre tipologie: ***unicast, multicast ed anycast***.

Lo *scope* specifica, cioè, la topologia nella quale l'indirizzo può essere utilizzato come identificatore unico di una interfaccia o di un gruppo di interfacce.

2.2.1 Indirizzi Unicast

Un indirizzo unicast IPv6 identifica una singola scheda di rete. Un pacchetto trasmesso ad un indirizzo unicast è trasportato all'interfaccia identificata da quell'indirizzo. Gli indirizzi unicast (e multicast) si distinguono, in funzione dello *scope*, a loro volta in: *unicast global*, *unicast site-local* ed *unicast link-local*.

- Un indirizzo *global* è visibile sull'intera Internet. La struttura è la seguente: un prefix fissato a 2000::/3 (001), un successivo prefisso di routing globale di 45-bit, un subnet ID di 16-bit e 64-bit per l'interface ID.
- Un indirizzo *unicast link-local* permette ai pacchetti di essere scambiati solo sul link a cui le interfacce sono connesse. La struttura è: un prefix fissato a FE80::/10 (1111 1110 10) e 64-bit per l'interface ID (ricavato dal MAC Address). I router non inoltreranno datagrammi con i indirizzi di destinazione di tipo unicast link-local.
- Un indirizzo *unicast site-local* limita la portata della consegna del pacchetto alla sola propria Intranet (sottorete o ***site***), i router di bordo non effetueranno routing con indirizzi site-local. Questi inirizzi funzionano analogamente alle classi di indirizzamento private IPv4. La struttura: un prefix fissato a FEC0::/10 (1111 1110 11), un campo di 16-bit per la *subnet ID*, e 64-bit per l'interface ID.

Ogni router può configurare automaticamente un indirizzo link-local unicast per le proprie interfacce utilizzando il network-prefix FE80::/10 ed un *interface ID* di 64-bit. L'indirizzo ottenuto è quindi sottomesso ad un processo per la rilevazione di eventuali indirizzi duplicati sulla rete (*DAD procedure*).

2.2.2 Indirizzi Multicast

Un indirizzo *multicast IPv6* trasporta copie di un pacchetto da una fonte a più destinatari appartenenti ad un set multicast. Un indirizzo multicast è caratterizzato da un prefix fissato ad FF00::/8. I successivi 4 bits definiscono l'indirizzo come permanente o temporaraneo. I successivi 4 bits specificano la portata dell'indirizzo (*node, link, site, organization, global*).

2.2.3 Indirizzi Anycast

Una tipologia di indirizzo del tutto nuova è l'indirizzo *anycast IPv6* il quale identifica un insieme di interfacce che appartengono tipicamente a nodi (router) differenti. Un indirizzo anycast può essere assegnato solo ad un *router* e non deve essere utilizzato come indirizzo di sorgente di un datagramma IPv6. Un pacchetto trasmesso ad un indirizzo anycast è trasportato ad una delle interfacce identificate da quell'indirizzo (tipicamente quella più vicina).

2.2.4 Indirizzi IPv6 speciali

- *Indirizzi IPv4-compatibili*. Sono indirizzi utilizzati nel processo di transizione da IPv4 ad IPv6. Realizzano tunnel IPv6 dinamici su infrastrutture Ipv4 esistenti. La struttura è caratterizzata da contenere un indirizzo IPv4 negli ultimi 32 bit e tutti zeri a monte (0:0:0:0:0:0:A.B.C.D)
- *Indirizzo di Loopback*: L'indirizzo 0:0:0:0:0:0:0:1 ovvero ::1 è utilizzato da ogni nodo per inviare pacchetti a ses stesso. Non è possibile assegnare l'indirizzo di Loopback ad una interfaccia fisica.
- *Indirizzo non-specificato*: L'indirizzo 0:0:0:0:0:0:0:0 ovvero :: può essere

utilizzato come indirizzo di sorgente da un nodo che non ha ancora configurato (stateless auto-configuration) un proprio indirizzo (site-local).

2.2.5 Frammentazione in IPv6

Le differenze più importanti fra IPv4 ed IPv6 riguardo a datagram size, MTU, frammentazione e riassemblaggio, sono:

- *Aumento del default MTU*: In IPv4, l' MTU minimo che i router ed i collegamenti fisici possono gestire era di 576 byte. In IPv6, tutti i collegamenti gestiscono formati dei datagrammi di almeno di 1280 byte. tale incremento nel formato migliora l'efficienza aumentando il rapporto fra carico utile massimo e la lunghezza dell'intestazione e riducendo la frequenza con cui la frammentazione è richiesta.
- *Eliminazione della frammentazione da parte dei Router*: In IPv4, i datagrammi possono essere frammentati dal nodo sorgente o dai router intermedi durante la consegna. In IPv6, soltanto la sorgete può effettuare la frammentazione, i router non hanno questa possibilità per motivi di efficienza. La sorgente deve quindi frammentare fino al più piccolo MTU sull'itinerario prima della trasmissione. Ciò presenta sia i vantaggi che gli svantaggi. Il riassemblaggio dei frammenti naturalmente è fatto soltanto dalla destinazione, come in IPv4.

- *Errori di formato del MTU*: Poiché i router non possono spezzettare i datagrammi, se rilevano pacchetti troppo grandi per un dato link fisico allora saranno costretti a lasciarli cadere.Per ovviare a tale inconveniente è stato definito un sistema di messaggi (ICMPv6) con i quali i router devono avviare i nodi sorgenti relativamente a datagrammi troppo grandi.
- *Path MTU Discovery*: Poiché i nodi sorgente devono decidere sulle corrette dimensioni dei frammenti, è necessario un meccanismo per la determinazione del MTU. Questa possibilità è fornita con una speciale tecnica denominata Path MTU Discovery, originalmente definita per IPv4 è stata raffinata per IPv6.
- *Eliminazione dei campi dell'Header relativi alla frammentazione*: data la minore occorrenza di frammentazione in IPv6 i relativi campi dell'Header sono stati eliminati e sostituiti, all'occorrenza, dalla relativa "Fragment extension header".

2.3 Autoconfigurazione di rete

IPv6 fornisce agli host la capacità (autoconfiguration state-less) di configurarsi in rete automaticamente senza l'uso di un protocollo statefull di configurazione (tipo DHCP). Altri meccanismi (Router Discovery) consentono ad un nodo di determinare gli indirizzi dei router ed altri parametri di configurazione necessari per la connessione ad Internet, senza alcun intervento umano.

Il processo di configurazione automatica si basa su messaggi broadcast inviati periodicamente da un router sul proprio local-link denominati: *router advertisement (RA)*. Questi messaggi (ICMPv6) contengono informazioni relative alla locale sottorete quali: il prefisso (64-bit) prefix e l'indirizzo del default router. Se un nodo (in particolare un nodo *mobile*) desidera autoconfigurarsi su questo link preleva il relativo prefisso dal messaggio RA e vi appende il proprio interface ID (MAC Address). Il nuovo indirizzo così ottenuto dovrà (MUST) sottoporsi alla procedura DAD (Duplicate Access Detection), per assicurarsi circa l'univocità sul link dell'indirizzo appena creato, prima di essere definitivamente assegnato all'interfaccia.

2.4 Duplicate Address Discovery (DAD)

La procedura *duplicate address detection* utilizza messaggi denominati *Neighbor Solicitation* e Neighbor Advertisement per verificare l'univocità degli indirizzi. In particolare il nodo invia in broadcast (Neighbor Solicitation) un messaggio, con :: (unspecified-IPv6v address) come indirizzo sorgente, del tipo: who has <my_new_address>?

Se nessuna risposta Neighbor Advertisement (Target is ...) interviente entro un fissato timeout (MAX_RTR_ SOLICITATION_DELAY come specificato nel RFC2462)

allora si assegna l'indirizzo creato alla relativa interfaccia altrimenti un eventuale nodo con quell'indirizzo risponderà con un Neighbor Advertisement: Target address is <my_new_address> .

2.5 Le estensioni dell'Header IPv6.

IPv6 è molto flessibile riguardo al supporto delle Opzioni (abolite dall'header IPv6) attraverso le *estensioni* dell'Header. Le estensioni dell'Header sono disposte fra l'intestazione IPv6 e l'intestazione di livello superiore e sono concatenate insieme usando il campo *Next Header* nell'intestazione IPv6. Ci sono sei intestazioni differenti di estensione:

1. Opzioni Hop by Hop (HBH)
2. Opzioni della Destinazione
3. Opzioni di Routing
4. Opzioni di Frammentazione
5. Opzioni di Autenticazione
6. Opzioni di Sicurezza.

Se necessiatano più estensioni dell'Header, il campo NEXT dell'intestazione indica la tipologia dell' estensione successiva fino ad indicare la tipologia del protocollo di livello superiore (TCP, UDP, ICMPv6, un pacchetto IPv6 incapsulato, etc).

2.6 Sicurezza ed Ipsec (AH ed ESP)

Il protocollo IPv4 è stato concepito per lavorare in un ambiente di tipo collaborativo e nell'ipotesi che i collegamenti di rete siano fisicamente sicuri. Tale ipotesi però non corrisponde alla realtà; le comunicazioni possono subire diversi tipi di attacchi. Si parla di *packet sniffing* quando i pacchetti in transito vengono letti da un nodo posto tra mittente e destinatario, acquisendo così informazioni riservate come la password. Altri tipi di attacchi sono noti come *IP spoofing* e *connection hijacking*. Nel primo caso si tratta di una falsificazione dell'indirizzo mittente, allo scopo di ingannare i servizi che autenticano in base a quel parametro o sovvertire l'ordine della rete falsificando i messaggi ICMP. Nell'altro caso il sabotatore si inserisce in una comunicazione in corso, introducendo dati errati.

In commercio esistono molte proposte di soluzioni a questi problemi, tutte a livello applicativo. Il loro maggior difetto è l'incompatibilità tra di esse e la duplicazione di funzionalità. Lo sviluppo di IPv6 ha permesso di offrire una risposta più efficiente alle richieste di sicurezza, in modo trasversale a tutti gli applicativi.

Un contesto in cui si possono usare AH (Authentication Header) e ESP (Encapsulation Security Payload) è quello delle reti private virtuali (VPN), cioè reti di un'impresa con sedi distribuite sul territorio che sono collegate tra loro non più da canali dedicati, ma mediante rete pubblica. Anche in IPv4 esistono queste reti e per garantire la sicurezza occorre proteggere crittograficamente i pacchetti IP e incapsularli in altri pacchetti IP creando tunnel sicuri tra i due firewall. Questa soluzione di incapsulamento può creare

problemi di compatibilità tra firewall di costruttori diversi e problemi con la frammentazione. Infatti, nel caso che i pacchetti da trasmettere abbiano la dimensione massima consentita in IP non sarà possibile incapsularli dentro un altro pacchetto IP, ma dovranno essere frammentati. Il firewall di destinazione dovrà estrarre ogni frammento e ricomporre il pacchetto per renderlo in chiaro e verificarne l'autenticità, quindi spedirlo alla corretta destinazione dopo un'eventuale frammentazione. Questo causa un decadimento delle prestazioni che può arrivare fino al 50% del throughput normale, soprattutto per pacchetti di grandi dimensioni.

In IPv6, non si deve riassemblare il pacchetto, ma semplicemente eliminare l'header più esterno che permette d realizzare il tunnel. La verifica circa l'autenticità del pacchetto è fatta direttamente dalla destinazione grazie ai nuovi meccanismi che IPv6 introduce. L'overhead introdotto dagli extension header AH e ESP ha dimensione fissa ed indipendente da quella del pacchetto originale quindi il degrado delle prestazioni è più contenuto. Questa soluzione può anche essere usata in presenza di un terminale mobile, in cui il firewall fa da *Home Agent*. Il successivo capitolo analizza nel dettaglio il protocollo Mobile-IPv6.

3 Il protocollo Mobile IPv6 (MIPv6)

Il protocollo mobile IPv6 (MIPv6) è lo standard proposto dalla IETF per fornire servizi di mobilità agli host IPv6 (RFC 3775). Il protocollo permette ad un nodo mobile di muoversi da una rete verso un altro senza la necessità di cambiare il relativo indirizzo IPv6. Un nodo mobile, pertanto, è sempre raggiungibile tramite il proprio Home Address (HADDR): l'indirizzo IPv6 che è assegnato al nodo all'interno della relativa ed usuale sotto-rete (HN: Home Network) anche al variare del corrente punto di connessione ad Internet. Quando un nodo mobile è assente dalla propria Home Network, i pacchetti possono ancora essere diretti ad esso utilizzando l'home address del nodo. In questo modo, il movimento di un nodo fra differenti sottoreti è completamente trasparente ai protocolli di livello superiore, primo fra tutti TCP.

Ogni nodo MIPv6 ha un indirizzo persistente (Home Address) che può essere usato per richiamare il nodo mobile indipendentemente dal relativo punto di collegamento corrente alla Rete. Ogni sotto-rete Ipv6 è caratterizzata da un determinato prefisso di sottorete ed è detta *Home Network* per tutti i nodi mobili ospitati al suo interno. I nodi mobili all'interno della propria HN adottano un *Home Agent*: un router IPv6 direttamente collegato alla Home Network responsabile dell'intercettazione e dell'inoltro dei pacchetti IPv6 ai nodi mobili anche (e sopratutto) quando questi ultimi sono assenti dalla HN. Il processo di adozione di un Home Agent (HA) può essere statico o dinamico ed avviene mediante un meccanismo denominato: MIPv6 Home Agent discovery.

Quando un nodo mobile IPv6 è connesso alla relativa HN funziona come

qualunque altro nodo di rete, perciò nessun particoloare protocollo di routing è richiesto. Mentre quando un nodo mobile si muove verso una rete "straniera" (FN: Foreign Network), si avvia un processo di autoconfigurazione di rete, proprio di IPv6, necessario per acquisire un indirizzo IPv6 aggiuntivo denominato: COA (Care Of Address). Il nuovo indirizzo, ottenuto accoppiando l'attuale prefisso di sottorete con il MAC Address del nodo, è di tipo "scope-site" e ha una validità temporale (lifetime) stabilita dall'Home Agent in fase di registrazione.

La tripla (HADDR, COA, lifetime) prende il nome di "Binding" del nodo mobile. Tale Binding consente al nodo mobile di "adattarsi" alla nuova rete e continuare ad essere raggiungibile dall'esterno sempre tramite l'HADDR e MIPv6.

In particolare per fare in modo che i pacchetti IPv6 destinati all'HADDR raggiungano efficientemente la corretta posizione, le informazioni di routing (bindings) relative all'home address devono essere aggiornate sia presso l'Home Agent che in tutti gli eventuali nodi corrispondenti con il nodo mobile.

MIPv6 fornisce questa funzionalità tramite l'introduzione di una "binding cache" ed un sistema di messaggi (Binding Messages) caratterizzati da una nuova estensione dell'header IPv6 denominata: Mobility Header.

3.1 Bindings Cache

La relazione tra l'HADDR del nodo mobile ed il suo attuale COA e detta *Binding*. Tutti i nodi MIPv6 devono mantenere una tabella di tali bindings nella cosidetta: Binding Cache. Ciascuna entry è mantenuta per ogni nodo mobile con cui la comunicazione sta attualmente avvenendo. La Binding Cache mantiene quattro informazione fondamentali al funzionamento di MIPv6, come illustrato nella Tabella 1 (altri campi sono presenti per accertare l'ordinamento corretto dei messaggi di controllo, ma questi sono stati omessi). L'HADDR costituisce il campo *chiave* della Cache:

Home Address	*Care of Address*	*Lifetime*	*Home Agent*
fec0:106:2700::4	fec0:106:1100:0:240:5ff:feae:c364	120	Yes
fec0:106:1100::6	2001:2101:0:b00:a00:6aff:fe2b:137c	43	No

Tabella1: Binding Cache

Quando un nodo desidera trasmettere un pacchetto IPv6 ad un host remoto, viene prima ricercato il relativo HADDR nel campo chiave della Binding Cache.

Se nessuna corrispondenza viene rilevata, il pacchetto è trasmesso secondo le usuali procedure di routing di Ipv6, altrimenti il pacchetto è incapsulato (IPv6-within-IPv6) per reindirizzare il pacchetto al Care of Address specificato nella Binding Cache. Questo assicura il percorso ottimale fino alla posizione corrente del nodo mobile. La forma dell' incapsulamento dipende dallo stato del *flag* Home Agent presente nella Binding Cache.

3.2 Operazioni dell'Home Agent

Se il flag Home Agent è settato allora il nodo che mantiene questa cache funge da Home Agent per il nodo mobile di destinazione, in tal caso e se il pacchetto è inviato in un contesto di forwarding allora il pacchetto è incapsulato utilizzando un Tunneling IPv6 in IPv6:

IPv6 Header (Outer)	*IPv6 Header (Inner)*	*Transport Header*	*Payload*
Source:Home Agent	Source: Corresp.Node	TCP/UDP	Data
Dest: CareOfAddress	Dest: HADDR		
(40 bytes)	(40 bytes)		

Tabella2: IPv6 within IPv6 Encapsulation

Il Tunneling IPv6 è usato dagli Home Agents per inoltrare i pacchetti IPv6 instradati (erroneamente) alla home network del nodo mobile mentre questo è assente dalla propria *usuale* sede. Il tunnel presenta il vantaggio della conservazione del pacchetto completo originale IPv6, che è importante poichè la modifica ad un'intestazione IPv6 potrebbe causare problemi con i protocolli degli strati superiori, quali il TCP. I pacchetti intercettati vengono così inviati, attrverso il tunnell, direttamente all'indirizzo topologicamente rilevante del nodo mobile: il care-of-address. Sarà poi il nodo mobile stesso ea effettuare il decapsulamento del pacchetto.

3.3 Operazioni del Correspondent Node

Se il flag Home Agent non è settato nella Bindin Cache o se il pacchetto non è stato ricevuto da un contesto di forwarding, allora viene utilizzato una nuova intestazione IPv6 per indicare il routing diretto tra correspondent Node e Nodo mobile (Routing Optimization)

IPv6 Header	***IPv6 Routing Header***	***Transport Header***	***Payload***
Source:Corresp. Node	NextHop: HADDR	TCP/UDP	Data
Dest: CareOfAddress			
(40 bytes)	(24 bytes)		

Tabella3: IPv6 Routing Header Encapsulation

Come si evince dalla tabella 3, l'uso dell'*IPv6 Routing Header* riduce la larghezza di banda effettiva richiesta per l'incapsulamento del pacchetto rispetto a IPv6 in IPv6 di 16 byte. Questa riduzione del formato del pacchetto è possibile a causa della ridondanza spaziale che si avrebbe se si utilizzasse il tunneling Ipv6-within-IPv6: gli indirizzi sorgente sarebbero gli stessi nell'header interno ed esterno con conseguente spreco di larghezza di banda.

3.4 Coerenza della Binding cache

L'uso della Binding Cache e dell'incapsulamento IPv6 fornisce un meccanismo per permettere l'instradamento ottimale dei nodi mobili. Questo meccanismo, tuttavia, conta sull'esattezza e la freschezza delle informazioni contenute all'interno della Cache. Effettivamente, per proteggersi da malfunzionamenti delle macchine (comuni in un ambiente mobile dati i vincoli di durata di batterie, ecc.) ed eventuali periodi lunghi di disconnessione dalla rete, le entry della Binding cache per un nodo mobile devono persistere anche dopo un lungo periodo di disconnessione totale.

Mobile IPv6 mantiene la coerenza della Binding Cache con l'uso di speciali messaggi: Binding Update (BU), Binding Acknowledgement (BA), Binding Request (BR), scambiati periodicamente o su richiesta tra nodo mobile e Home Agent/Correspondent Node.

Il resto di questa sezione descrive dettagliatamente questi messaggi e come interagiscono per fornire una accurata coerenza della Binding Cache.

I messaggi di binding (richiesta, aggiornamento e riscontro) sono tutti inviati attrverso le *opzioni di destinazione* IPv6. Ciascuno con il loro proprio *tipo* di opzione della destinazione. L'utilizzazione delle opzioni fornisce diversi vantaggi rispetto a metodi integrati di messaggistiche di controllo. In primo luogo, i messaggi possono essere disposti in linea con l'intestazione dei pacchetti esistenti IPv6, riducendo così le spese generali della trasmissione dei messaggi. Secondariamente, poichè nessun protocollo di strato di trasporto è coinvolto nella trasmissione del messaggio, non si hanno problemi con eventuali sistemi di firewall relativamente al blocco dei messaggi di controllo.

3.5 Messaggi di Binding Update (BU)

I messaggi di *Binding Update* (BU) sono trasmessi dai nodi mobili agli Home Agent ed ai nodi corrispondenti per generare o aggiornare la relativa entry nella Binding Cache di destinazione riguardo l'HADDR del nodo mobile mittente. I messaggi BU possono essere generati in qualunque momento da un nodo mobile, ma sono sempre trasmessi dopo la rilevazione di un pacchetto IPv6 che ha viaggiato attraverso un tunnel Ipv6-within-IPv6 generato dall'Home Agent di quel nodo. La ricezione di un tal pacchetto indica che il nodo corrispondente che ha generato attualmente il pacchetto non ha il binding per questo nodo mobile (altrimenti il pacchetto sarebbe stato trasportato via IPv6 Routing Header). Per garantire che binding non più operativi non siano mantenuti indefinitamente nella Binding Cache, MIPv6 adotta un meccanismo 'soft state' (tramite un lifetime per ciascun binding) per la loro eliminazione. Allo scadere del lifetime, il binding è rimosso dalla cache. Il valore del lifetime è regolato e aggiornato dal campo corrispondente contenuto all'interno dei messaggi BU. Un valore di lifetime pari a zero indica la rimozione del relativo binding (così come avviene quando un nodo mobile ritorna nella propria Home Network dopo essere stato in una Foreign Network).

3.6 Messaggi di Binding Acknowledgement (BA)

I messaggi BA forniscono il riscontro di avvenuta ricezione di un BU e sono trasmessi ai nodi mobili da nodi correspondenti ed agenti domestici. Forniscono le risposte di controllo, prima fra tutte il valore del *lifetime*, ai nodi mobili in risposta ai Binding update e sono usati per fornire la conferma certa dell'aggiornamento e indicare tutti gli eventuali errori generati durante l'elaborazione dei BU. I nodi mobili abbinano i BA con i loro BU corrispondenti tramite il confronto di numeri di sequenza progressivi.

3.7 Messaggi di Binding Request (BR)

I nodi corrispondenti e gli Home Agent che rilevano un'entry nella loro BC che sta si sta avvicinando alla scadenza del relativo lifetime possono decidere di trasmettere un messaggio di richiesta (BR) al rispettivo nodo mobile. La ricevuta di un messaggio BR da un nodo del mobile provoca la trasmissione di nuovo BU alla fonte di quella richiesta. Questo meccanismo permette ai nodi corrispondenti di evitare brevi periodi di routing non ottimale, dovuto alla scadenza di un binding operativo.

3.8 Binding Update List

Poichè un nodo mobile vaga dalla rete alla rete, è essenziale che i messaggi BU siano trasmessi appena possibile all'Home Agent e/o Correspondent Node, in modo da facilitare un handoff veloce. I nodi mobili inoltre non possono contare sul meccanismo di timeout del lifetime usato per rinfrescare bindings stagnanti e(valori tipici del lifetime sono dell'ordine dei minuti). Pertanto una

struttura di dati supplementare, la *Binding Update List* (BUL), si rende necessaria per i nodi mobili. Questa lista tiene traccia dello stato di tutti i nodi corrispondenti o Home Agents che via via si incontrano.

La Binding Update List contiene una voce per ogni nodo corrispondente o Home Agent a cui un BU è stato trasmesso. Le entry della lista contengono informazioni quali l'indirizzo (COA) ed il tempo in cui un BU è stato trasmesso, lo stato di tutti i binding non ancora riscontrati, il lifetime del binding corrente, un Home Agent Flag ed il numero di sequenza progressivo dellultima trasmissione. Le entry della lista sono cestinate nel momento in cui il relativo binding espira. Il mantenimento della BUL permette prestazioni significativamente più veloci di handoff, infatti dopo che un handoff è stato completato e l' autoconfigurazione IPv6 (stateless o statefull) è stata completata, la BUL viene consulatata ed un BU viene trasmesso ad ogni nodo contenuto all'interno della lista, quindi vengono aggiornante le binding cache di tutti i nodi corrispondenti.

3.9 Proxy Neighbour Discovery

Poiché i pacchetti destinati ad un nodo mobile possono essere diretti erroneamente alla relativa home network, posizionare gli Home Agent all'interno di un router di bordo IPv6 permetterebbe l'intercettazione efficiente di questi pacchetti, poichè probabilmente attraverserebbero quel router. Tuttavia, il presupposto che i pacchetti raggiungeranno automaticamente questo router di bordo non è un evento certo. Per esempio, si consideri il caso di un nodo corrispondente situato sulla stessa home network del nodo mobile. Se il nodo corrispondente dovesse trasmettere un pacchetto a quel nodo mobile, la relativa tabella di routing detterebbe che il nodo mobile è direttamente accessibile e non è richiesto l'inoltro da un router. In questo caso, l'home agent non potrebbe intercettare il pacchetto. Mobile IPv6 risolve questa condizione con una tecnica chiamata proxy neighbour discovery (PND).

Neighbor discovery (scoperta del vicinato) è un protocollo standard IPv6 per la rilevazione degli indirizzi MAC dagli indirizzi IPv6, analogamente al protocollo di ARP per IPv4. La tecnica PND coinvolge un nodo IPv6 mascherandolo come se fosse un altro nodo a livello MAC, rispondendo erroneamente ai messaggi di Neighbour Solicitation con il proprio MAC address. In questo caso gli Home Agent utilizzando PND si mascherano da MN per assicurarsi l'intercettazione di ogni pacchetto IPv6 destinato ad un nodo mobile trasmesso sulla relativa home network. Per compiere questo gli Home Agent gestiscono una *PND Table* contenente gli indirizzi IPv6 dei nodi di cui sta fungendo da Proxy. Le entry di questa tabella sono aggiunte e rimosse così come i messaggi BU con gli Home Agent Set Flag sono aggiunti

e rimossi dalla Binding Cache.

3.10 Home Address Option

Quando i nodi mobili sono fuori dalla loro rete usuale hanno una scelta su quale indirizzo IPv6 (di portata *site* o *global*) utilizzare come source address per i loro pacchetti IPv6 uscenti. Potrebbe essere utilizzato sia l'HADDR che l'attuale Care-of Address (COA). Tuttavia, nessuna di queste scelte è particolarmente desiderabile, infatti, se si utilizza il COA, allora l'indirizzo di sorgente per i pacchetti successivi potrebbe cambiere all'avvenire di un handoff. Ciò comporterebbe problemi irreparabili per i protocolli di strato superiore quali il TCP, che mantengono identificatori di livello di trasporto e le checksum basandosi su informazioni (indirizzi) di livello di rete. D'altra parte, se fosse utilizzato l'HADDR, i pacchetti uscenti IPv6 diventerebbero suscettibili dell'*ingress filtering* operato dai router (HA) di bordo. L'ingress filtering è implementato da molti router per migliorare la sicurezza della sottorete in cui operano. La filtrazione in ingresso comporta l'ispezione dell'indirizzo di sorgente di tutti i pacchetti IP ricevuti e di verifica che l'itinerario a quell'indirizzo si trovi lungo l'interfaccia su cui il pacchetto è stato ricevuto. Tutti i pacchetti che vengono a fallire questa prova vengono eliminati per motivi di sicurezza (ed evitare eventuali attacchi di *IP address spoofing*).
Mobile IPv6 definisce una nuova opzione di destinazione IPv6, conosciuta come l'opzione di *Home Address Option*, la quale fornirsce una soluzione al problema dell' indirizzo di sorgente sicura per i protocolli di trasporto e non suscettibile dell'ingress filtering. Questa soluzione è realizzata mediante una forma ottimizzata di *reverse tunneling e con* un livello minimo

d'incapsulamento. La tabella 4 illustra l'*Home Adrress Option*:

IPv6 Header	***Home Address Option***	***Transport Header***	***Payload***
Source:Care-Of Address	Home Address	TCP/UDP	Data
Dest: Correspondent Node			
(40 bytes)	(18 bytes)		

Tabella 4: MIPv6 Home Addre Option

La specifica mobile IPv6 stabilisce che nodi mobili in tal caso debbano utilizzare il COA come Source Address, per evitare l'ingress filtering, ma allo stesso tempo tutti i protocolli di strato superiori dovrebbero presupporre che l'indirizzo di sorgente dei pacchetti uscenti sia l'Home Address del nodo monbile. Tutti i pacchetti uscenti da un nodo mobile includono l'Home Address Option. Il nodo corrispondente alla ricezione di un pacchetto sostituirà, prima di tutte le elaborazioni degli strati superiori, l'indirizzo di sorgente del pacchetto (COA) con quello presente nella Home Address Option (HADDR).

3.11 Home Agent Discovery

MIPv6 fornisce un meccanismo ai nodi mobili per rilevare automaticamente la presenza degli Home Agents sulla relativa rete. Questo meccanismo coinvolge tutti gli Home Agents caratterizzati, oltre che dal proprio indirizzo ***IPv6 link-local***, anche da un ulteriore indirizzo IPv6 di tipo ***anycast site-local***. I nodi mobili nel momento in cui ritornano nella propia home network, dopo essere stati altrove, rilevano la presenza di Home Agent tramite la ricezione di messaggi ICMPv6 denominati ***Agent advertisement*** (con il flag H settato), a questo punto per deregistrare il vecchio COA, trasmettono un messaggio di Binding Update (con richiesta di lifetime nulla dato che ritorna a 'casa') a questo indirizzo anycast. Questo messaggio sarà trasportato ad al più un Home Agent sulla home network poiche l'indirizzo di destinazione è tipo *anycast site local*. Alla ricezione del messaggio, l'HA risponde con un Binding ACK, informante il nodo mobile dell'indirizzo IPv6 link-local dell'HA.

3.12 Mobility Header (MH)

L'*intestazione di mobilità* è una nuova estensione dell'intestazione IPv6 destinata ai messaggi di segnalazione MIPv6. La MH è usata da Nodi mobili, Home Agents e nodi corrispondenti per tutti i messaggi relativi alla creazione ed amministrazione dei *Bindings*.

L'intestazione di mobilità è identificata dal valore 62 del campo Next-header dell'header IPv6. Oltre ai soliti campi *Header Lenght* e *Checksum*, Il campo *type* della MH identifica uno dei seguenti possibili messaggi:

- Home Test Init (HoTI)
- Care-of Test Init (CoTI)
- Home Test (HoT)
- Care-of Test (CoT)
- Binding Request (BR)
- Binding Update (BU)
- Binding Acknowledgement (BA)
- Binding Missing (BM).

3.13 Return Routability (RR)

Il metodo di Return Routability (RR) è un nuovo meccanismo di autorizzazione dei messaggi di BU fra nodi mobili e nodi corrispondenti. È basato sul principio di scambio di 'cookies' per verificare che il nodo mobile è 'on-line' sull'indirizzo indicato (COA). I cookies sono usati dal nodo mobile per cifrare e proteggere il messaggio finale di BU diretto (routing optimization) al nodo corrispondente.

Se un nodo mobile (MN) vuole comunicare con un altro nodo (CN) esistono diverse possibilità: o il MN si trova nella Home Network, in tal caso utilizzerà le usuali procedure di routing IPv6 (se il CN è raggiungibile), o si trova in una Foreign Network, in questo caso potrà o comunicare via Home Agent (Triangular Routing) o direttamente con il CN (Routing Optimization). Nell'ultimo caso un volta che il nodo mobile ha completato l''handoff ha inizio la procedura di Return Routability per stabilire un binding sicuro (autenticato) con il CN.

I nodi mobili avvieranno la procedura di Routing Optimization non appena ricevono datagrammi provenienti dal correspondent node attraverso l'Home Agent poichè il CN non ha ancora registrato nella propria Binding Cache il corrente COA del nodo mobile. La registrazione del COA presso il CN però necessita di autenticazione sicura, per proteggersi da eventuali attacchi di ridirezione, tramite il processo di *Return routability*, le cui fasi sono:

- Il MN invia 2 cookie al CN: il primo è contenuto in un messaggio denominato CoTI (Care-Of Test Init) direttamente destinato al CN (usa il COA come source address). Il secondo cookye è inviato all'interno di un messaggio denominato HoTI (Home Test Init) al CN attraverso l'Home Agent (in tunneling).

- Il CN risponde al MN con i messaggi CoT (Care-of Test) e Hot (Home Test) contenenti i dati (i cookies ricevuti, un nonce, ed la sua chiave pubblica) necessari al MN per generare una chiave simmetrica di sessione con cui cifrerà il Binding Update.

- Il Mn invia il Binding Update cifrato al nodo corrispondente.

- Il CN alla ricezione del BU può verificare le informazioni e generare la relativa entry nella Binding Cache. Il BA è facolatativo:

La seguente Fiugura evidenzia le varie fasi della Return Routability Procedure:

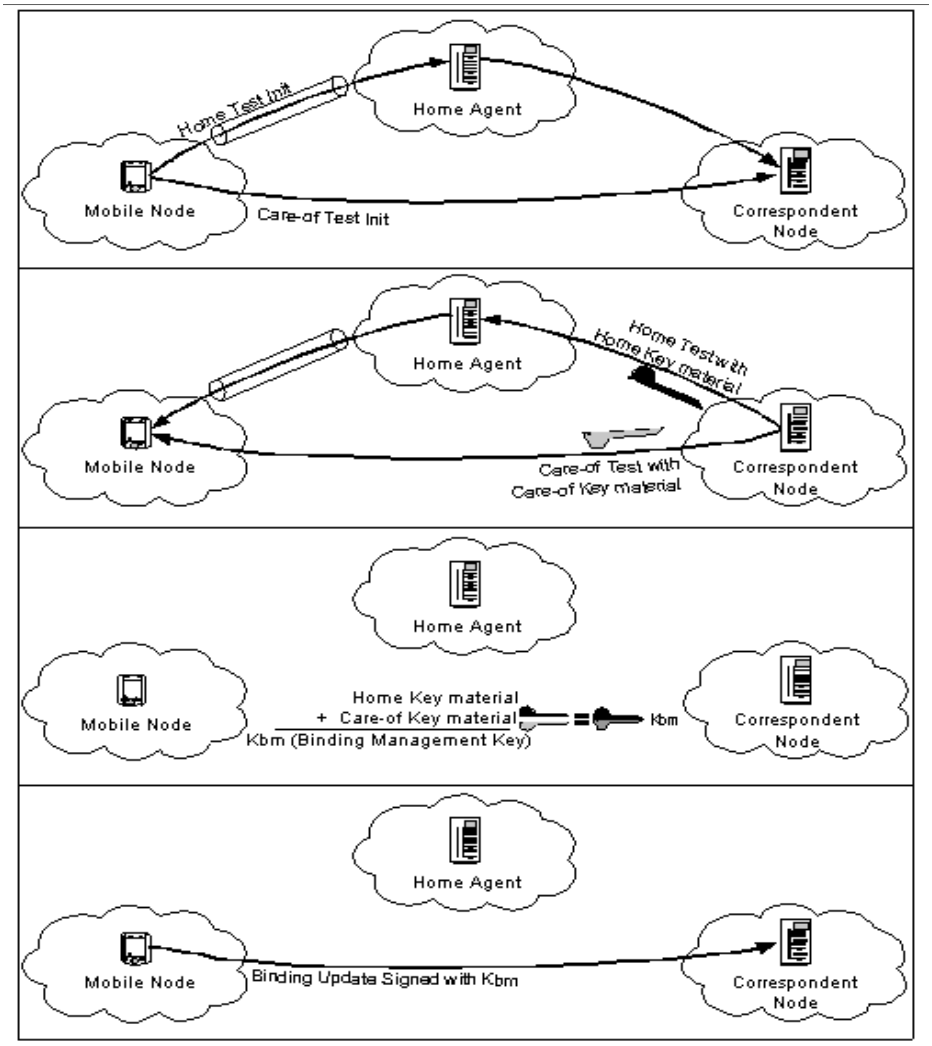


Figura 1: Return routability Procedure

Una procedura del tutto analoga è utilizzata per autenticare la deregistrazione di un COA nel momento in cui il nodo mobile ritorna presso la propria Home Network e desidera continuare a comunicare con il CN (con routing IPv6).

3.14 Movement Detection

Attualmente esistono due differenti algoritmi per la rilevazione del movimento. Il primo, oramai obsoleto, è denominato *Lazy Cell Switching (LCS)*:

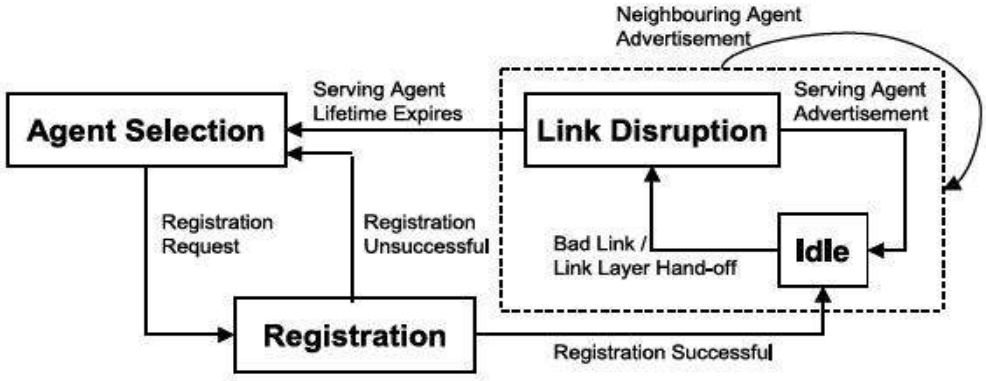


Figura 2: Lazy Cell Switching (LCS)

L'obiettivo principale della LCS è evitare gli handoff fino a che non siano assolutamente necessari (da cui il nome: *commutazione pigra di celle wireless*). La figura rappresenta la relativa macchina a stati finiti.

Lo stato Idle indica la situazione in cui il nodo mobile risiede all'interno della Home Network, o si trova in una Foreign Network dopo aver effettuato un binding valido.

Lo stato Link Disruption rappresenta la condizione che segue l'handoff a livello 2, in questo stato l'eventuale binding del nodo mobile non è più valido. La transizione fra Idle e Link Disruption si presenta senza alcuna notifica dal/al MIPv6. Fino a quando il lifetime del binding in corso non espira, l'home Agent del nodo mobile è ignaro del cambiamento di posizione. Per questo motivo i due stati sono rappresentati nella figura all'interno di un

rettangolo punteggiato indicante che per MIPv6 essi sono indistinguibili. Solo quando arriva la notifica di Lifetime scaduto si accede allo stato *Agent Selection* per poi finalmente avviare una nuova procedura di Binding Update.
Un secondo algoritmo (abilitato di default in MIPv6 per Linux) più perfomante è quello denominato *Eager Cell Switching:*

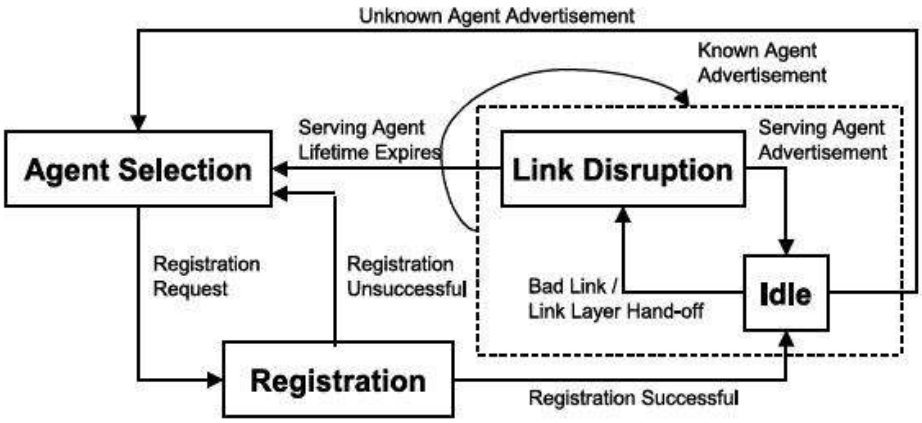


Figura 3 Eager Cell Switching

Al contrario del caso LCS questa procedura è fortemente orientata ad handoff frequenti, infatti è possibile passare direttamente dallo stato Idle allo stato Agent Selection non appena arriva un Router Advertisement da parte di router sconosciuto.

3.15 Confronto fra le implementazioni di MIPv6

	MIPL	Cisco[1]	Microsoft[2]	KAME	HP-UX
Platform	Linux 2.6.8.1[a]x	Cisco IOS	2000/XP/CE	FreeBSD	HP-UX-11i
Modes	MN/HA/CN	HA/CN	MN/HA/CN	MN/HA/CN	HA/CN
PND	Yes	Yes	Yes	Yes	Yes
IPv6-in-IPv6 tunnelling	Yes	Yes	Yes	Yes	Yes
DHAAD	Yes	Yes	Yes	Yes	Yes
Binding Management	Yes	Yes	Yes	Yes	Yes
HAO	Yes	Yes	Yes	Yes	Yes
Movement Detection	RAs	N/A	RAs and NDIS notifications	RAs	N/A
Smooth Handoff	Yes	Yes	Yes	Yes	N/A
IPsec	No (v1.1) Yes (v2.0)	No	Yes	Yes	Yes with HP-UX IPsec
Key exchange	MD5 or SHA-1	No	Manual	Manual	Unknown
Support for notebooks/ PDAs	Yes	N/A	Yes	Poor	N/A
MIPv6 built-in	No	Yes	No	Yes but not enabled by default	No. Is a component of TOUR 2.0
No. of patches	1	0	1 (XP and CE)	0	1
Set-up tools	mipdiag	command line tools	Auto-configuration and command line	Command line tools	Unknown
Licence	GNU	Commercial	Commercial	GNU	Commercial

Tabella 5 : Implementazioni di Mobile IPv6

MIPL 2.0 RC1 è stato rilasciato di recente ed è tuttora in fase di testing con kernel 2.6.8.1. In questo lavoro di tesi, iniziato nel luglio 2004, si è utilizzata MIPL 1.1 su kernel 2.4.26.

La nuova versione è la prima rilasciata per il kernel serie 2.6, supporta IPsec per la sola protezione della segnalazione *Home Registration*, mentre non supporta ancora la cifratura dei tunnel HoTI/HoT (scambiati durante la procedura di return routability) e dei payload in generale (ESP). Queste caratteristiche saranno implementate nelle prossima release, la quale sarà pertanto completamente *RFC3775 ed RFC3776 compliant.*

4 -Metodologie di analisi

L'analisi delle performance di Mobile IPv6 è stata condotta su due livelli: analisi attiva, ovvero sulla base di segmenti UDP appositamente generati, ed analisi passiva, ovvero sulla base dei file di log di *tcpdump*.

4.1 Analisi attiva

L'obiettivo di questa analisi consiste nella misura dei principali parametri tipici di *network benchmarking* nel caso in cui si utilizzi Mobile-IPv6. In questo caso l'attenzione è posta al funzionamento a regime del protocollo ed i suoi effetti in termini di Throughput e Frame Loss Rate. In particolare i test e le misure sono state effettuate seguendo le indicazioni del BMWG (*BenchMarking Working Group*) fornite nel RFC 2544 (*Benchmarking Methodology for Network Interconnect Devices). I parametri considerati sono:*

- *Throughput (TRH):* il massimo data rate di un dispositivo in condizione di nessuna perdita di frame da parte del dispositivo.
- *Frame Loss Rate (FLR)*: percentuale di frame persi per scarsità di risorse.
- *System Recovery (SYR)*: caratterizza la velocità con cui un DUT (Device Under Test) recupera da una condizione di overload.
- *Back-To-Back Test (B2B)*: E' il massimo numero di frame, di dimensione fissata, trasmissibili in un *burst* senza perdite di frame.

Questi parametri sono stati rilevati al variare degli scenari (SUT: System Under Test) tipici di un sistema di mobility:

- *MN at Home*: in questo scenario il nodo mobile risiede nella propria Home Network. Il supporto alla mobilità in tal caso comporta, a regime, l'overload causato dai messaggi di advertisement emanati periodicamente dall'Home Agent. Tali messaggi devono essere inviati abbastanza frequentemente (dell'ordine di pochi msec) per assicurare veloci handoff di nodi mobili che, o ritornano nella Home Network dopo essere stati altrove, o si registrano in questa rete perchè "stranieri".

- *MN at Foreign Network with Routing Optimization*: in questo scenario il nodo mobile non risiede nella sua usuale rete ma in una Foreign Network e può comunicare con un nodo corrispondente per via diretta senza l'intermediazione dell'Home Agent. Il supporto alla mobilità comporta overhead di comunicazione dovuti a: gli advertisements emanati dall'Access Router, le estensioni degli header IPv6: Home Address Option, Mobility Header, Tunneling IPv6 (in fase di Return Routability i messaggi HoTI sono inviati al Correspondent Node attraverso -tunnel- l'Home Agent) .

- *MN at Foreign Network with Triangular Routing*: in questo scenario il nodo mobile risiede in una Foreign Network e può comunicare con altri nodi solo attraverso il tunneling gestito dall'Home Agent. Il supporto alla mobilità comporta overhead di comunicazione dovuti a:

gli advertisements emanati dall'Access Router e le estensioni degli header IPv6: Home Address Option, Mobility Header, Tunneling IPv6. In tal caso tutti i datagrammi sono inviati al Correspondent Node attraverso l'Home Agent.

4.2 Analisi passiva

Questa metodologia si basa sullo studio analitico dei messaggi scambiati, e le relative temporizzazioni, fra i componenti tipici di un ambiente wireless MIPv6-enabled: Home Agent, Access Router, Correspondent Node e Mobile Node. L'obiettivo dell'analisi passiva è determinare la tempistica di tutte le fasi costituenti il principale collo di bottiglia di Mobile-IP: l'*handoff*.

Infatti, esclusa la fase di transizione da una rete ad una altra (handover), grazie alla procedura di *routing optimization* (che sostituisce il *routing triangolare*) ed alle estensioni dgli Header IPv6 misurare le performance di MIPv6 equivale a misurare quelle di IPv6. A tal proposito verrà proposta una metodologia per individuare i (macro)processi di segnalazione rilevanti ai fini del calcolo della latenza dovuta all'handoff.

MIPv6 è completamente integrato all'interno della suite IPv6 e quindi di quest'ultima utilizza al massimo tutte le funzionalità offerte. I parametri da analizzare scaturiscono da entrambi i protocolli.

La seguente tabella evidenzia le procedure cooperanti durante un handoff (sia nel caso di *Home De/Registration* che nel caso di *Correspondent Node De/Registration*) che pertanto devono necessariamente essere monitorate:

Procedura (tempi in sec.)	***Protocollo***
Movement Detection (MDT)	IPv6
Care-Of Address formation (COA)	IPv6
Duplicate Address Discovery (DAD)	IPv6
Home Registration (HRG)	MIPv6
Home Deregistration (HDR)	MIPv6
Return Routability Procedure (RRP)	MIPv6
Correspondent Registration	MIPv6
Correspondent Deregistration	MIPv6

Tabella 6: Le procedure

Ulteriori overload introdotti da MIPv6 sono dovute all'eventuale utilizzo di IPSec per cifrare la segnalazione dei Binding (*ESP Header* nei messaggi di Binding Update e Binding Acknowledgement).

4.1 Device e Systems under Test

Il documento RFC 2544 definisce un set completo di test che possono essere implementati per misurare le performance caratteristiche di un dispositivo di rete (DUT: Device Under Test) o un intero sistema di networking (SUT: System Under test)

Il set proposto ha caratteristiche di carattere generico pertanto è necessario considerare solo i test applicabili allo specifico caso di una rete IPv6 con link Ethernet e Wireless (802.11x).

Le tecniche di testing devono comunque tenere conto della ripetibilità, la varianza e la rilevanza statistica di un certo numero di prove che sia il più piccolo possibile.

Alcune definizioni:

- *DUT: Device Under Test*: E' un singolo dispositivo di cui si vogliono misurare le prestazioni. Riceve i frame da una o più delle sue interfacce di input e li inoltra ad una o più delle sue interfacce di output in accordo con le informazioni sugli indirizzi ricavate dai frame.
- *SUT: System Under Tes*t, un insieme di dispositivi di rete visti come una singola entità sia per quanto riguarda il traffico offerto che per quanto riguarda i benchmark e le misure.

In generale le performance sono ottenute applicando un input, tramite un ***Sender***, ad un DUT/SUT ed osservando l'output sul ***Receiver*** (il DUT o uno dei componenti del SUT).

4.1.1 Definizione dei DUT/SUT di un sistema MIPv6

Un sistema di mobilità che minimizza il numero di link e dispositivi interconnessi è il segente:

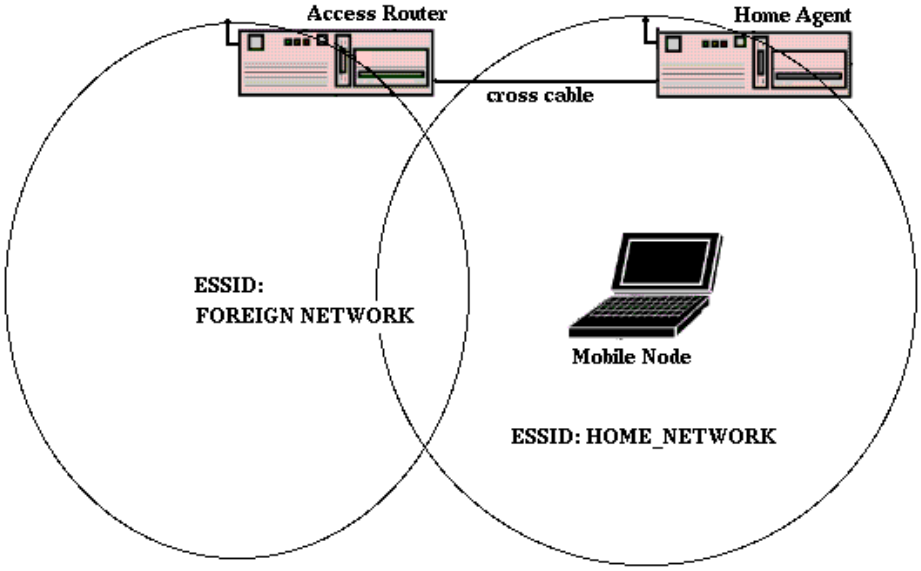


Figura 4: System under Test

Il sistema comprende due subnet IPv6 interconnesse da un link Wired, in ciascuna dei quali è presente un Agente di mobilità che irradia messaggi RADV (Router Advertisement) attraverso un interfaccia wireless. Un Nodo Mobile MIPv6 si sposta da una rete all'altra attraverso la propria interfaccia wireless. L'agente di mobilità della Home Network è anche un Home Agent (RADV con flag H). Sulla Visited Network è presente un Access Router (RADV senza flag H) che funge anche da Correspondent Node.
Sulla base di un siffatto Testbed è possibile individuare i seguenti DUT e SUT:

- **DUT1: MN...HA**: Il Sender è il Nodo Mobile, il Receiver è l'Home Agent, il DUT risultante è rappresntato dal Canale Hertziano interposto tra gli host comunicanti. I parametri rilevati da questo sistema possono essere utilizzati come base per successivi confronti in quanto, essendo il nodo mobile nella propria Home Network, nessun processo di mobilità viene avviato. Le prestazioni attese sono quelle di un usuale link wireless con IPv6.
- **DUT2: HA-AR**: In questo caso il Sender è l'Home Agent, il Receiver è l'Access Router, il DUT è il link Ethernet (cross cable). A livello 3 l'attenzione sarà posta sulla *Return Routability Procedure (HoTi e HoT Messages)* . Le prestazioni attese sono quelle di un link Ethenet con IPv6.
- **DUT3: MN...AR**: In questo caso il Sender è il Nodo Mobile, il Receiver è l'Access Router, il DUT è il link wireless, a livello 3 interviene il processo di *Routing Optimization*. Le prestazioni attese sono quelle di un link Wireless con IPv6.
- **SUT: MN...AR-HA**: Il sender è il Nodo Mobile, il Receiver è l'Home Agent. Il SUT è costituito dalla sequenza link_wireless+Access Router+link_ethernet. In questo caso sarà rilevante analizzare globalmente gli effetti della mobilità (handoff).

4.1.2 Frame Size

Come indicato nel RFC 2544 tutti i test devono essere eseguiti a differenti frame size, misurati in byte. In particolare devono includere il minimo ed il massimo e alcuni valori intermedi per una piena caratterizzazione delle performance del collegamento. Salvo eccezioni almeno cinque valori devono essere considerati. Il minimo e il massimo frame size dipende dalle limitazioni dei media e protocolli utilizzati. Valori tipici sono:

```
64, 128, 256, 512, 768, 1024, 1280, 1518
```

Frame size per Ethernet (in byte)

In presenza di connessioni tra più DUT, per i canali che supportano differenti valori di MTU deve essere utilizzato l' MTU maggiore.

Per l'esecuzione dei benchmark del protocollo IPv6 la lughezza totale (Header Gap) degli header è :

```
Header_gap = [14 + 4 (802.11 + FCS)] + 40 (IPv6) + 8 (UDP) =
                                66
```

Il frame size sarà comprensivo dell'*Header Gap* pertanto i payload di test per IPv6 su ethernet 100Mbps risultano:

```
-2, 62, 190, 446, 702, 958, 1214, 1452
```

Payload length su Ethernet (in byte)

il primo valore ovviamente non sarà applicabile, in quanto negativo, pertanto il minimo payload length ammissibile è 62 byte.

4.1.3 Maximum frame rate

Il massimo frame rate a cui è eseguito il test quando si sta testando una connessione Ethernet o 802.11b deve essere il massimo frame rate teorico per quel frame size sul media. Tuttavia per il Throughput test si possono usare dei frame rate anche superiori allo scopo di scoprire l'effettiva portata del collegamento o per osservare le reazioni del collegamento a condizioni di overlaod.

4.1.4 Traffico Bursty

Misurare le performance con un flusso statico di frame non sempre è sufficiente per simulare delle condizioni di uso tipiche. Per tale motivo l'RFC 2544 dichiara che in alcuni test si può far uso di ***burst:*** i frame inviati con il minimo gap possibile di inter-frame space.

In tal caso è necessario misurare il minimo intervallo di tempo possibile fra burst senza perdita di frame. Inoltre, data l'impossibilità di sincronizzare ogni singnolo frame, questa si rivela una scelta obbligata quando si deve inviare uno stream di frame ad uno specifico rate con una specifico frame size.

4.1.5 Il generico Test

Ogni test consiste di un certo numero di prove. Ogni prova restituisce uno specifico parametro. Ogni prova consiste delle seguenti fasi:

1. Inviare un stream di frame al solo scopo di svegliare le interfacce per il test.
2. Aspettare per almeno 5 secondi affinché il collegamento si restabilizzi.
3. Eseguire la prova.
4. Aspettare 2 secondi per ricevere ogni frame residuo.
5. Il server inoltra i risultati al client.
6. Aspettare per almeno 5 secondi affinché il collegamento si restabilizzi.
7. Tornare al passo 3.

La durata del test è un compromesso tra lo scopo del test e la durata del benchmark. La durata di ogni prova deve essere di almeno 60 secondi. I test che comportano ricerche binarie, come ad esempio il Throughput test, può comportare prove di durata minore.

4.2 Performance test (RFC 2544)

Di seguito vengono riportati una descrizione dei test utilizzati sulla base delle indicazioni riportate nel' RFC 2544:

4.2.1 Throughput Test

- *Definizione*: *Il throughput è il massimo data rate di un dispositivo in condizione di nessuna perdita di frame da parte del dispositivo stesso. Si misura in: bit al secondo (bps) o in frame di N-ottetti (Byte) al secondo.*
- *Procedura*:

  1. Inviare uno specifico numero di frame, con un determinato frame size, ad uno specifico rate al DUT
  2. Contare i frame ricevuti
  3. Se non sono stati ricevuti tutti i frame inviati ridurre il rate del test, altrimenti aumentarlo (implica un ricerca binaria).
  4. Tornare al passo 1, almeno che non sia sia raggiunto un grado di tolleranza della misura che sia abbastanza soddisfacente.

- *Formato dei risultati*: un grafico in cui sulle ordinate abbiamo il frame size e sulle ascisse il Throughput. Dovrebbe essere presente un linea per il Throughput teorico e una per quello misurato più eventuali altre linee per ogni tipo di data stream testato. Il grafico deve essere accompagnato dalle seguenti indicazioni: protocollo, formato dei frame, tipo di media usati per i test.

La misura può essere espressa anche in bit al secondo o byte al secondo. Le statistiche devono includere:

- la dimensione dei frame usati.

- il limite teorico del media per quel frame size.
- il tipo di protocollo usato nel test.

4.2.2 Latenza

Definizioni: *Per "store and forward devices" (dispositivi con code): l'intervallo di tempo fra l'ultimo bit di un frame in ingresso al dispositivo e il primo bit dello stesso frame in uscita.*
Per "bit forwarding devices": l'intervallo di tempo fra la fine del primo bit di un frame in ingresso al dispositivo e il primo bit dello stesso frame in uscita.
Procedura:

1. Determinare il Throughput per ogni frame size.
2. Inviare uno stream ad un dato frame size, Throughput e destinazione per circa 120 secondi.
3. Dovrebbe essere incluso un frame di test con un tag di identificazione dopo circa 60 secondi
4. Rilevare con Timestamp A il tempo in cui il suddetto frame è inviato e con Timestamp B il tempo in cui il suddetto frame è ritornato al tester.
5. Latenza = B - A (definita come in uno dei due casi specificati in precedenza)
6. Il test deve essere eseguito almeno 20 volte e prendere la media dei valori rilevati.

Il test dovrebbe essere essere eseguito con il frame di test inviato alla stessa destinazione dello stream totale e anche ad una nuova rete destinazione.

- *Formato dei risultati*: deve essere specificata la definizione di latenza usata (dipende dal tipo di dispositivo). I risultati vanno riportati in una tabella nelle cui righe abbiamo il frame size e nelle colonne abbiamo il rate a cui è eseguito il test e la latenza di ogni tipo di data stream testato.

4.2.3 Frame loss rate

- *Definizione*: *Percentuale di frame non inviati per scarsità di risorse*.

  Serve per misurare le performance in condizione di overload.

  Si misura nella percentuale di frame persi.

- *Procedura:*

1. Inviare uno specifico numero di frame ad uno specifico rate e contare i frame in uscita.
2. Frame Loss Rate = ((input_count - output_count)*100)/input_count
3. Il primo valore del data rate deve essere pari al 100% del massimo per frame size su quel media. Proseguire per 90% (secondo la granularità prefissata), 80%, ecc... fino ad ottenere 2 prove consecutive senza frame persi. La granularità massima è del 10% ma granularità più fini sono preferibili.

Formato dei risultati: Un grafico con il frame rate percentuale sull'ascissa ed il frame loss rate sull' ordinata. Nell' origine degli assi abbiamo 0% e all'estremità 100%. Possono essere usate linee multiple per indicare i valori a differenti frame size.

4.2.4 Back-to-Back frame

Definizione: E' il massimo numero di frame che è possibile inviare in un burst, ovvero il numero di frame di dimensione fissa inviati, su un dato mezzo trasmissivo, ad un tasso tale che il tempo di separazione tra 2 frame successivi è minimo. Si misura in Numero di N-ottetti nella trasmissione (burst).

Procedura:

1. Inviare un burst di frame con il minimo interframe gap
2. Contare in numero di frame inoltrati dal DUT
3. Se il numero di frame ricevuti è corretto incrementare il burst altrimenti diminuirlo.

Il Back-to-back value è il massimo numero di frame in un burst che il link può ricevere senza perdere frame. La prova deve durare almeno 2 secondi e deve essere ripetuta per 50 volte. Il Back-to-back value è la media dei valori ottenuti.

Formato dei risultati: Una tabella con una riga per ogni frame size del test, mentre nelle colonne abbiamo il risultante numero di frame per burst. Può anche essere inclusa la deviazione standard dalla media.

4.2.5 System recovery

Definizione: E' la velocità con cui un DUT recupera la connessione da una condizione di overload.

Procedura:

1. Determinare il Throughput per gli specificati frame size
2. Inviare uno stream a un rate del 110% del valore del Throughput o del massimo valore del media per almeno 60 secondi
3. Al tempo timestamp A ridurre il rate al 50 % e rilevare al timestamp B l'ultimo frame perso.
4. Il System recovery time è B - A
5. Ripetere la prova un certo numero di volte e riportarne la media

Formato dei risultati: una tabella con le righe per il frame size e nelle colonne il frame rate usato e la misura del recovery time.

4.2.6 Handoff Test

Definizione: Per handoff s'intende il tempo impiegato da un nodo mobile per completare la procedura di handover spostandosi da un rete (Home Network/Foreign Network) ad un'altra (risp. Foreign Network/Home Network).

Durante l'Handover il Nodo Mobile esegue i processi, a livello di rete, che consentono il mantenimento, in maniera trasparente, di eventuali connessioni a livello di trasporto al variare della cella wireless a cui il MN è sottoposto (managed).

Procedura:

1. Determinare il Throughput per gli specificati frame size
2. Inviare dal client (MN) uno stream con framerate pari al 50% del Throughput trovato in precedenza per il determinato frame size al Server (AR/CN)
3. Dopo HO_LenA secondi (Fase A) eseguire un handoff per portare il MN in un'altra rete; continuare ad inviare pacchetti per altri HO_LenB secondi (Fase B).
4. Dalla Server rilevare il tempo in cui arriva l'ultimo pacchetto dello stream della Fase A. Sia questo istante indicato con TS_A.
5. Dal Server rilevare il tempo in cui arriva il primo pacchetto dello stream della Fase B, sia questo TS_B.
6. Handoff_Time = (TS_B – TS_A)

7. Ripetere la prova un certo numero di volte e riportarne la media e la varianza al variare del frame size.

Formato dei risultati: una tabella con le righe per il frame size e nelle colonne media e varianza dell'Handoff Time.

4.3 MIPv6: analisi passiva

In questa sezione viene proposta una metodologia di analisi di Mobile IPv6 sulla base dei (macro) processi di segnalazione che intervengono nelle operazioni eseguite fra nodi mobili, home agent ed access router. L'analisi della tempistica di tali processi su un testbed reale (non simulato) consentirà di individuare eventuali colli di bottiglia durante l'*handover*. La metodologia è stata ottenuta dallo studio dei file di log ottenuti con tcpdump poi confermata dalle indicazioni riportate nel *TAHI conformance test suite for Mobile IPv6* (www.tahi.org).

4.3.1 Considerazioni preliminari

Come già accennato la topologia utilizzata in questi test è quella riportata nel paragrafo 4.1.1. Gli scenari possibili di testing sono diversi in funzione della particolare operazione MIPv6 che si vuole analizzare. Tuttavia, in generale, il nodo mobile si sposta dalla propria *Home Network,* gestita dal proprio Home Agent, verso la *Visited Network.* Qui compie eventalmente delle operazioni MIPv6 per poi ritornare nella Home. Si suppone, per uniformità, che in tutti gli agenti di mobilità sia utilizzato lo stesso RADV_INTERVAL (intervallo di tempo in sec entro il quale viene scelto casualmente l'istante in cui verrà emanato il prossimo router advertisement. Valori di default per tale intervallo sono [1;3]sec).

4.3.1 Scenario 1: Il nodo mobile si sposta verso una rete straniera

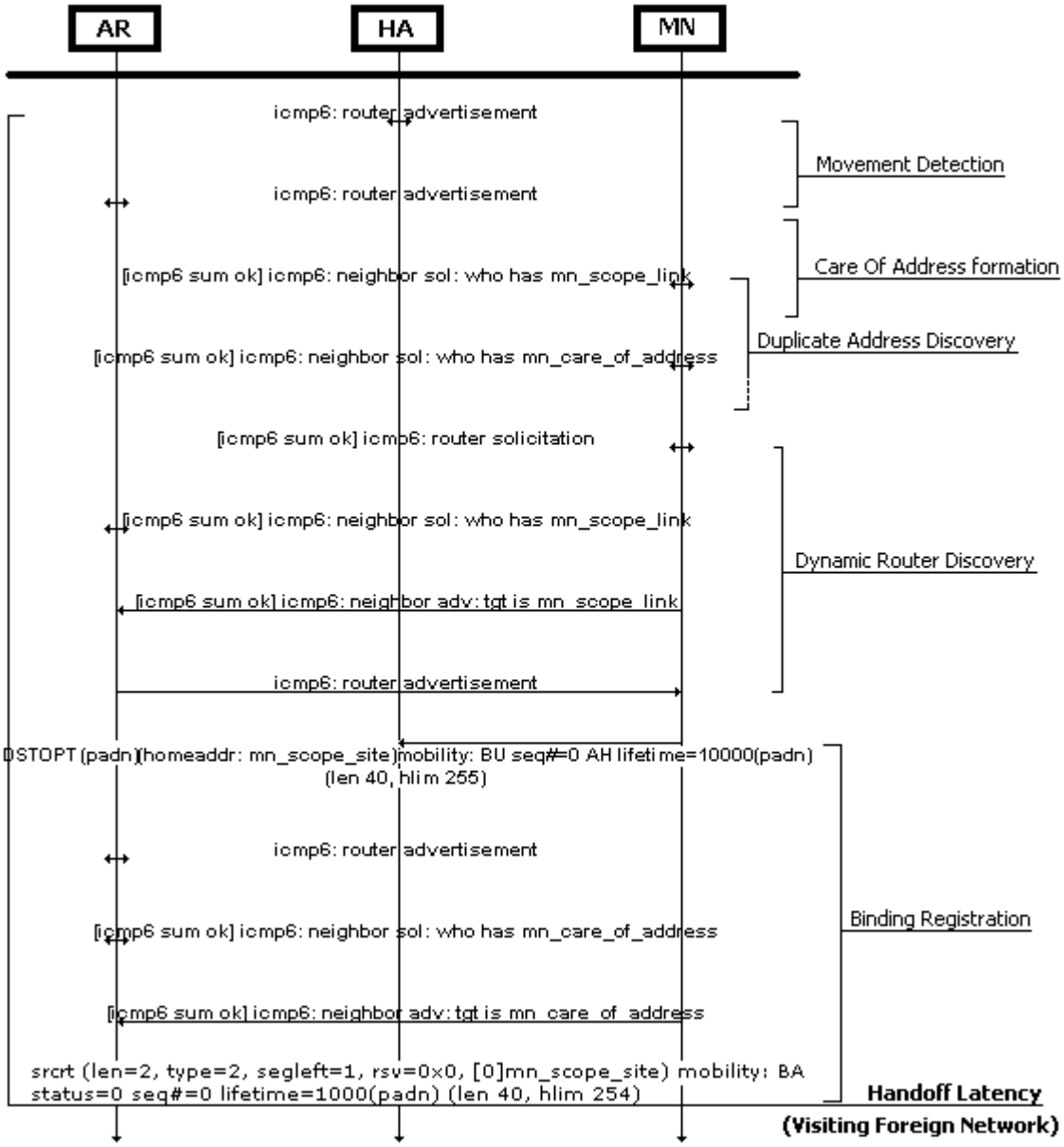


Figura 5: Fasi dell'Handoff durante la visita di una rete straniera (fonte: tcpdump log file)

Come si evince da questo *Sequence Diagram* la procedura di handoff è

costituita da diverse fasi, solo l'ultima però (Binding Registration) è attribuibile a Mobile IPv6, le precedenti sono tutte native di IPv6. Tuttavia, ai fini del calcolo globale dell'handover Latency, tutte queste fasi devono essere prese in considerazione e monitorate.

L'handoff a livello di rete inizia subito dopo quello a livello 2, questo evento è riconducibile al primo router advertisement ricevuto dall'Access Router (Movement Detection).

Subito dopo IPv6 preleva il nuovo prefisso di sottorete ed, affiancandolo al MAC address dell'interfaccia, generara un nuovo Care-Of Address (*anycast site local*).

Quest'ultimo (insieme all'indirizzo *link-local* della scheda di rete) deve (MUST) essere sottoposto alla procedura IPv6 DAD per assicurarsi circa l'univocità dell'indirizzo appena creato. Subito dopo il nodo mobile effettua la procedura qui chiamata *Dynamic Router Discovery* con cui si sollecitano i router presenti nella nuova rete circa la presenza del MN.

Finalmente è possibile avviare la procedura di *Binding Update*: il nodo mobile invia un binding request all'Home Agent richiedendo un lifetime di 10000 sec. Quest'ultimo accetta (Binding ACK) ma impone in lifetime del binding pari ad un decimo di quanto richiesto (1000 sec).

Di seguito vengono riportate le procedure utilizzate per monitorare i tempi delle precedenti fasi (alcune delle quali sono state raggruppate per essere calcolabili). Successivamente considereremo le fasi dell'Handoff durante il ritorno del MN presso la Home Network.

4.3.1.1 IPv6 Movement Detection Time (MDT)

Definizione: Per Movement Detection Time s'intende il tempo intercorrente fra l'ultimo Router Advertisement ricevuto dal precedente agente di mobilità (Home Agent o Access Router) ed il primo Router advertisement ricevuto dall'attuale agente di mobilità (rispettivamente Access Router o Home Agent).

Osserviamo che il MDT è comprensivo dell'handoff a livello di collegamente (Layer 2) tipicamente pari a 250 msec per 802.11b.

Procedura:

- `Sia HOFF_TIME una stima dell'handoff latency (esempio 5 sec) per un dato RADV_INTERVAL`
- `Eseguire un fissato numero (almeno 30) di handoff ogni 2*HOFF_TIME secondi`
- `Rilevare il valore medio e la deviazione standard dei tempi intercorrenti (B-A) tra due messaggi consecutivi del tipo:`

Sender	Messaggio	timestamp
Home Agent	**`icmp6: router advertisement H`**	A
Access Router	**`icmp6: router advertisement`**	B

Osserviamo che il RADV inviato dall'Home Agent, a differenza di quello emanato dall'Access Router è identificato dal flag H.

- *Formato dei risultati:* Ovviamente l'MDT dipenderà dalla frequenza con cui vengono emanati gli advertisement pertanto dovrebbero essere considerati almeno tre RADV_INTERVAL tra cui il minimo possibile. Una tabella con una riga per ciascun RADV_INTERVAL e due colonne per la media e la deviazione standard dell'MDT.

4.3.1.2 Router Discovery Time (RDT)

Definizione: Per Router Discovery Time s'intende il tempo intercorrente fra il primo Router advertisement ricevuto dall'attuale agente di mobilità (Access Router o Home Agent) ed il messaggio di Binding Update.

Osserviamo che il RDT è comprensivo delle fasi: COA formation e DAD procedures.

Procedura:

- Sia HOFF_TIME una stima dell'handoff latency (esempio 5 sec) per un dato RADV_INTERVAL
- Eseguire un fissato numero (almeno 30) di handoff ogni 2*HOFF_TIME secondi
- Rilevare il valore medio e la deviazione standard dei tempi intercorrenti (C-B) tra due messaggi consecutivi del tipo:

Sender	Messaggio	timestamp
Home Agent	**icmp6: router advertisement(chlim=64, H**	A
Access Router	**icmp6: router advertisement(chlim=64, pref**	B
...	... DAD, Router Solicitation, Router ADV ...	...
MN (COA)	DSTOPT: Mobility BU seq#=0 lifetime=10000	C

- *Formato dei risultati:* Una tabella con una riga per ciascun RADV_INTERVAL e due colonne per la media e la deviazione standard dei valori ottenuti.

4.3.1.3 Binding Registration Time (BRT)

Definizione: Per Binding registration Time s'intende il tempo intercorrente fra il messaggio di Binding Update ed il messaggio Binding Acknoldgment .
Procedura:

- Sia HOFF_TIME una stima dell'handoff latency (esempio 5 sec) per un dato RADV_INTERVAL
- Eseguire un fissato numero (almeno 30) di handoff ogni 2*HOFF_TIME secondi
- Rilevare il valore medio e la deviazione standard dei tempi intercorrenti (D-C) tra due messaggi consecutivi del tipo:

Sender	Messaggio	timestamp
MN (COA)	DSTOPT: Mobility BU seq#=0 lifetime=10000	C
Access Router	... **router advertisement** ...	...
Home Agent	SRCRT: Mobility BA seq#=0 lifetime=1000	D

- *Formato dei risultati:* Una tabella con una riga per ciascun RADV_INTERVAL e due colonne per la media e la deviazione standard dei valori ottenuti.

4.3.2 Scenario 2: Il nodo mobile ritorna nella Home Network

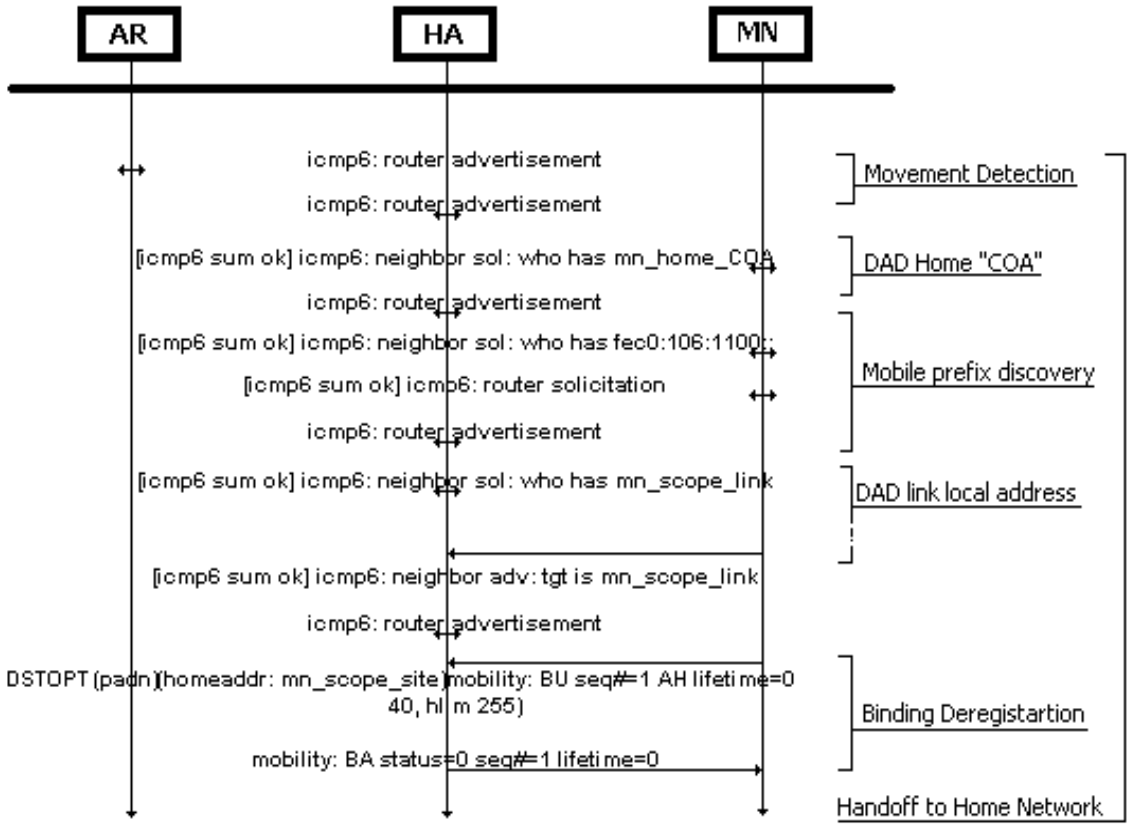


Figura 6: Returning to Home Network

Il ritorno nella Home Network è caratterizzato da fasi analoghe seppur con qualche differenza: dopo la rilevazione del movimento, IPv6 comunque và a generare un indirizzo *anycast site local* (nella figura è denominato Home-COA) fittizio in quanto non verrà mai utilizzato, dato che il nodo mobile è nella Home Network. Per quest'ultimo indirizzo verrà in ogni caso eseguita la procedura DAD (come per tutte le tipologie di indirizzi IPv6 dell'interfaccia del nodo mobile). Successivamente si passa alla fase di Binding Deregistration.

4.3.2.1 Home Movement Detection Time

Vale quanto detto nello scenario precedente, questo parametro non dipende dal percorso che compie il nodo mobile (Home -> Foreign o Foreign -> Home network) ma solo da RADV_INTERVAL.

4.3.2.2 Home Agent Discovery Time (HAD)

Definizione: Per Home Agent Discovery Time s'intende il tempo intercorrente fra il primo Router advertisement ricevuto dall'Home Agent ed il messaggio di Binding Update con lifetime nullo.

Osserviamo che il RDT è comprensivo delle fasi: Home COA formation, Home Prefix detection time e procedure DAD .

Procedura:

- Sia HOFF_TIME una stima dell'handoff latency (esempio 5 sec) per un dato RADV_INTERVAL
- Eseguire un fissato numero (almeno 30) di handoff ogni 2*HOFF_TIME secondi
- Rilevare il valore medio e la deviazione standard dei tempi intercorrenti (C-B) tra due messaggi consecutivi del tipo:

Sender	Messaggio	timestamp
Access Router	**icmp6: router advertisement(chlim=64,**	A
Home Agent	**icmp6: router advertisement(chlim=64, H**	B
...	... DAD, Router Solicitation, ...	...
MN (COA)	DSTOPT: Mobility BU seq#=1 AH lifetime=0	C

- *Formato dei risultati:* Una tabella con una riga per ciascun RADV_INTERVAL e due colonne per media e deviazione standard.

4.3.2.3 Binding Deregistration Time (BDT)

Definizione: Per Binding Deregistration Time s'intende il tempo intercorrente fra il messaggio di Binding Update con lifetime nullo ed il relativo messaggio Binding Acknoldgment .

Procedura:

- Sia HOFF_TIME una stima dell'handoff latency (esempio 5 sec) per un dato RADV_INTERVAL
- Eseguire un fissato numero (almeno 30) di handoff ogni 2*HOFF_TIME secondi
- Rilevare il valore medio e la deviazione standard dei tempi intercorrenti (D-C) tra due messaggi consecutivi del tipo:

Sender	Messaggio	timestamp
MN (COA)	Mobility BU seq#=1 lifetime=0	C
Access Router	... **router advertisement** ...	...
Home Agent	Mobility BA status=0 seq#=1	D

- *Formato dei risultati:* Una tabella con una riga per ciascun RADV_INTERVAL e due colonne per la media e la deviazione standard dei valori ottenuti.

4.3.3 Foreign Handover Latency Time (FHLT)

Definizione: Per Foreign Handover Latency Time s'intende il tempo intercorrente fra la creazione del Care-Of Address ed il messaggio di Binding Acknoldgment allorquando il nodo mobile si sposta dalla Home alla Foreign Network.

Questo è il parametro fondamentale di tutta l'analisi svolta ed è comprensivo di tutte le fasi costituenti l'handoff.

Procedura:

- Eseguire un fissato numero (almeno 30) di handoff ogni 10 secondi
- Rilevare il valore medio e la deviazione standard dei tempi intercorrenti (F-E) tra due messaggi consecutivi del tipo:

Sender	Messaggio	timestamp
MN (::)	**neighbor sol: who has mn_care_of_address**	E
...	... DAD procedure, BU request ...	...
Home Agent	SRCRT: Mobility BA seq#=0 lifetime=1000	F

- *Formato dei risultati:* Una tabella con una riga per ciascun RADV_INTERVAL e due colonne per la media e la deviazione standard dei valori ottenuti per il FHLT.

4.3.4 Home Handover Latency Time (FHLT)

Definizione: Per Home Handover Latency Time s'intende il tempo intercorrente fra la creazione dell'Home Care-Of Address ed il messaggio di Binding Acknoldgment allorquando il nodo mobile si sposta dalla Foreign alla Home Network.

Procedura:

- Eseguire un fissato numero (almeno 30) di handoff ogni 10 secondi
- Rilevare il valore medio e la deviazione standard dei tempi intercorrenti (F-E) tra due messaggi consecutivi del tipo:

Sender	Messaggio	timestamp
MN (::)	**neighbor sol: who has mn_home_COA**	E
...	... DAD procedure, BU request ...	...
Home Agent	**mobility: BA status=0 seq#=1 lifetime=0**	F

- *Formato dei risultati:* Una tabella con una riga per ciascun RADV_INTERVAL e due colonne per la media e la deviazione standard dei valori ottenuti per il HHLT.

4.3.4 Scenario 3: Routing Optimization

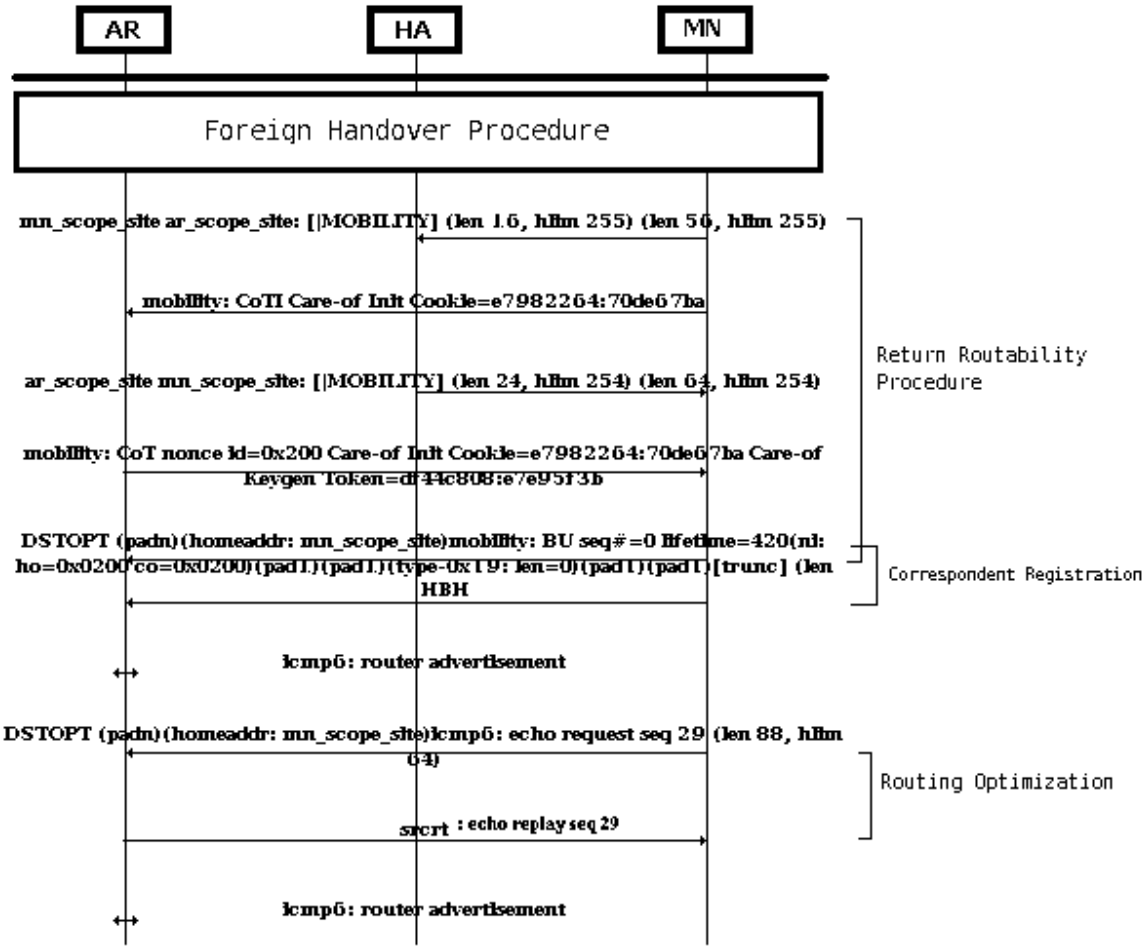


Figura 7: Return routability Procedure e Routing Optimization

In questo scenario il nodo mobile, inizialmente nella home network, sta comunicando (*ping6 continuo*) con un Correspondent Node (in questo caso il router della *Foreign Network* ovvero l'Access Router) attraverso l'usuale routing effettuato dall'Home Agent.

Dopodichè si sposta nella Foreign Network, effettua l'handover, come

descritto nelle pagine precedenti, e rileva l'esistenza di un persorso migliore per continuare a pingare con l'Access Router: può comunicare direttamente senza passare dall'Home Agent. A questo punto si avvia la Return Routability Procedure per realizzare una connessione diretta e *sicura* con l'attuale Correspondent Node.

Tale procedura ha un tempo (RRPT) che si aggiunge al *Foreign Handover latency Time* (FHLT) necessario per ristabilire la comunicazione (in questo caso un semplice *echo request ogni secondo*).

L'operazione di Correspondent Registration è effettuta con il solo Binding Update Request (il riscontro in questo caso è opzionale come da RFC3775).

Osserviamo che, per motivi di sicurezza, nel caso di *Correspondent Node Registra*tion, il lifetime del Binding è meno della metà (420 sec) di quello richiesto tramite *Home Agent Registration* (1000 sec).

4.3.4.1 Foreign Correspondent Registration Time (FCRT)

Definizione: Per Foreign Correspondent Registration Time s'intende il tempo intercorrente tra il completamento (Binding Ack) dell'Home Agent Registration ed il messaggio di Binding Update inviato al Correspondent Node allorquando il nodo mobile si sposta dalla Home alla Foreign Network.
Questo tempo è comprensivo del ritardo dovuto alla procedura di Return Routability.

Procedura:

- Eseguire un fissato numero (almeno 30) di handoff ogni 10 secondi
- Rilevare il valore medio e la deviazione standard dei tempi intercorrenti (H-G) tra due messaggi consecutivi del tipo:

Sender	Messaggio	timestamp
Home Agent	**mobility: BA status=0 seq#=0 lifetime=1000**	G
...	... HoTI, CoTI, HoT, CoT ...	...
MN (COA)	**mobility: BU seq#=0 lifetime=420**	H

- *Formato dei risultati:* Una tabella con una riga per ciascun RADV_INTERVAL e due colonne per la media e la deviazione standard dei valori ottenuti per il RRPT.

4.3.4.2 Home Correspondent Deregistration Time (HCDT)

Definizione: Per Home Correspondent Deregistration Time s'intende il tempo intercorrente tra il completamento (Binding Ack) dell'Home Agent Deregistration ed il messaggio di Binding Update inviato al Correspondent Node allorquando il nodo mobile si sposta dalla Foreign alla Home Network.
Questo tempo è comprensivo del ritardo dovuto alla procedura di Return Routability.

Procedura:

- Eseguire un fissato numero (almeno 30) di handoff ogni 10 secondi.
- Rilevare il valore medio e la deviazione standard dei tempi intercorrenti (L-I) tra due messaggi consecutivi del tipo:

Sender	Messaggio	timestamp
Home Agent	**BA status=0 seq#=1 lifetime=0**	I
...	... HoTI, CoTI, HoT, CoT ...	...
MN	**BU seq#=1 lifetime=0**	L

- *Formato dei risultati:* Una tabella con una riga per ciascun RADV_INTERVAL e due colonne per la media e la deviazione standard dei valori ottenuti per il RRPT.

5 Testbed

Il testbed utilizzato è il seguente:

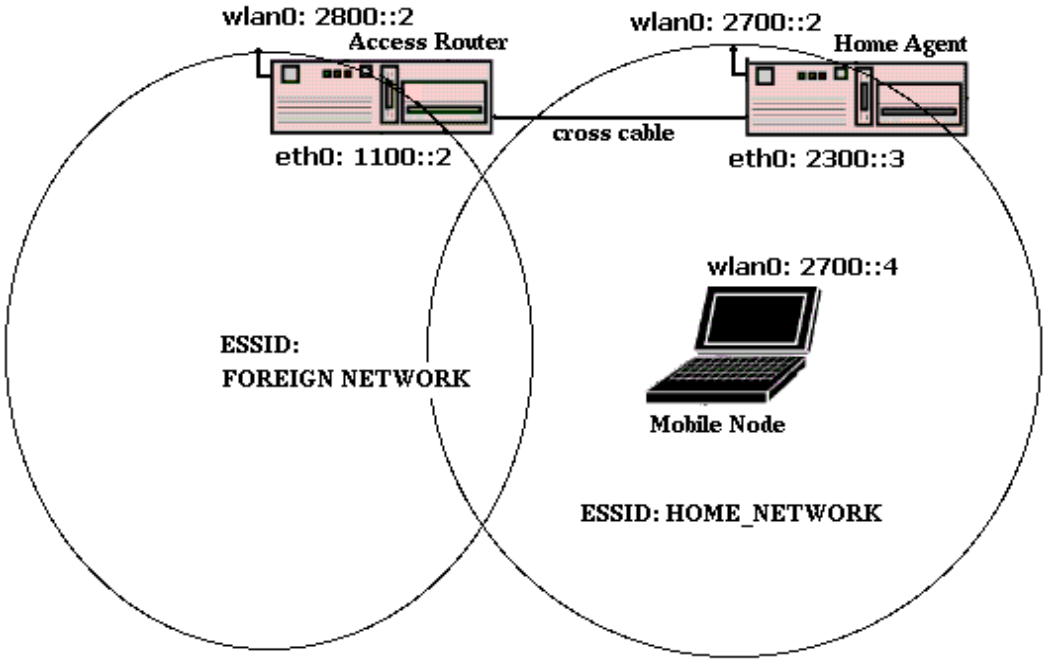


Figura 8: MIPv6 testbed

Con l'obiettivo di minimizzare i link tra i componenti, l'Home Agent è direttamente connesso, tramite un cavo incrociato (la nostra "Internet"), all'Access Router della Foreign Network. Entrambi fungono anche da Access Point grazie all'utilizzo di schede di rete Wireless PCI e driver HostAP (http://hostap.epitest.fi). L'Access Router è stato configurato anche come Correspondent Node MIPv6-enabled. Per semplicità nella figura è stata omessa la parte iniziale comune a tutti gli indirizzi IPv6 *scope-site*: (`fec0:106:`).

5.1 MIPv6 for Linux

Mobile IPv6 per Linux (MIPL) è una implementazione originariamente creata come progetto del corso di *Reti di Computer* presso la Helsinki University of Technology (HUT), con l'obiettivo di creare un prototipo di implementazione del protocollo Mobile IPv6 in ambiente Linux. Successivamente il progetto è stato sviluppato nel contesto del GO/Core project presso l' *HUT Telecommunications and Multimedia Lab*. E' una implementazione open source disponibile sotto licenza GNU GPL.

In Linux lo stack IPv6 standard è già incluso nel kernel ma per supportare la mobilità necessità di una patch la quale comporta la sua ricompilazione.

MIPL v1.1 è la versione dell'implementazione analizzata in questa tesi, questa si basa sulle specifiche del RFC3775 e richiede il Linux kernel v. 2.4.26.

La nuova versione MIPL 2.0 RC1 richiede il Linux kernel v. 2.6.8.1 è già disponibile anche se i lavori di ultimazione sono ancora in corso.

La patch mipv6-1.1-v2.4.26tar.gz ed il tool di monitoraggo (mipdiag) è disponibile presso http://www.mobile-ipv6.org

5.2 Setup

1. Decomprimere il pacchetto mipv6-1.1-v2.4.26tar.gz in /usr/src. Verrà creata la directory /usr/src/mipv6-1.1-v2.4.26

2. cd /usr/src/linux

3. Applicare la patch:

```
shell>> patch -p1 < /usr/src/mipv6-1.1-v2.4.26/mipv6-1.1-v2.4.26.patch
```

4. Riconfigure il Kernel (`make xconfig`) settando le seguenti opzioni:

```
CONFIG_EXPERIMENTAL=y
CONFIG_SYSCTL=y
CONFIG_PROC_FS=y
CONFIG_MODULES=y
CONFIG_NET=y
CONFIG_NETFILTER=y
CONFIG_UNIX=y
CONFIG_INET=y
CONFIG_IPV6=m
CONFIG_IPV6_SUBTREES=y
CONFIG_IPV6_IPV6_TUNNEL=m
CONFIG_IPV6_MOBILITY=m
CONFIG_IPV6_MOBILITY_MN=m
CONFIG_IPV6_MOBILITY_HA=m
CONFIG_IPV6_MOBILITY_DEBUG=y
```

5. Compiliamo il Kernel ed i moduli di Mipv6:

```
Shell>> make dep && make clean && make bzImage modules
modules_install
```

6. Installare il nuovo kernel:
Copiare il file `bzImage` sotto `/boot/vmlinuz-2.4.24` e modificare il file `/etc/lilo.conf` in modo opportuno, quindi eseguire

```
shell>> lilo
```

7. Aggiungere il device MIPv6:

```
shell>>mknod /dev/mipv6_dev c 0xf9 0
```

8. Installare il tool mipdiag:

```
shell>>cd /usr/local/src/mipv6-1.1-v2.4.26
Shell>>./configure
Shell>>make && make install
```

9. Installare il router advertisement daemon (sull'HA ed AR) disponibile presso http://v6web.litech.org/radvd/

10. Riavviare ed eseguire:`/etc/init.d/mobile-ip6 start`

5.3 Configurazione Home Agent

- HA_setup.sh

```
#!/bin/sh
#------------ STARTING MIPV6 HOME AGENT
modprobe ipv6
iwconfig wlan0 mode Master ESSID HOME_NETWORK channel 2 enc off
ifconfig wlan0 inet6 add fec0:106:2700::2/64
echo "1" > /proc/sys/net/ipv6/conf/wlan0/forwarding
echo "0" > /proc/sys/net/ipv6/conf/wlan0/autoconf
echo "0" > /proc/sys/net/ipv6/conf/wlan0/accept_ra
echo "0" > /proc/sys/net/ipv6/conf/wlan0/accept_redirects
ifconfig eth1 inet6 add fec0:106:2300::2/64
ip route add fec0:106:1100::/64 via fec0:106:2300::1
./mobile-ip6 restart
# ----------------------------------------
```

- Home Agent setting in `/etc/network-mip6.conf`

```
FUNCTIONALITY=ha
MIN_TUNNEL_NR=1
MAX_TUNNEL_NR=5
TUNNEL_SITELOCAL=yes
```

- Home Agent RADV Daemon Configuration:

`/etc/radvd.conf`

```
interface wlan0
{
        AdvSendAdvert on;
        MaxRtrAdvInterval 3;
        MinRtrAdvInterval 1;
        AdvIntervalOpt off;
        AdvHomeAgentFlag on;
        HomeAgentLifetime 10000;
        HomeAgentPreference 20;
        AdvHomeAgentInfo on;
        prefix fec0:106:2700::2/64
        {
                AdvRouterAddr on;
                AdvOnLink on;
                AdvAutonomous on;
                AdvPreferredLifetime 10000;
                AdvValidLifetime 12000;};};
```

5.3 Configurazione Access Router

- AR_setup.sh

```
#!/bin/sh

#------------- STARTING ACCESS ROUTER MIPV6
iwconfig wlan0 mode master essid FOREIGN_NETWORK channel 5 enc off
ifconfig wlan0 inet6 add fec0:106:2800::2/64
ifconfig eth0 inet6 add fec0:106:1100::2/64
echo "1" > /proc/sys/net/ipv6/conf/wlan0/forwarding
echo "0" > /proc/sys/net/ipv6/conf/wlan0/autoconf
echo "0" > /proc/sys/net/ipv6/conf/wlan0/accept_ra
echo "0" > /proc/sys/net/ipv6/conf/wlan0/accept_redirects
ip route add ::/0 via fec0:106:1100::1
./mobile-ip6 restart

# ----------------------------------------
```

- Access Router setting in /etc/network-mip6.conf

```
FUNCTIONALITY=cn
MIN_TUNNEL_NR=1
MAX_TUNNEL_NR=5
TUNNEL_SITELOCAL=yes
```

- Access Router RADV Daemon Configuration

```
#This interface has HA support disabled
interface wlan0
{
      AdvSendAdvert on;
      MaxRtrAdvInterval 3;
      MinRtrAdvInterval 1;
      AdvIntervalOpt on;
      AdvHomeAgentFlag off;
      prefix fec0:106:1100::/64
        {
            AdvRouterAddr on;
            AdvOnLink on;
            AdvAutonomous on;
      };
};
```

5.4 Configurazione Nodo Mobile

- MN_setup.sh

```
#!/bin/sh

#------------- STARTING MIPV6 MOBILE NODE
#-- Paolo Cavone, 04/02/2005
# -Setting MN at HOME NETWORK
/usr/local/sbin/iwconfig wlan0 mode Manage essid HOME_NETWORK channel 2
/usr/local/sbin/iwconfig wlan0 ap 00:05:5D:5B:F8:88 enc off nickname
MobileNode
ifconfig wlan0 inet6 add fec0:106:2800::4/64
echo "0" > /proc/sys/net/ipv6/conf/wlan0/forwarding
echo "1" > /proc/sys/net/ipv6/conf/wlan0/autoconf
echo "1" > /proc/sys/net/ipv6/conf/wlan0/accept_ra
echo "1" > /proc/sys/net/ipv6/conf/wlan0/accept_redirects
ip route add ::/0 via fec0:106:2700::2
./mobile-ip6 restart
# ----------------------------------------
```

- Access Router setting in /etc/network-mip6.conf

```
FUNCTIONALITY=mn
TUNNEL_SITELOCAL=yes
MIN_TUNNEL_NR=1
MAX_TUNNEL_NR=3
HOMEDEV=mip6mnha1
HOMEADDRESS=fec0:106:2800::4
HOMEAGENT=fec0:106:2800::2
```

5.5 Il tool mipdiag

E' uno strumento per monitorare numerosi parametri relativi a Mobile IPv6, ovviamente la valorizzazione di tali paramentri dipenderà dalla modalità di funzionamento. Ad esempio eseguito sull'Home Agent si ha:

```
shell>>mipdiag
Mobile IPv6 Statistics
NEncapsulations           : 36
NDecapsulations           : 104656
NBindRefreshRqsRcvd       : 0
NHomeTestInitsRcvd        : 0
NCareofTestInitsRcvd      : 0
NHomeTestRcvd             : 0
NCareofTestRcvd           : 0
NBindUpdatesRcvd          : 9
NBindAcksRcvd             : 0
NBindNAcksRcvd            : 0
NBindErrorsRcvd           : 0
NBindRefreshRqsSent       : 0
NHomeTestInitsSent        : 0
NCareofTestInitsSent      : 0
NHomeTestSent             : 0
NCareofTestSent           : 0
NBindUpdatesSent          : 0
NBindAcksSent             : 5
NBindNAcksSent            : 4
NBindErrorsSent           : 0
NBindUpdatesDropAuth      : 0
NBindUpdatesDropInvalid   : 0
NBindUpdatesDropMisc      : 0
NBindAcksDropAuth         : 0
NBindAcksDropInvalid      : 0
NBindAcksDropMisc         : 0
NBindRqsDropAuth          : 0
NBindRqsDropInvalid       : 0
NBindRqsDropMisc          : 0
```

6 -Misure dei Test RFC2544

Per le misure dei parametri indicati nel RFC2544 e rilevanti in questa tesi è stato utilizzato un tool sviluppato presso il nostro Laboratorio dall'Ing. Antonio Bianco durante la sua Tesi relativa alle performance di Mobile Ipv4. Il tool, denominato *NBMark,* ha una architettura client-server, in particolare l'applicazione lato client invia stream di test UDP (di differenti *frame size*) al Server il quale, conta i datagrammi pervenuti, e riponde al client con le statistiche rilevate.

I parametri considerati sono quelli indicati nella Sezione 4.2, in particolare:

- Throughput
- Frame Loss rate
- Back-to-Back Test
- System Recovery Time

Con i mezzi a disposizione è stato impossibile effettuare il test sulla *latenza* come descritto nell'RFC 2544. Il test in questione necessita di una perfetta sincronizzazione fra il client ed il server.

Come indicato nel RFC2544 i risultati relativi al Throughput devono essere comparati con i rispettivi valori teorici per un dato framesize e protocollo di livello 2 (Ethernet o 802.11b). Il suddetto tool aggiunge un ulteriore test quelli elencati ed specifico per misurare la bontà della mobilità di una connessione wireless: l' *Handoff Test*.

Nei due paragrafi successivi calcoleremo i valori teorici del Throughput per i frame-size opportuni, compatibili con Header IPv6, e protocolli di data-link utilizzati.

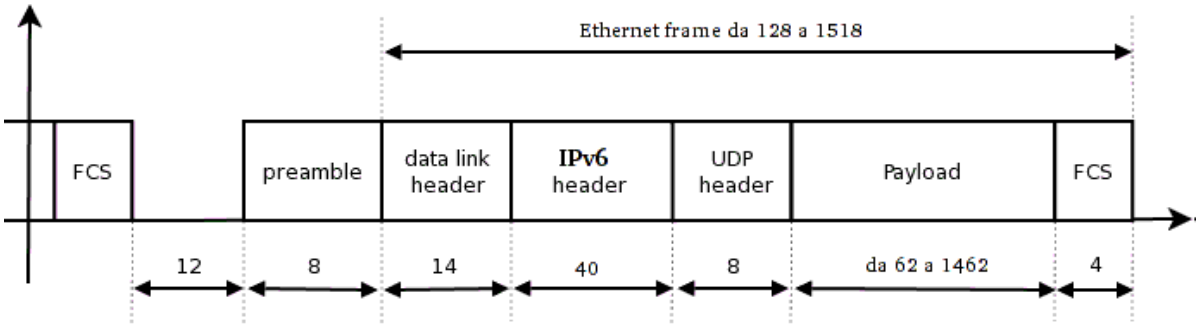


Figura 9: Ipv6 Frame size

I Frame Size (FS) da utilizzare secondo RFC2544 sono:

128, 256, 512, 769, 1024, 1280, 1518

Il frame size minimo indicato nel'RFC2544 è 64, nel caso di IPv6 però non è utilizzabile visto che con i soli Headers+FCS si arriva a 66 Byte.

Il frame size massimo utilizzato è 1518, questo supera il consentito ma è utile per analizzare gli effetti della frammentazione. I Payload IPv6 corrispondenti sono:

Frame Size FS	***Payload Ipv6 su Ethernet*** FS-(14+40+4)=FS -58	***Payload IPv6 su 802.11b*** FS-(30+40+4)= FS -74
128	70	54
256	198	182
512	454	438
769	711	695
1024	966	950
1280	1222	1206
1518	1460	1386

6.1 Throughput teorico per Ethernet 100Mbps

Nel caso reale ci sono molti fattori che influiscono sulle prestazioni in modo negativo. Quindi, poiché siamo interessati a delle performance ideali, i risultati ottenuti valgono sotto le seguenti assunzioni:

- Bit error rate (BER) uguale a zero (collegamento ideale)
- Non ci sono perdite per collisione
- Non c è perdita di pacchetti a causa del buffer overflow del nodo ricevente
- Il nodo trasmettente ha sempre sufficienti pacchetti da spedire
- Il livello MAC non usa la frammentazione
- Non sono considerati i pacchetti dovuti al management

Tenendo presente che su Ethernet 100Mbps l'inter-frame Gap (IF_GAP) è pari a 0,96usec ovvero uno spazio pari a 12byte (0,96usec = 96bit*100Mbps => 96bit = 12 Byte) si ha che il Periodo per un frame in bit (PF) è dato da:

```
PF = (FS+IF_GAP+Preamble)*8 = (FS+12+8)*8
```

da cui la seguente tabella dei valori teorici per i Throughput terorici di Ethernet 100Mbps:

Frame size (FS)	***Payload IPv6 (FS-58)***	***Periodo per Frame (PF)***	***Throughput Teorico (100Mbps/PF)***
128	70	1.184	84.459
256	198	2.208	45.289
512	454	4.176	23.496
769	711	6.312	15.862
1.024	966	8.352	11.973
1.280	1.222	10.400	9.615
1.518	1.460	12.304	8.127

Tabella 9: Throughput terorici di Ethernet 100Mbps

6.2 Throughput teorico per 802.11b (CSMA/CA)

Il massimo throughput teorico (TMT) per 802.11b, con payload IPv6, è noto tramite la formula (Cfr. *"Theoretical Maximum Throughput of IEEE 802.11 and its application*"):

$$TMT(x) = \frac{8x}{a(x+IP\_HEADER)+b} \times 10^6 \, bps$$

Throughput 802.11b+IPv6

dove, nel caso di CSMA/CA-DSSS, i parametri sono: a=0.72727, b=890.73, IP_HEADER=40, *x* è il Payload di 802.11b in byte, l'andamento è:

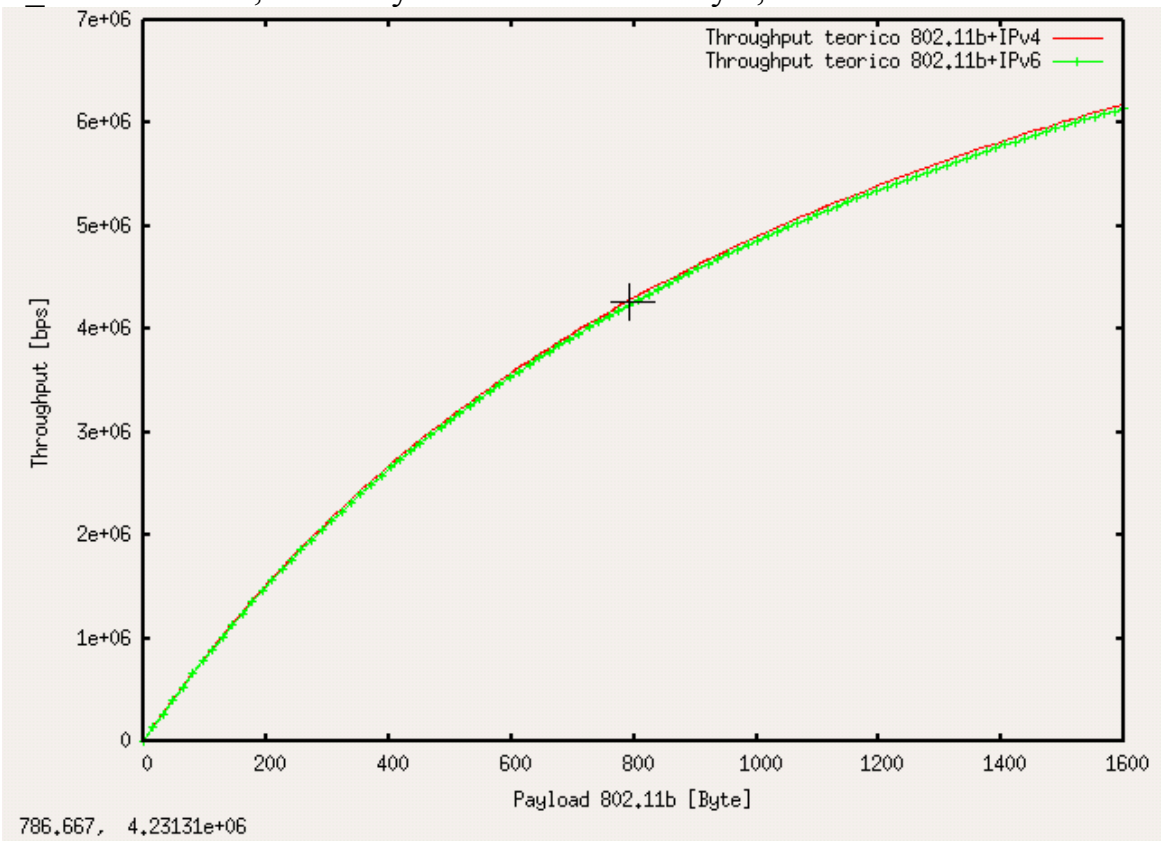


Figura 10: Andamento del Throughput Teorico su 802.11b

6.3 Risultati dei test RFC2544

6.3.1 Configurazione Hardware

- Home Agent ed Access Router sono due macchine identiche con le seguneti caratteristiche:

CPU	*intel pentium3 700Mhz*
Mainboard Chipset	*VIA Technologies, Inc. VT8605 [ProSavage PM133] (rev 81)*
Memoria RAM	*512 Mb sdr*
Network card	*Intersil Corporation Prism 2.5 Wavelan chipset (rev 01) (wireless 802.11b)*
Network card	*3Com Corporation 3c905B 100BaseTX [Cyclone] (rev 64) (Ethernet)*
OS	*Linux Debian (kernel 2.4.26)*
Ambiente grafico	*nessuno*

- La configurazione del Nodo Mobile:

Modello	*Notebook: Fujitsu Siemens Amilo A*
CPU	*AMD Athlon XP Mobile – 2,4 GHz*
Memoria RAM	*512 Mb sdr*
Network card	*D-Link DWL-650 11Mbps WLAN Card Version 01.02*
Network card	*Ethernet controller: Realtek Semiconductor Co., Ltd. RTL-8139/8139C/8139 C+ (rev 10)*
OS	*Linux Slackware (kernel 2.4.26)*
Ambiente grafico	*KDE 3.2*

6.3.2 Configurazione del Tool NBMark

Il tool NBMark implementa i test descritti nella Sezione 4.2 sulla base delle indicazioni riportate nel RFC 2544, l'applicazione client si basa sul seguente file di configurazione (bench.conf):

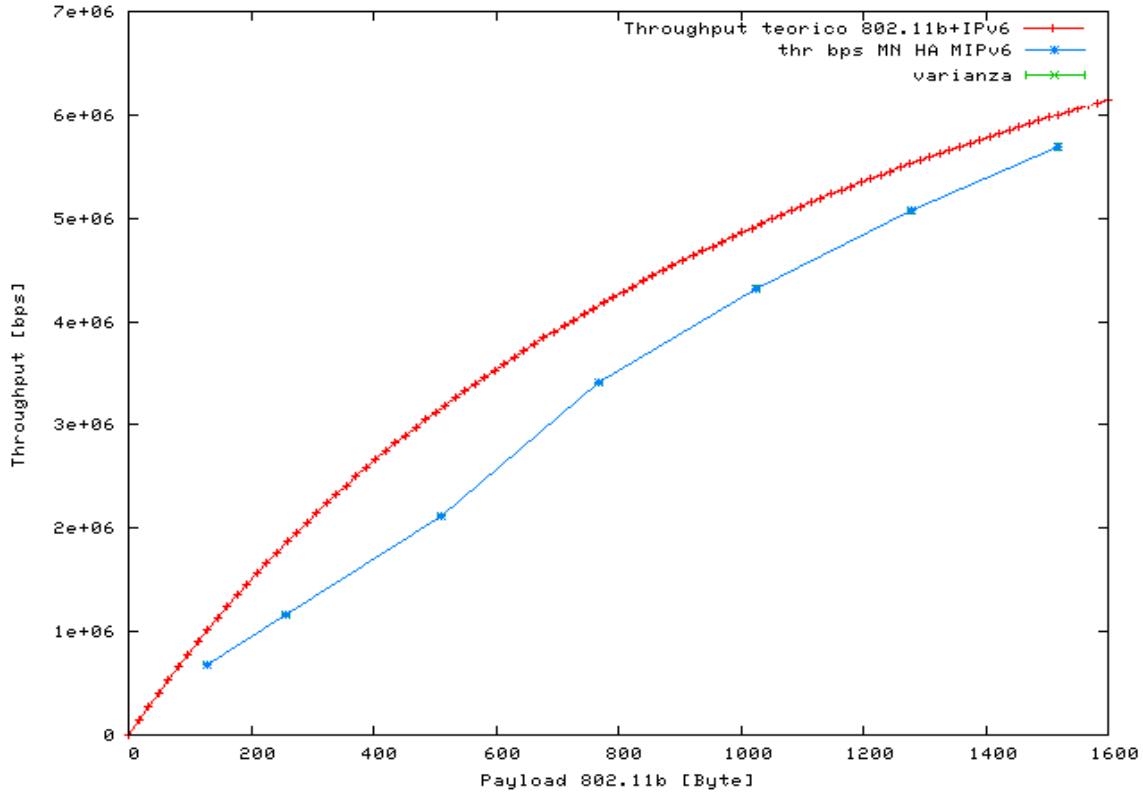
```
# ----------------- Configurazione Comune a tutti i Test -------
# - Address for server
#serv_address = "fec0:106:2700::2"  # (Home Agent)
serv_address = "fec0:106:1100::1"   # (Access Router)
# this option MUST be set when IPv6 LINK LOCAL address is used
# iface = "wlan0"
#destination port on which the server is listening
PORT_ADDR = "35000"
#source port
PORT_LISTEN = "5000"
#file for printing results
result = "result_test.txt";

#definition of the options of common tests
# Number of tried for receive answers from the server for
# throug_start and throug_end messages
climaxtry = 30
#time of attempt between a trial and the successive one (1/100)
time_itrial = 5

# Tolerance of the interval of time of the trials
time_toll = 100 # 1000 = 1%

# The stream it is divided in small burst sendes every clocktime
# milliseconds minor is the clocktime better is the interframe gap
but
# she increases the overhead of the calculations
clocktime = 100         #from 10 to 100 step 10

# Interframe-Gap in bps between a frame and the next one
iframe_gap = 160        # 160 bit = 20 byte ethernet 100Mbps
# iframe_gap = 0        # local loop

#bitrate of the device on which the measures are executed
bitrate = 11000000      #802.11b max bitrate
#bitrate = 1000000
```

```
# Gap for the headers LV2-3-4 without payload
#header_gap = 66 # 14 + 4 (ethernet + FCS) + 40 (IPv6) + 8 (UDP)
#header_gap = 82 # 30 + 4 (802.11 + FCS) + 40 (IPv6) + 8 (UDP)
#header_gap = 62 # 14     (ethernet in local loop) + 40 (IPv6) + 8
(UDP)
header_gap = 66  # 14 + 4 (802.11 + FCS) + 40 (IPv6) + 8 (UDP)

#framesize definitions

size=128
size=256
size=512
size=768
size=1024
size=1280
size=1518

# Startup test in order to wake up the interface with
# the payload constant of 28 byte
st_test = 0             #0 disabled / 1 enables
st_len = 10             #lenght in seconds


# Throughput test options definition

thr_test = 0            #0 disabled / 1 enables
thr_bitrate = 15000000  #bitrate for Throughput test only.
                        #If it is regulated to 0, then the
                         # thr_bitrate ones
                         # will be regulated equal to option of the
bitrate ones

thr_ntrial = 5          # number of iterances on which it is made
                         # the average
thr_toll = 10000        # tolerance of the measure (influences the
number
                         # of trial)
thr_len = 10            # lenght in second of every single trial
```

```
#Back-to-back test options definition

b2b_test = 0            #0 disabled / 1 enables (RFC 2544)
b2b_ntrial = 15         #number of iterances on which it is made the
                         # average
b2b_toll = 100          #for b2b test RFC 2544: tolerance of the
measure (influences the number of trial), value "0" is not valid
b2b_len = 2             #lenght in second of every single trial
                         #an other various test from that one of RFC
2544
b2bquick_test = 0       #another b2b test: 0 disabled / 1 enables
(not                     #RFC 2544)
b2bquick_ntrial = 15    #number of iterances on which it is made the
average


#Frame loss rate test options definition

flr_test = 1            #0 disabled / 1 enables
flr_ntrial = 10         #number of iterances on which it is made the
                         # average
flr_len = 60            #lenght in second of every single trial
flr_gr = 10             #granularity of the measurements:  from 1 to
10

#System recovery  options definition (Throughput test MUST be
anabled!!!)

sr_test = 0             #0 disabled / 1 enables
sr_ntrial = 10          #number of iterances on which it is made the
average
sr_lena = 30            #first stream lenght (110% Throughput value)
sr_lenb = 3             #second stream lenght (50% Throughput value)

#
#Hand-off  options definition (Throughput test MUST be anabled!!!)

ho_test = 0             #0 disabled / 1 enables
ho_ntrial = 1           #number of iterances on which it is made the
average
ho_lena = 10            #first stream lenght (110% Throughput value)
ho_lenb = 10            #second stream lenght (50% Throughput value)
iscript = "init_net.sh" #script to initialize the network
hscript = "handoff.sh"  #script to achieve the hand-hoff
```

6.4 Throughput at Home

- Grafico in bps:

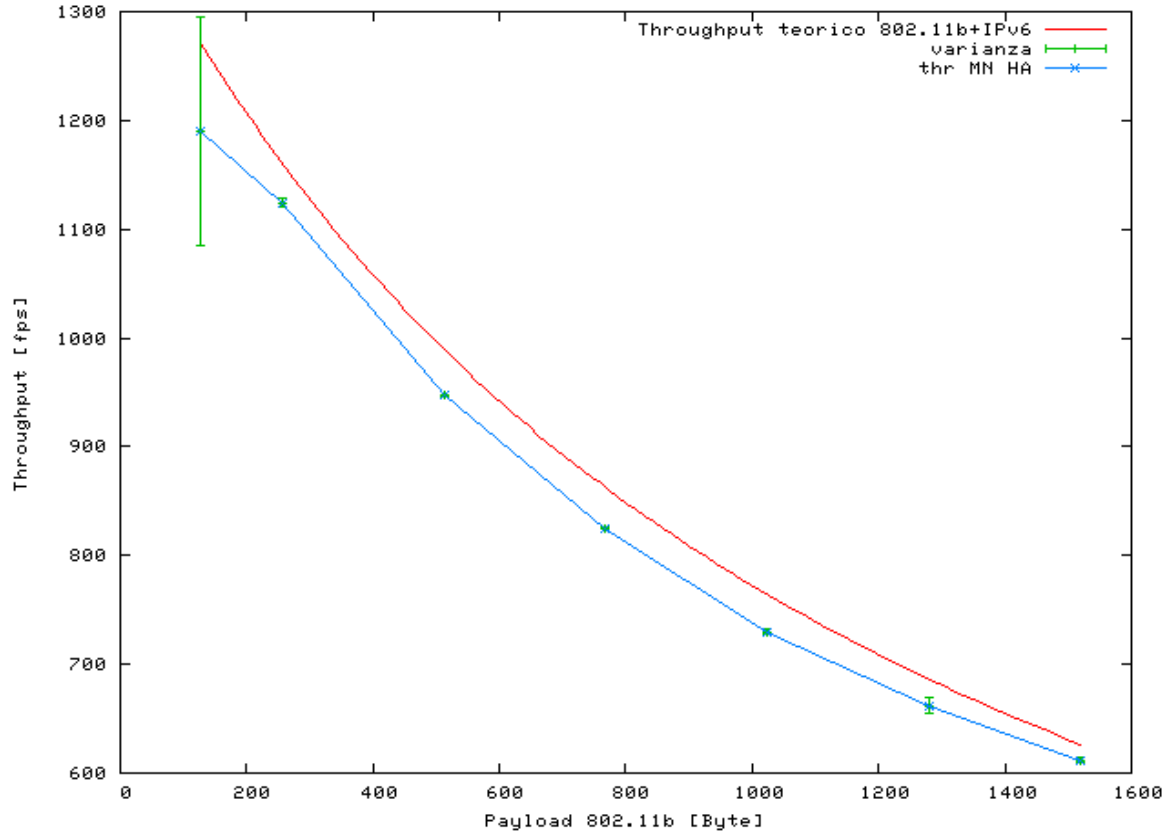


- La Tabella dei risultati:

Payload IPv6	*Thr (bps)*	*Varianza*	*TMT (bps)*	*Diff %*
54	684.079	5.859,200218	998.636	31,50
182	1.161.619	10.961,987511	1.850.250	37,22
438	2.122.556	7.176,515519	3.152.390	32,67
695	3.405.760	11.718,800246	4.160.670	18,14
950	4.322.750	23.437,250022	4.909.170	11,95
1206	5.069.820	21.528,565373	5.516.370	8,09
1386	5.685.056	34.165,570884	6.020.460	5,57

6.5 Throughput at Home (Client: MN, Server HA)

- Grafico in fps:

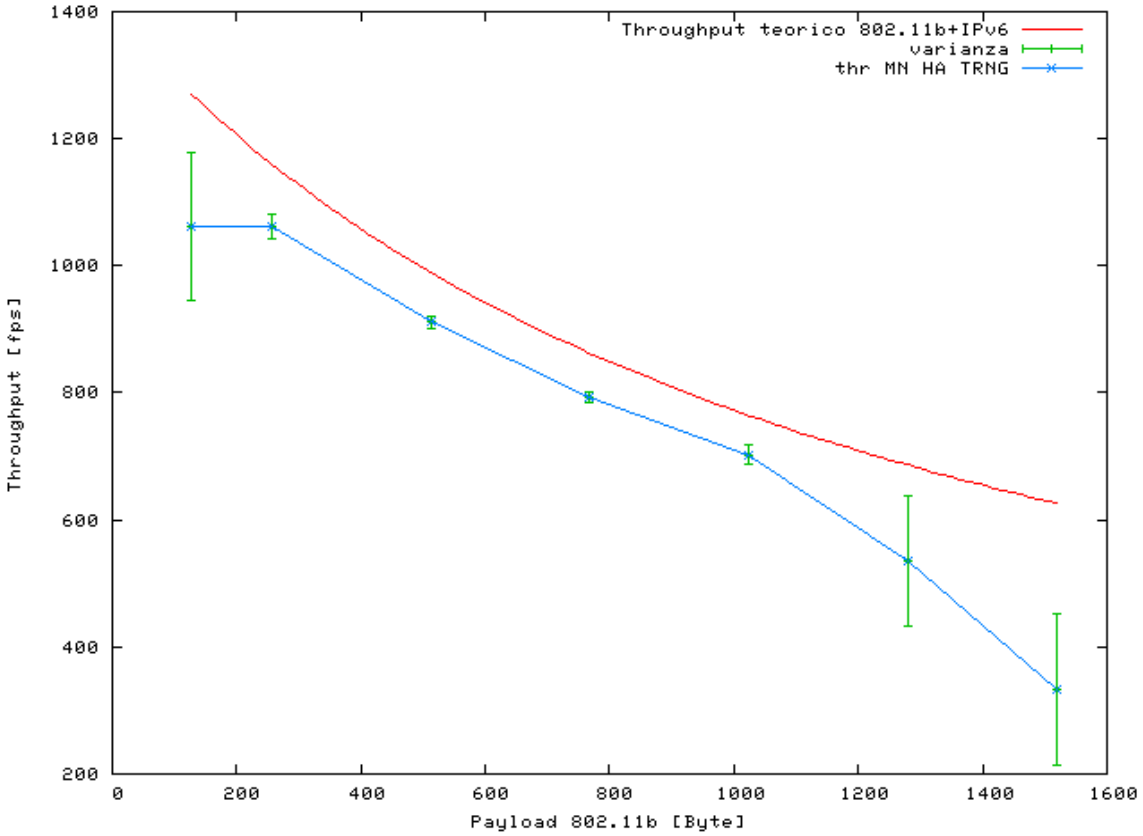


- La Tabella dei risultati:

Payload IPv6	*Thr (fps)*	*Varianza*	*TMT (fps)*	*Diff %*
54	1.190	104,502632	1.270	6,30
182	1.124	3,847077	1.158	2,94
438	947	0,774597	984	3,76
695	825	1,000000	859	3,96
950	730	2,097618	764	4,45
1206	662	7,028513	686	3,50
1386	612	2,529822	624	1,92

6.6 Throughput at Foreign Network (Client: MN, Server AR via HA)

- Grafico (in fps) nel caso di *Triangular Routing*:

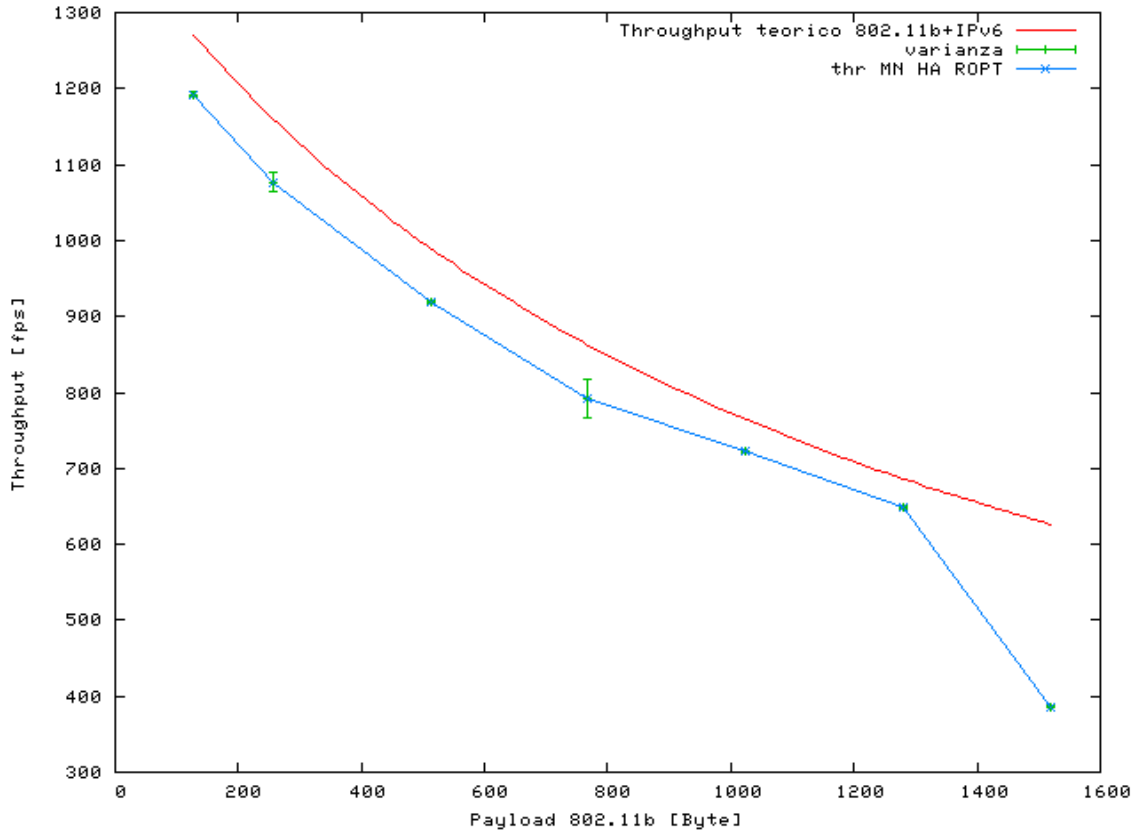


- La Tabella dei risultati:

Payload IPv6	*Thr (fps)*	*Varianza*	*TMT (fps)*	*Diff %*
54	1.062	117,446158	1.270	16,38
182	1.062	19,514097	1.158	8,29
438	911	10,677078	984	7,42
695	794	8,024961	859	7,57
950	703	180.36.00	764	7,98
1206	535	101,534231	686	22,01
1386	333	120,414214	624	46,63

6.7 Throughput at Foreign Network (Client: MN, Server AR)

- Grafico (in fps) nel caso di *Routing Optimization*:

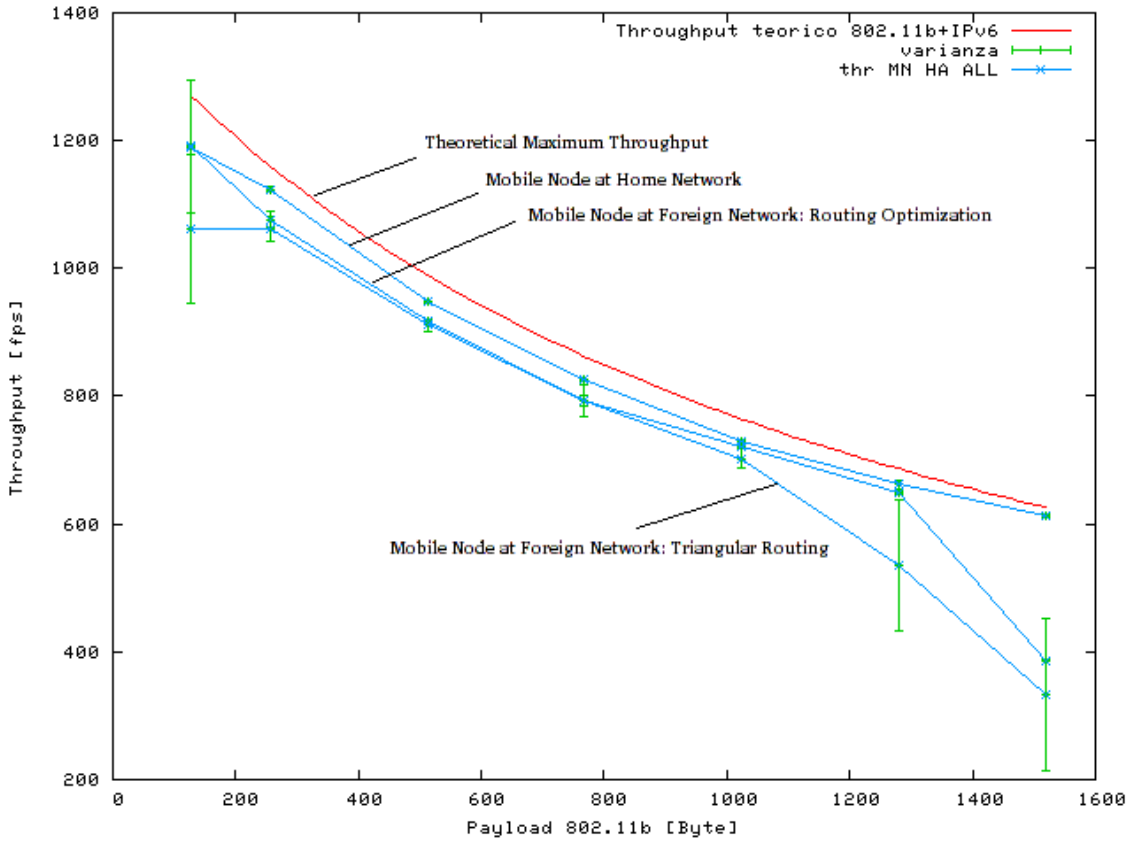


- La Tabella dei risultati:

Payload IPv6	*Thr (fps)*	*Varianza*	*TMT (fps)*	*Diff %*
54	1.192	3,872983	1.270	6,14
182	1.077	11,755850	1.158	6,99
438	919	1,414214	984	6,61
695	792	25,151541	859	7,80
950	722	1,612452	764	5,50
1206	649	2,097618	686	5,39
1386	386	1,414214	624	38,14

6.8 Confronto dei Throughput

- Grafico (in fps):

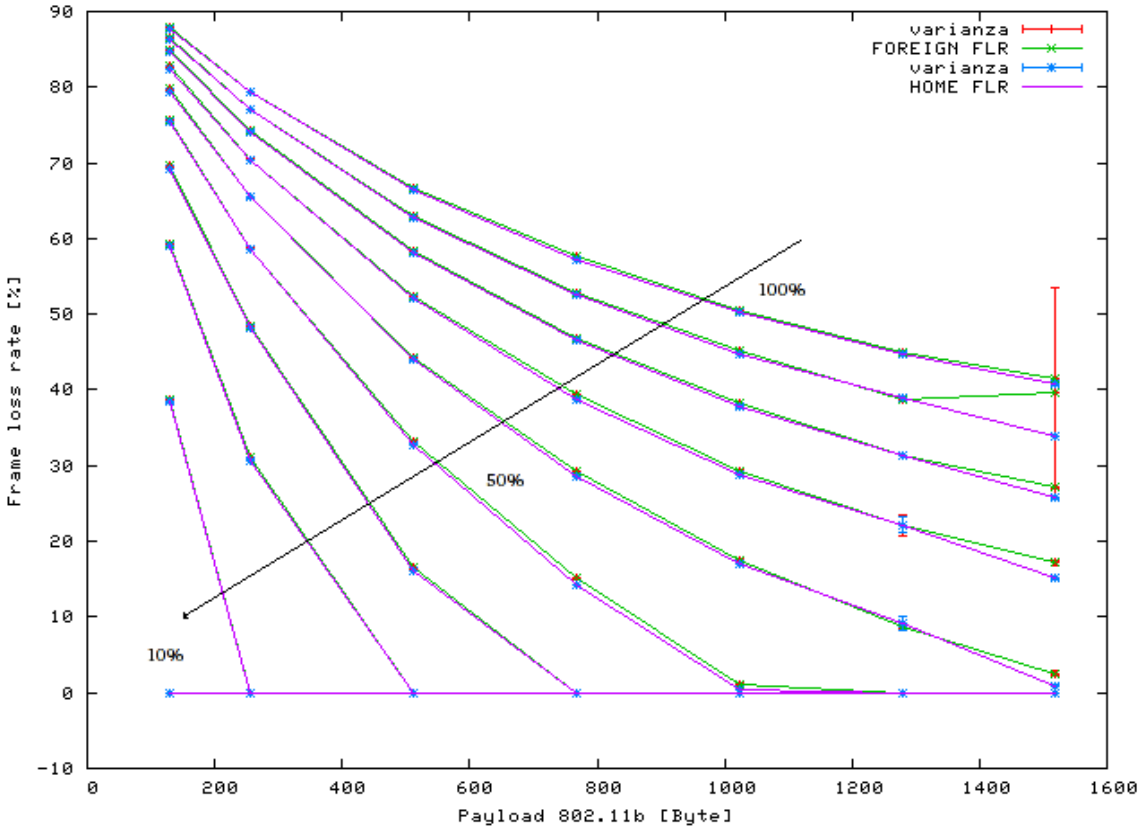


- Confronto fra i risultati in fps:

Payload IPv6	*Thr at Home*	*Routing Optimization*	*Triangular Routing*	*TMT (fps)*
54	1.190	1.192	1.062	1.270
182	1.124	1.077	1.062	1.158
438	947	919	911	984
695	825	792	794	859
950	730	722	703	764
1206	662	649	535	686
1386	612	386	333	624

6.9 Confronto dei Frame Loss rate

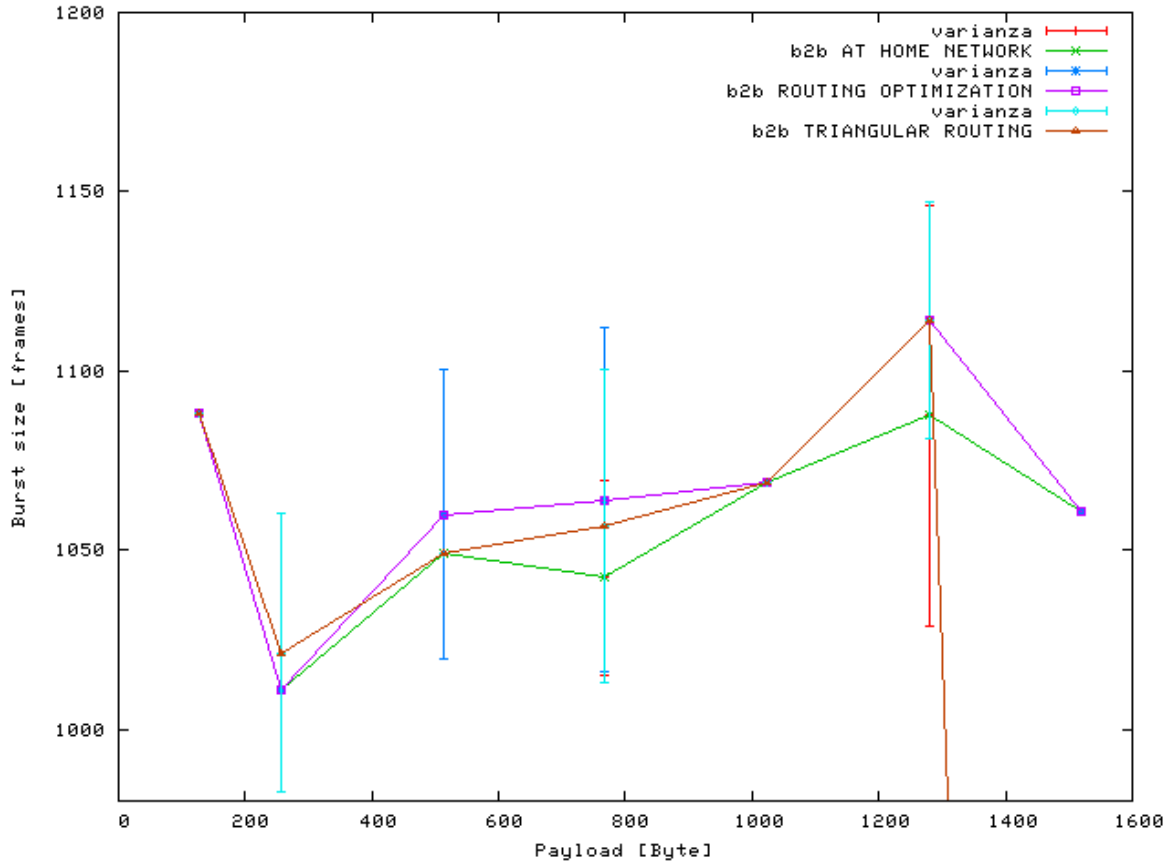
- Grafico:



Le differenze fra il FLR nelle due situazioni, MN at Home ed MN at Foreign Network, sono minime almeno fino al penultimo frame size. In corrispondenza dell'ultimo frame size, che comporta la frammentazione eseguita dai due end-point, le differenze si fanno più evidenti: il FLR-at-Home è mediamente minore rispetto al FLR-at-Foreign, del 2%.

6.10 Confronto dei Back-to-Back Frame Test

- Grafico:

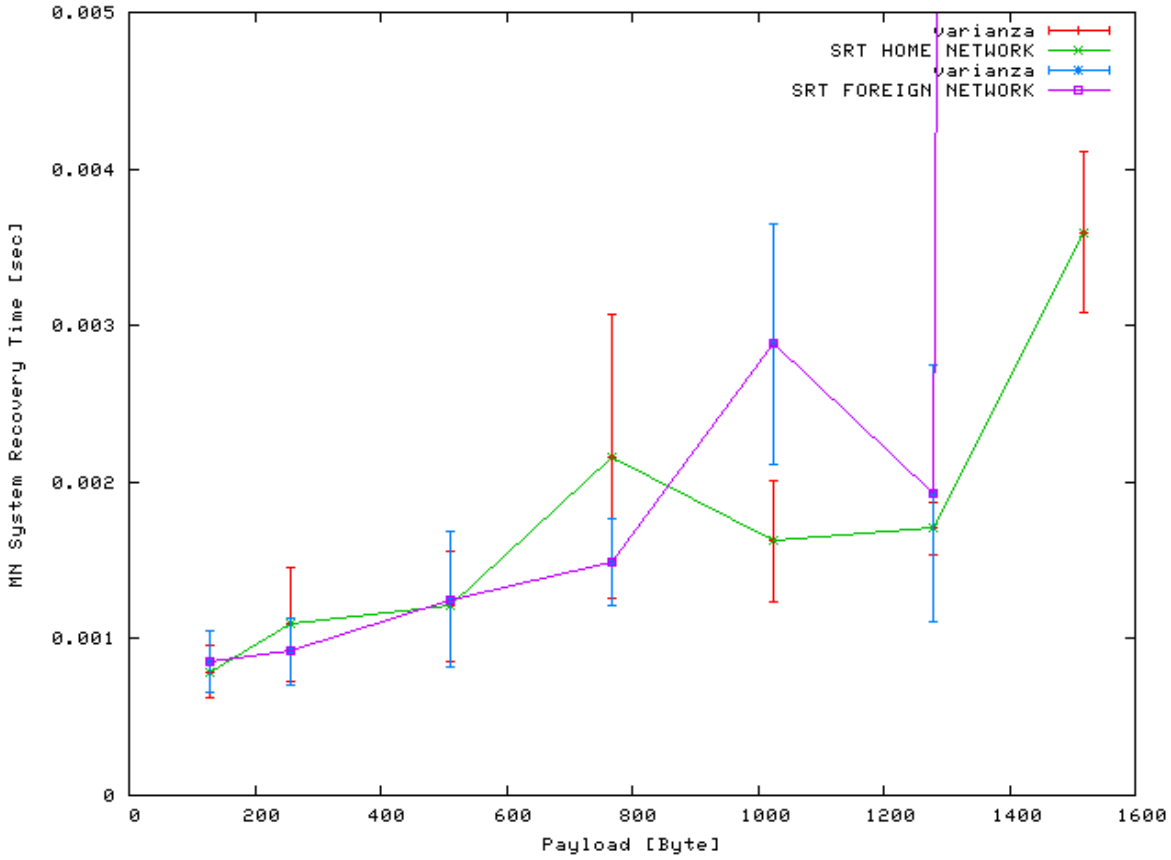


- Confronto fra i risultati:

Payload IPv6	*B2B at Home*	*B2B Routing Optimization*	*B2B Triangular Routing*
54	1.088	1.088	1.088
182	1.011	1.011	1.021
438	1.049	1.059	1.049
695	1.042	1.064	1.056
950	1.069	1.069	1.069
1206	1.087	1.114	1.114
1386	1.061	1.061	55

6.11 Confronto dei System Recovery Time Test

- Grafico:

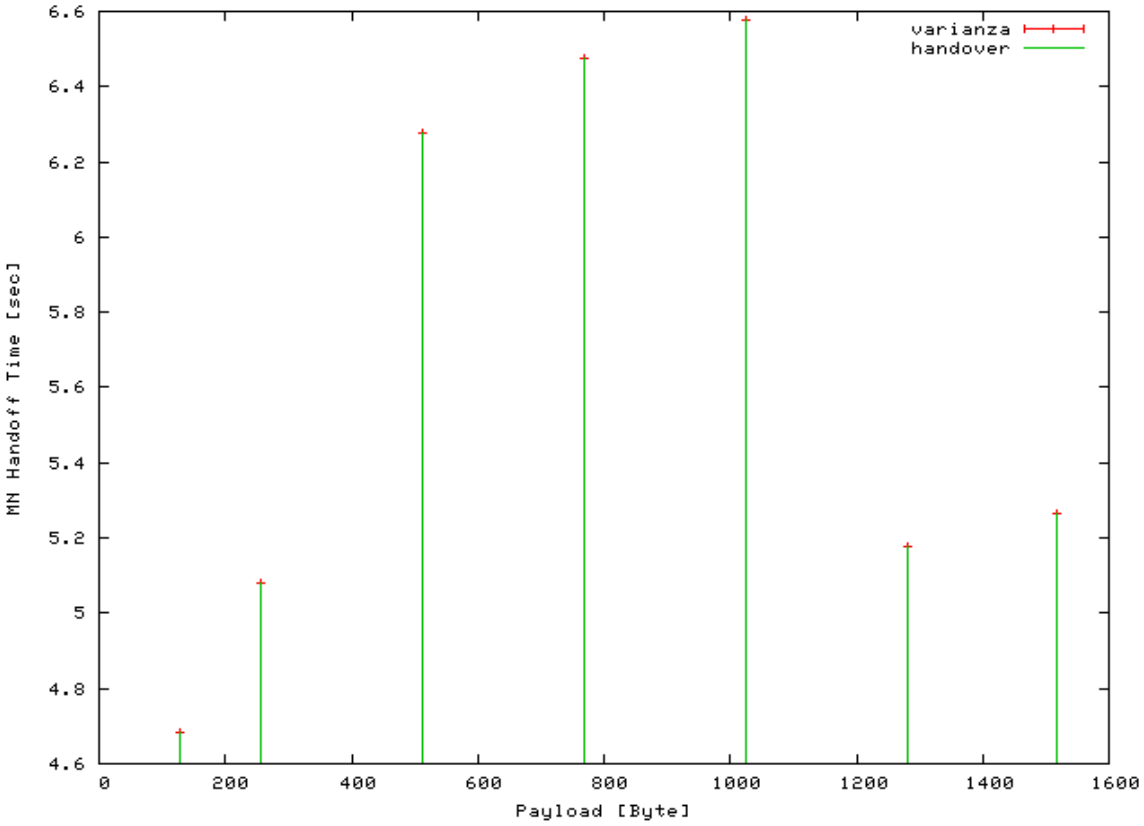


- Confronto fra i risultati:

Payload IPv6	*SRT at Home Network (msec)*	*SRT at Foreign Network (msec)*
54	0,790	0,853
182	1,095	0,920
438	1,207	1,249
695	2,165	1,489
950	1,624	2,883
1206	1,708	1,928
1386	3,593	198,993

6.12 Handoff Test

- Grafico:



- Tabella dei valori:

Payload IPv6	*Handoff Time (sec)*
54	4,683
182	5,081
438	6,278
695	6,473
950	6,675
1206	5,175
1386	5,267
Valore Medio	*5,660 sec*

6.13 Interpretazione dei risultati dei Test RFC2544

6.13.1 Throughput:

Le performance del Nodo Mobile diminuiscono in modo sensibile (fino a 11,61%) quando questo si trova in una Foreign Network. Comportamento tra l'altro ben prevedibile considerato che dato un payload IP costante si ha un aumento del framesize di 40 byte dovuti al tunneling Ipv6-within-Ipv6.

Dal confronto con il Throughput Teorico, il test at home offre i migliori risultati con una differenza media dal teorico del 4,15%. Per il test del Throughput via *Routing Optimizaion* la differenza media dal teorico risulta 6,41%, ovviamente migliore del Throughput via *Triangular Routing* che si distanzia dal teorico del 11,61%. In dettaglio gli scarti percentuali sono (l'ultimo frame size non è stato considerato in quanto introduce frammentazione):

FS	*% THR Home*	*% THR ROPT*	*% THR TRG*
54	6,30	6,14	16,38
182	2,94	6,99	8,29
438	3,76	6,61	7,42
695	3,96	7,80	7,57
950	4,45	5,50	7,98
1206	3,50	5,39	22,01
Medie	***4,15%***	***6,41%***	***11,61%***

Tabella: Gli scarti percentuali dal Throughput Teorico al variare dei SUT.

6.13.2 Back-to-Back Test

Il massimo numero di frame in un burst che il lato Server (Home Agent/Access Router) può ricevere senza perdere frame (B2B) è lievemente influenzato dalla mobilità infatti sia che il nodo mobile si trovi at Home che si trovi nella Foreign Network il suo valore si aggira intorno ai 1000 frame per burst.

Diverso è il caso in cui si utilizza il massimo frame size, quest'ultimo introducendo la frammentazione eseguita dai soli end-point e non dal router (Home Agent) interposto, produce un crollo del B2B a soli 55 frame per burst nel caso di Routing Triangolare (MN->AR->HA->CN/AR).

Payload IPv6	***B2B at Home***	***B2B Routing Optimization***	***B2B Triangular Routing***
54	1.088	1.088	1.088
182	1.011	1.011	1.021
438	1.049	1.059	1.049
695	1.042	1.064	1.056
950	1.069	1.069	1.069
1206	1.087	1.114	1.114
1386	1.061	1.061	55

Tabella 10: I valori del Back-to-Back burst size al variare dei SUT.

6.13.3 System Recovery Time

Il System Recovery Time, ovvero il tempo con cui il lato server del test (Home Agent o Correspondent Node) recupera la connessione da una condizione di overload ovviamente cresce al crescere del frame size ma differenze sostanziali fra Home e Foreign Network non sono state rilevate. Il valore medio è dell'ordine di 1,5 msec.
Nel caso di frammentazione (massimo frame size) il valore del System Recovery Time esplode fino a 198,993msec nel caso di Routing Triangolare.

Payload IPv6	*SRT at Home Network (msec)*	*SRT at Foreign Network (msec)*
54	0,790	0,853
182	1,095	0,920
438	1,207	1,249
695	2,165	1,489
950	1,624	2,883
1206	1,708	1,928
1386	3,593	198,993

Tabella 11: I valori del Back-to-Back burst size al variare dei SUT.

6.13.4 Handoff Time

L'handoff misurato mediamente consta di 5,16 sec, un valore sicuramente importante e che sicuramente non soddisfa le aspettative. Necessita pertanto un'analisi più approfondita del comportamento dell'implementazione di Mobile-IPv6 per scoprire eventuali colli di bottiglia e su quali parametri di funzionamento è possibile agire per diminuire la latenza dovuta all'handoff.
Questa ulteriore analisi è fattibile solo se si contemplano nel dettaglio, per esempio tramite i Log-file di tcpdump, le varie fasi costituenti un handoff ed i relativi tempi necessari al loro compimento.
A tal proposito ho sviluppato un'applicazione web ad-hoc a cui ho dato il nome: *MIPv6-Analyzer*. Questa applicazione implementa le misure indicate nella Sezione 4.3.
Nel prossimo capitolo presentiamo l'Applicazione MIPv6-Analyzer ed i risulati da essa ottenuti.

7 – MIPv6-Analyzer: handoff analysis from *tcpdump*

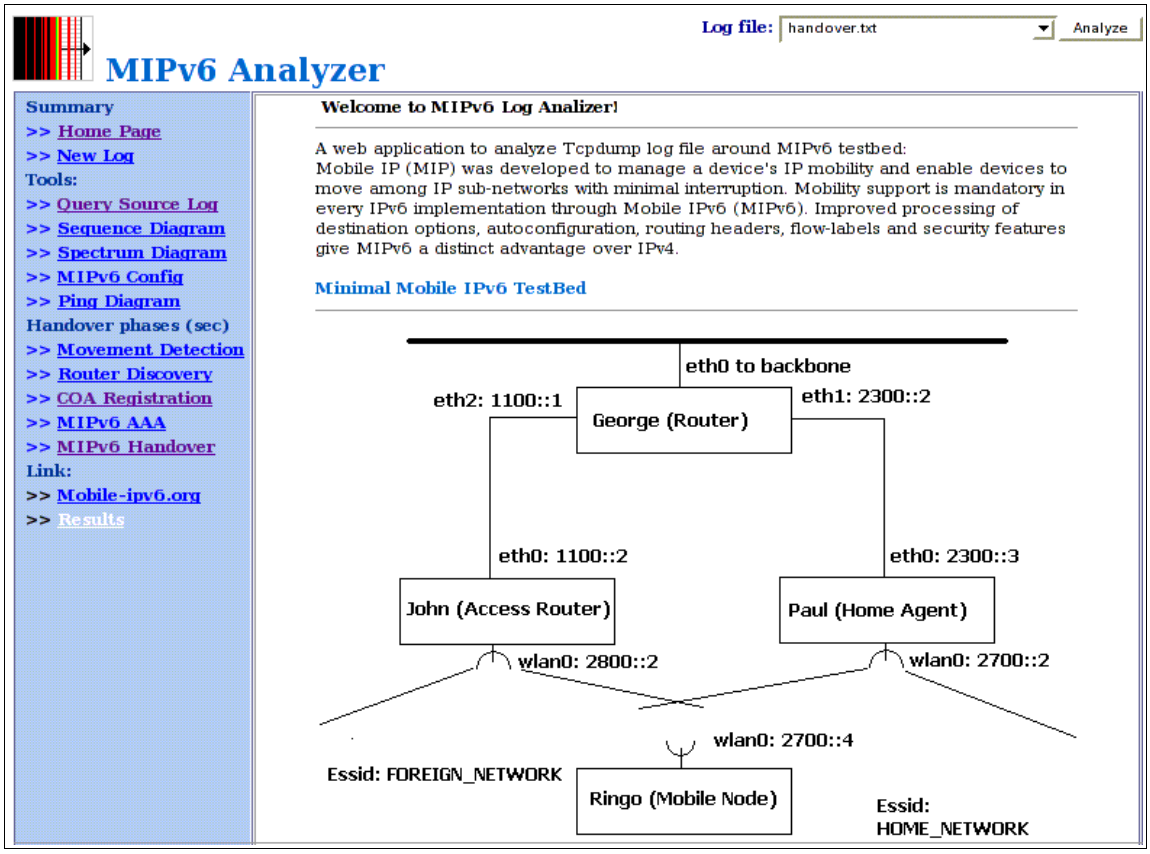


Figura 11: Mipv6-analyzer (www.cavone.com/mipv6-analyzer)

7.1 Presentazione:

Mipv6-analyzer è una applicazione web per lo studio di Mobile IPv6 per Linux sulla base dell'analisi temporale dei dati riportati nei file di log di tcpdump, in particolare ottenuti tramite il seguente filtro:

```
shell>> tcpdump -i wlan0 -vv ip6 or proto ipv6
```

L'applicazione consente o di analizzare un file di log già disponibile sul web-

server e selezionabile dal menu a tendina presente in alto a destra, o di analizzare un nuovo file di log tramite copia-incolla sull'apposito *form* (Summary >> New Log).

7.2 Funzionalità

Come si evince dal menu principale le caratteristiche offerte da Mipv6-analyzer sono:

7.2.1 Query Source Log:

Consente di interrogare l'intero file di log relativamente al tempo trascorso tra due messaggi successivi selezionati. Il file di log viene proposto in forma tabellare i cui campi sono:

- **Timestamp**: il tempo in cui cui è stato rilevato l'i-mo messaggio. Calcolato come il numero di secondi trascorsi dalle ore 00:00:00,000000 con la precisione del usec.
- **Sender**: Indirizzo IPv6 (o alias) del sender.
- **Receiver**:Indirizzo IPv6 (o alias) del receiver.
- **Message**: il messaggio così come decodificato dal tcpdump.
- **TimeDiff**: Tempo trascorso (in sec) dal messaggio precedente.
- **TimeSub**: Tempo totale (in sec) trascorso dal primo messaggio rilevato.

La tabella è ordinabile in funzione del nome del campo selezionato.

7.2.2 Sequence Diagram:

Questo funzione genera un dettagliato diagramma delle sequenze (in stile UML) sulla base del file di log attualmente in uso. Molto utile per analizzare nel dettaglio le messaggistiche di Mobile IPv6. Le figure riportate nella sezione 4.3 di questa tesi sono state generate con questo strumento. Un esempio:

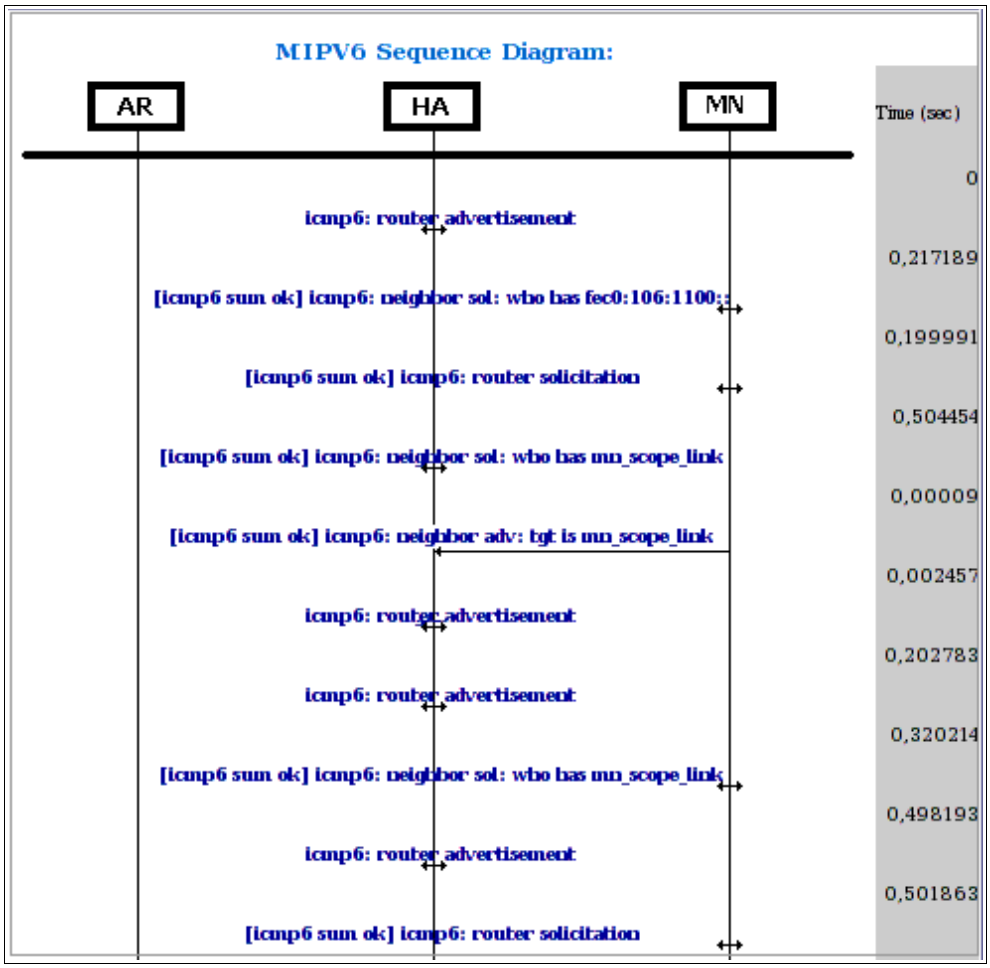


Figura 12: MIPv6-analyzer. Sequence Diagram (Returning to Home Network)

L'applicazione individua mittente e destinatario tramite un opportuno file di

configurazione nel quale si associano determinati alias agli indirizzi IPv6 presenti nel file di Log in uso. Inquesto caso:

```
################################################
# MIPv6 Analizer Configurator v 1.0
# Author: Paolo at Cavone.com - 04/05/2005
################################################
# Mobile Node Addresses
fec0:106:2700::4 MN_Scope_Site
fe80::240:5ff:feae:c364 MN_Scope_Link
fec0:106:2700:0:240:5ff:feae:c364 MN_COA_HOME
fec0:106:1100:0:240:5ff:feae:c364 MN_COA_FOREIGN
# Home Agent Addresses
fec0:106:2700::2 HA_Scope_Site
fe80::205:5dff:fe5b:f888 HA_Scope_Link
fec0:106:2300::2 HA_AR_Scope_Site
fe80::250:4ff:fed0:695f HA_AR_Scope_Link
# Access Router Addresses
fec0:106:2300::1 AR_HA_Scope_Site
fe80::201:2ff:feab:9dab AR_HA_Scope_Link
fec0:106:1100::1 AR_Scope_Link
fe80::240:5ff:feaf:1049 AR_Scope_Link
# EOF
```

7.2.3 Spectrum Diagram:

Questo è un utile strumento per distingure differenti file di log a colpo d'occhio. Genera infatti una sorta di spettro del file di log selezionato disegnando, su una scala temporale, una barra colorata per ogni tipologia di messaggio rilevato. In particolare:

- *Router Advertisement (rosso)*
- *Router Solicitation (azzurro)*
- *Neighbor solicitation (verde)*
- *Neighbor Advertisement (giallo)*
- *Echo Request (fucsia)*
- *MIPv6 Messages (marrone)*

Il diagramma riportato è zoomabile in funzione del relativo parametro e scorrendo il mouse sul diagramma vengono evidenziati i corrispondenti messaggi nella casella di testo sottostante:

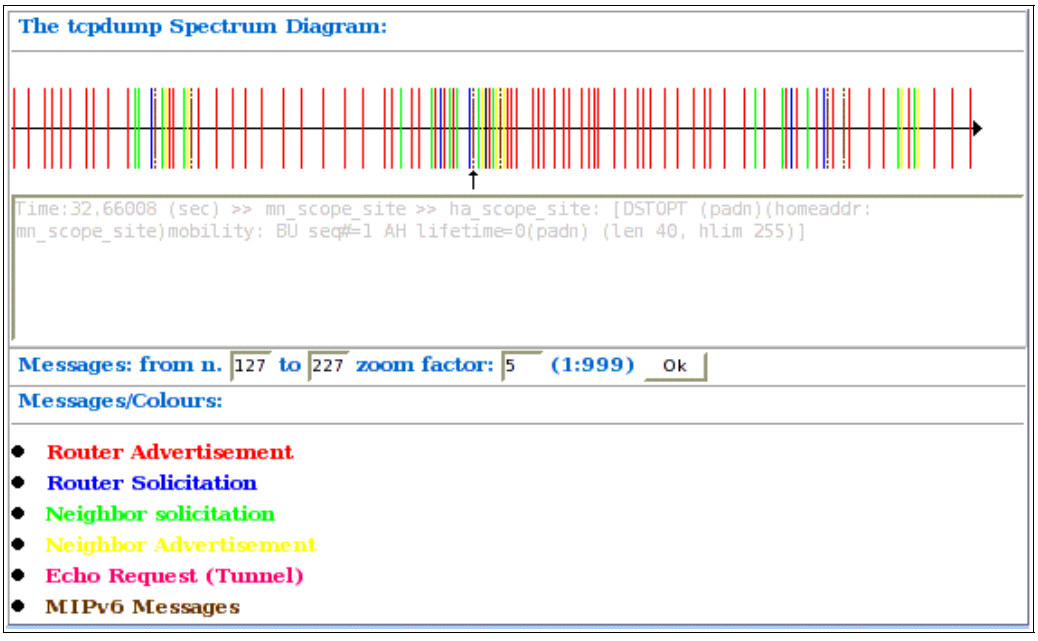


Figura 13: MIPv6-analyzer. Spectrum Diagram (Binding Deregistration)

7.2.4 Fasi dell'handoff

In questa sezione dell'applicazione vengono riportate l'analisi statistica dei tempi impiegati dalle tre fasi costituenti l'handoff:

1. Movement Detection (Router Advertisement)
2. Router Discovery (COA formation, DAD procedures)
3. Binding Registration (BU/BA)
4. Handover Latency (1+2+3)
5. Correspondent Registration (MIPv6 Authentication)

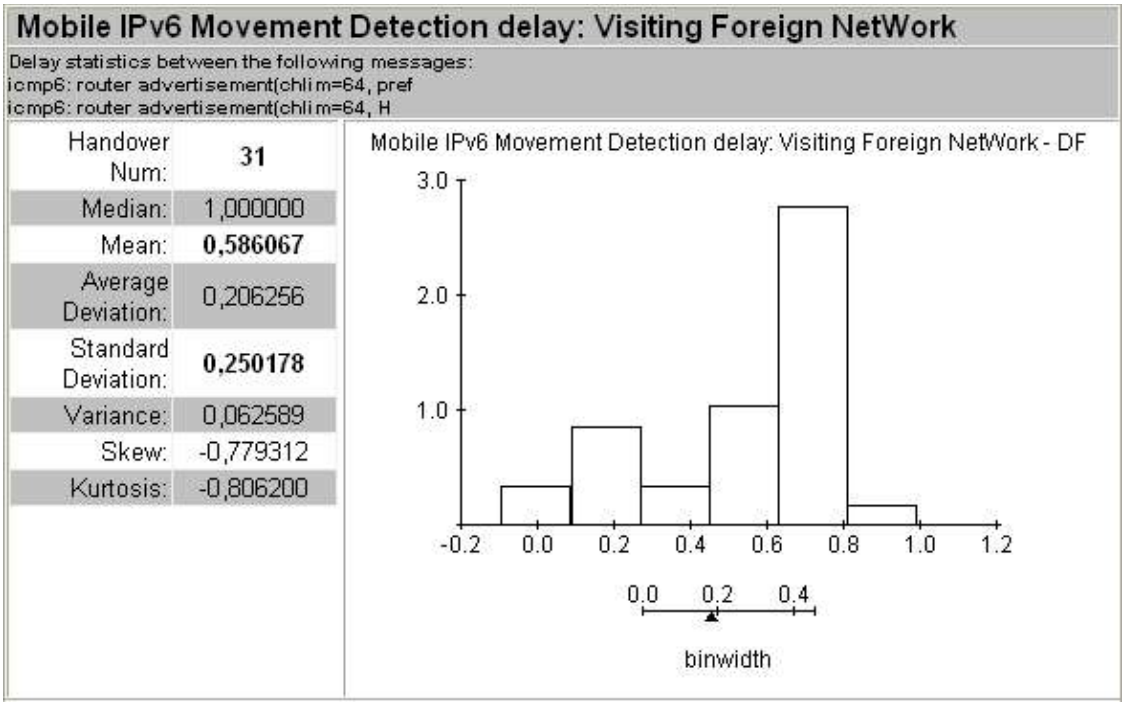
La metodologia impiegata è quella presente nei paragrafi 4.3.

Per ciascuna fase e per ciascun verso dello spostamento del nodo mobile (Visiting Foreign Network/Returning to Home Network) viene generato un report riportante:

- Oggetto e *direzione* (HN< ->FN) della misura
- Metodologia di calcolo: differenza fra i timestamp (target) dei relativi messaggi identificativi.
- Numero di prove (ovvero num. di handoff)
- Parametri caratteristici delle distribuzioni dei tempi rilevati:
  - Media e Mediana
  - Deviazione Standard e Varianza
  - Curtosi e Asimmetria (Skew)
- Distribuzione di Probabilità (a posteriori)

Di seguito i risultati:

7.3 Movement Detection Time (MDT)



L'MDT è calcolato come la differenza tra il timestamp dell'ultimo adv ricevuto dalla precedente rete ed il primo ricevuto da quella attuale. In corrispondenza dell'intervallo minimo di Advertisement [0,1:0,2]sec, testato utilizzando l'attuale versione (0.8) del radvd daemon, il valore del Movement Detection Time medio è pari a 0,58 sec con deviazione standard pari 0,25. Tale valore è prossimo a quello aspettato dato che:

$$MDT = L2\_Handover\_Time + Media\_RADVD\_Interval =$$
$$= (0{,}350 + 0{,}150) = 0{,}5sec$$

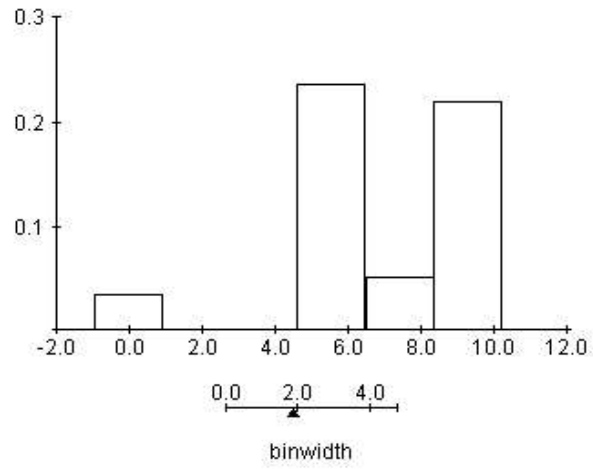
Tuttavia dalla DF si evince che il valore più frequente è 0,7 sec.

7.3 Router Discovery Time (RDT)

Mobile IPv6 Router Discovery delay: Visiting Foreign NetWork

Target Messages:
-> neighbor sol: who has fec0:106:1100:0:240:5ff:feae:c364
-> mobility: BU seq#=0 AH lifetime=10000

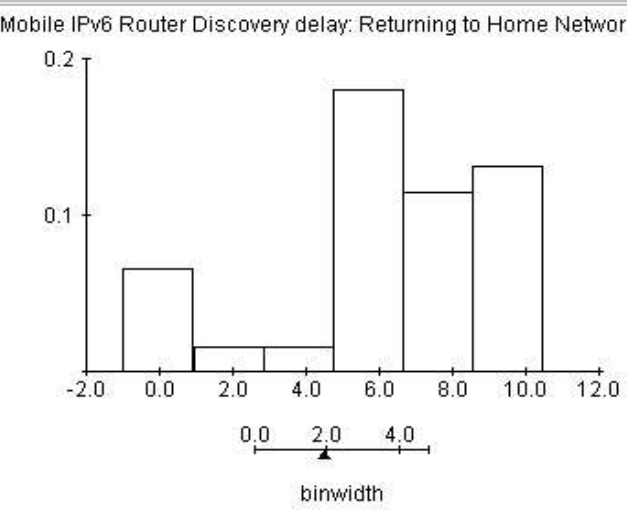
Handover Num:	**31**
Median:	6,000000
Mean:	**7,060293**
Average Deviation:	1,792528
Standard Deviation:	**2,077608**
Variance:	4,316454
Skew:	-0,889708
Kurtosis:	0,786174



Mobile IPv6 Router Discovery delay: Returning to Home Network

Target Messages:
-> neighbor sol: who has fec0:106:2700:0:240:5ff:feae:c364
-> mobility: BU seq#=1 AH lifetime=0

Handover Num:	**31**
Median:	8,000000
Mean:	**6,421899**
Average Deviation:	2,173776
Standard Deviation:	**2,728477**
Variance:	7,444586
Skew:	-0,916325
Kurtosis:	-0,222315



Questa fase dell'handover Latency, tutta gestita da IPv6, è quella più critica in quanto è comprensiva dei seguenti sotto processi:

- COA formation: creazione del Care-Of Addresses attraverso l'autoconfigurazione state-less: new_subnet_prefix+MAC address.
- DAD procedures: controllo di univocità, sull'attuale subnet, degli indirizzi IPv6 associati all'interfaccia wireless del nodo mobile: MN_scope_site_address (COA) e MN_scope_link_address.
- Neighbour Unreachability Detection: Insieme al controllo di univocità degli indirizzi, il nodo mobile in fase di transizione da una rete all'altra, cerca di ricontattare il proprio vecchio agente di mobilità tramite un Neighbour advertisement (who has *old-subnet prefix*?) seguito da *Router Solicitations*. A quest'ultima sollecitazione risponderà il l'attuale Access Router è finalmente si avvia il processo di Binding Update dato che il COA nel frattempo è stato definitivamente assegnato al nodo mobile.

Il valore medio di questa fase è risultato essere di ben 7,06 sec nel caso di movimento HN->FN e di 6,42 sec nel caso contrario (FN->HN). I valori più frequenti invece si aggirano intorno ai 5 sec.

Range (sec)	***RDT HN->FN***	***RDT FN-HN***
[0,0:1,0]	0,03	0,06
[1,0:3,0]	0,00	0,01
[3,0:5,0]	0,00	0,01
[5,0:6,5]	***0,25***	***0,18***
[6,5:8,5]	0,05	0,11
[8,5;10,5]	0,22	0,13

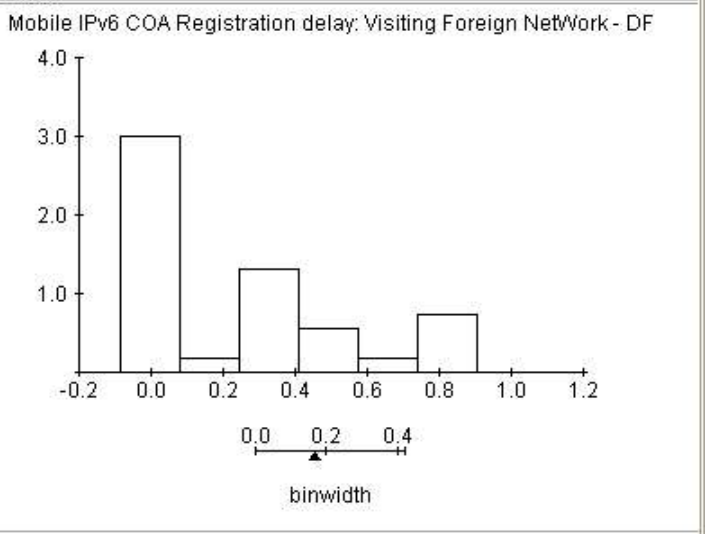
Tabella: RDT: tempi e probabilità

7.4 Binding De/Registration Time (BDT)

Mobile IPv6 COA Registration delay: Visiting Foreign NetWork

Target Messages:
-> mobility: BU seq#=0 AH lifeti me=10000
-> mobility: BA status=0 seq#=0 lifeti me=1000

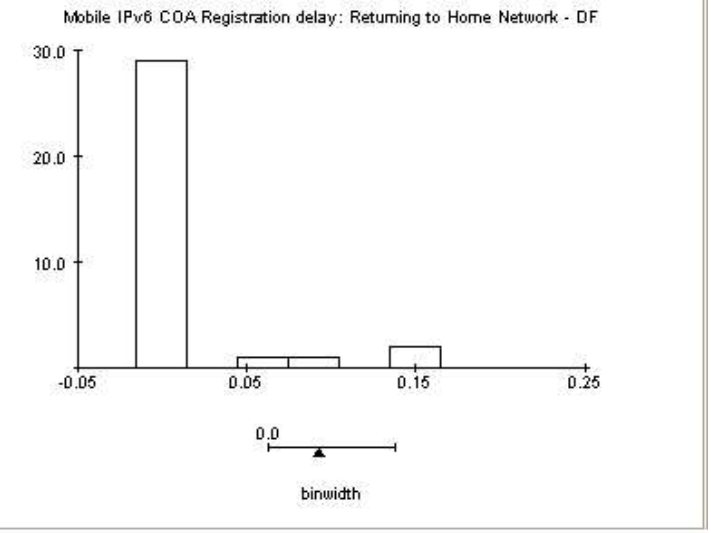
Handover Num:	**31**
Median:	0,000000
Mean:	**0,260772**
Average Deviation:	0,254246
Standard Deviation:	**0,296382**
Variance:	0,087842
Skew:	0,691186
Kurtosis:	-0,947380



Mobile IPv6 COA Registration delay: Returning to Home Network

Target Messages:
-> mobility: BU seq#=1 AH lifeti me=0
-> mobility: BA status=0 seq#=1

Handover Num:	**31**
Median:	0,000000
Mean:	**0,014142**
Average Deviation:	0,024288
Standard Deviation:	**0,039722**
Variance:	0,001578
Skew:	2,599219
Kurtosis:	5,430667



Osservazioni:

La fase di registrazione del Binding del nodo mobile (Home Addresss, Care-of Address e Lifetime) presso l'Home Agent caratterizza l'apporto di Mobile-IPv6 all'Handoff di livello 3.

Essa inizia con l'invio del messaggio di Binding Update (BU) dal nodo mobile all'Home Agent e termina con la ricezione del Binding Acknoldgment. La sua durata varia al variare del tipo di movimento effettuato: dalla Home Network verso la Foreign Network il valore medio è pari a 0,26 sec con varianza di 87msec e coincide con quello più frequente, viceversa quando il nodo mobile ritorna nella Home Network il valore medio scende a soli 14msec con varianza praticamente nulla (1msec).

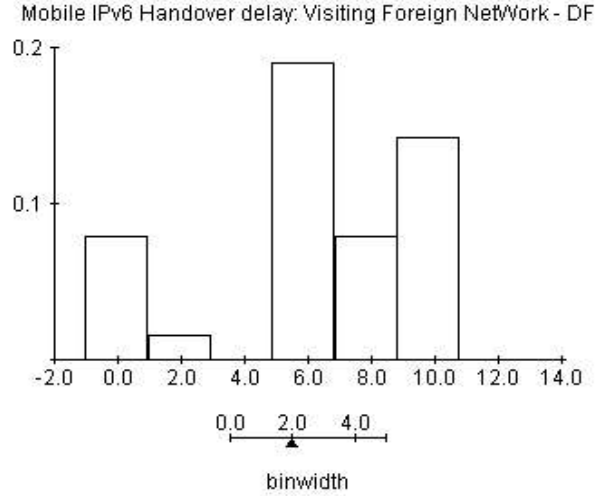
La deregistrazione del care-of address è quindi quasi istantanea a differenza della registrazione la cui durata varia da 0.2 a 0.8 sec.

7.5 Handoff Latency Time (HLT)

Mobile IPv6 Handover delay: Visiting Foreign NetWork

Target Messages:
-> neighbor sol: who has fec0:106:1100:0:240:5ff:feae:c364
-> mobility: BA status=0 seq#=0 lifetime=1000

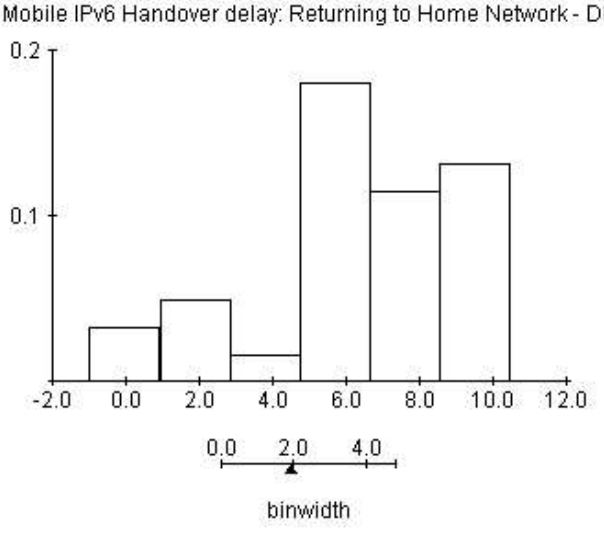
Handover Num:	**31**
Median:	6,000000
Mean:	**6,353701**
Average Deviation:	2,483906
Standard Deviation:	**3,170578**
Variance:	10,052565
Skew:	-0,820791
Kurtosis:	-0,484167



Mobile IPv6 Handover delay: Returning to Home Network

Target Messages:
-> neighbor sol: who has fec0:106:2700:0:240:5ff:feae:c364
-> mobility: BA status=0 seq#=1 lifetime=0

Handover Num:	**31**
Median:	8,000000
Mean:	**6,436041**
Average Deviation:	2,163484
Standard Deviation:	**2,704485**
Variance:	7,314238
Skew:	-0,899365
Kurtosis:	-0,249499



Osservazioni:

L'*Handover Latency Time* (HLT), comprensivo di tutte le fasi precedenti, risente in massima parte del Router Discovery Time, ovvero da tutti i processi dovuti alla *robustezza* di IPv6:

- Duplicate Address Detection
- Neighbour Unreachability

Il valore medio dell'HLT è 6,35sec con deviazione standard 3,17, il valori più frequenti ruotano intorno all'intervallo [5,0:7,0] sec.

In dettaglio la tabella tempi/probabilità (a posteriori) :

Range (sec)	***HLT HN->FN***	***HLT FN-HN***
[0,0:1,0]	0,08	0,04
[1,0:3,0]	0,02	0,05
[3,0:5,0]	0,00	0,01
[5,0:7,0]	***0,19***	***0,18***
[7,0:9,0]	0,02	0,12
[9,0;11,0]	0,15	0,13

Tabella: Handoff Latency: tempi e probabilità

I valori rilevati sono pressochè gli stessi della relativa tabella RDT.

In totale i contributi percentuali di ciascuna fase sono:

Handoff Phase	***Time (sec) HN->FN***	***Contrib.% HN->FN***	***Time (sec) FN->HN***	***Contrib.% FN->HN***
Movement Detection	0,58	9,13%	0,62	9,64%
Router Discovery	5,51	86,77%	5,80	90,20%
Binding Registration	0,26	4,09%	0,01	0,15%
Handoff Latency	**6,35**	**100,00%**	**6,43**	**100,00%**

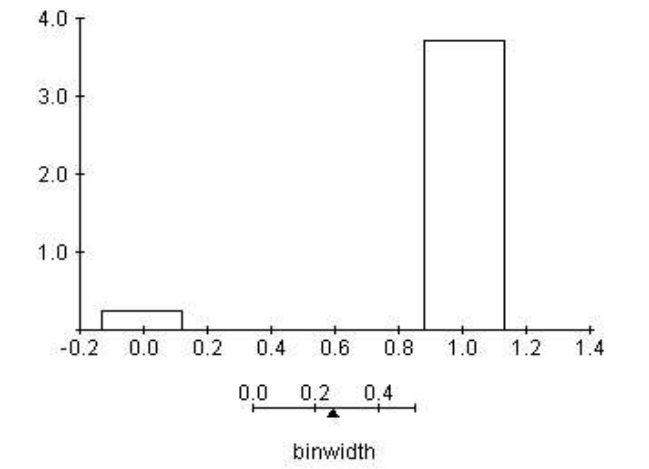
Come già accennato si evince che il collo di bottiglia dell'Handoff Latency è causato da IPv6 (oltre l'80% dell'intera latenza).

7.6 Correspondent Registration Time (CRT)

Mobile IPv6 Authentication delay: Visiting Foreign NetWork

Target Messages:
-> mobility: BA status=0 seq#=0 lifetime=1000
-> mobility: BU seq#=0 lifetime=420

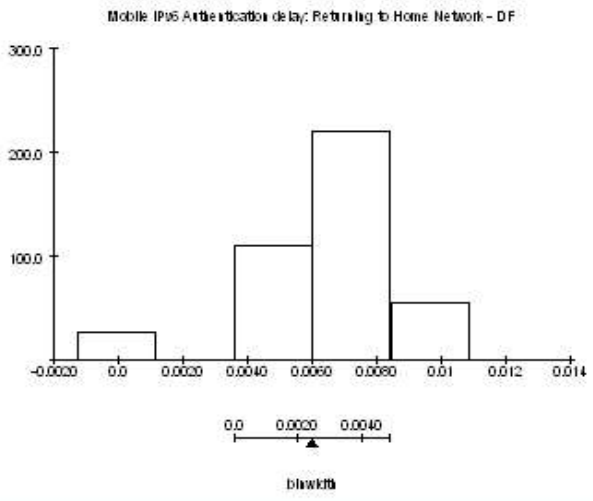
Handover Num:	**15**
Median:	1,000000
Mean:	**0,993383**
Average Deviation:	0,031290
Standard Deviation:	**0,038086**
Variance:	0,001451
Skew:	-1,137571
Kurtosis:	-0,534810



Mobile IPv6 Authentication delay: Returning to Home Network

Target Messages:
-> mobility: BA status=0 seq#=1 lifetime=0
-> mobility: BU seq#=1 lifetime=0

Handover Num:	**14**
Median:	0,000000
Mean:	**0,006852**
Average Deviation:	0,001050
Standard Deviation:	**0,001345**
Variance:	0,000002
Skew:	0,919141
Kurtosis:	-0,504550



Osservazioni:

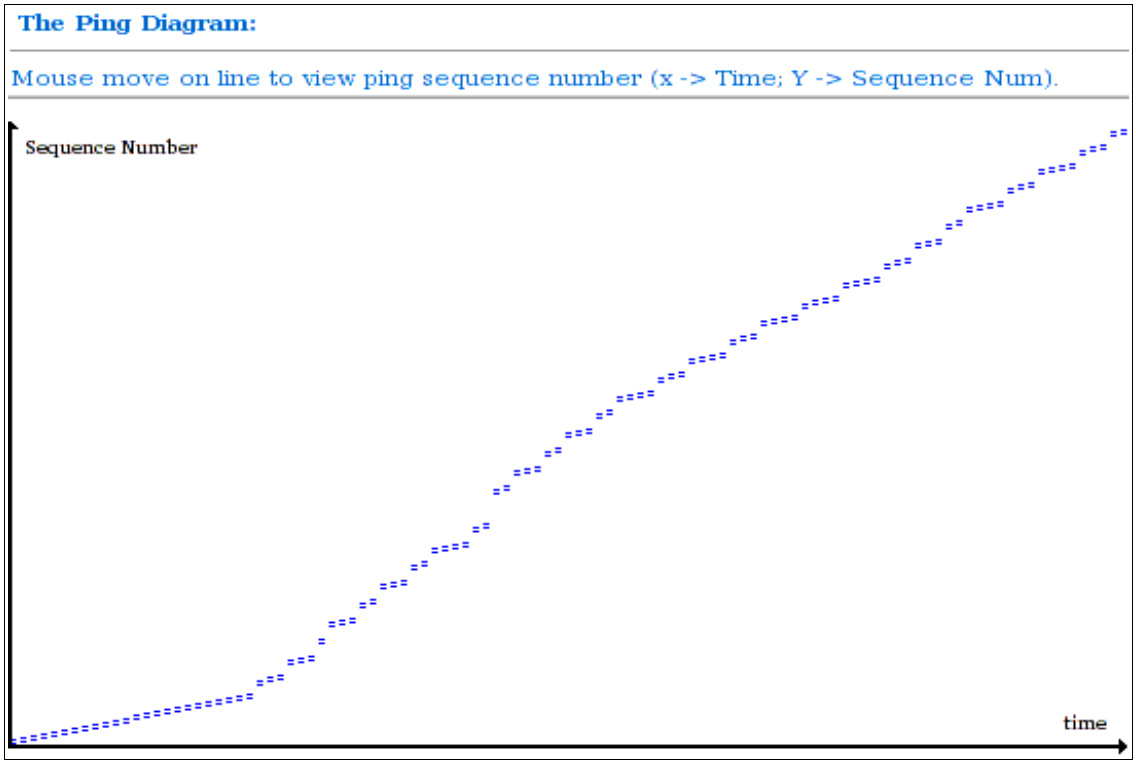
Nel caso in cui il nodo mobile effettua un handoff mentre è già in corso una connessione con un nodo corrispondente allora, se quest'ultimo è MIPv6 enabled e se esiste un persorso di routing migliore del precedente, si attiva il processo di Routing Optimization: i segmenti (TCP, UDP, ICMPv6) viaggeranno direttamente fra MN e CN senza passare (via tunneling) dall'Home Agent. Questa eventualità, per evitare eventuali attacchi di ridirezione, deve però essere autenticata ed autorizzata dall'Home Agent attraverso la procedura di Return Routability. Il tempo impiegato per completare questa fase (Correspondent Registration) post-handoff è riportato nelle figure precedenti.

Il valore medio del tempo impiegato per la registrazione è risultato essere di 0,99sec con una varianza di 38msec, viceversa la Correspondent Deregistartion è molto più veloce: 6,8msec con varianza di 1,3sec.

Nei test effettuati la comunicazione MN<->CN è stata realizzata con un semplice Ping6 (messaggi ICMPv6 echo request/replay). Analizzando i relativi numeri di sequenza è stato possibile riscontrare il valore dell'handoff latency: effettuando un ping ogni sec durante l'handoff vengono persi mediamente 6,5 numeri di sequenza.

Il seguente grafico evidenzia i salti nei numeri di sequenza dovuti all'Handoff:

7.7 Ping Diagram con Handoff



Il grafico evidenzia i salti nei numeri di sequenza dovuti ad handoff e, sapendo che un echo request viene inviato ogni secondo, è possibile ricavare una stima dell'handoff valutando il valore del salto.

Mediamente è stato rilevato che si perdono 6,5 numeri di sequenza, da cui l'handoff risulterebbe pari a circa 6sec in accordo con quanto rilevato nei test dell'analisi macroscopica.

8 – Conclusioni

Le performance di Mobile IPv6 sono molto influenzate dalla robustezza e dalla *giovinezza* di IPv6, infatti se da un lato le procedure di rilevazione di indirizzi duplicati (DAD) aumentano la robustezza del protocollo, dall'altro gli algoritmi utilizzati, tutti sviluppati negli ultimi anni, sono ancora lontani dall'essere ottimizzati e raffinati. Inoltre, dato che gli indirizzi di rete (Care-Of Address) autogenerati sulla base dell'attuale prefisso di sottorete e del MAC Address (unico per ogni interfaccia) del nodi mobili, ci si potrebbe interrogare sulla effettiva necessità di tali processi. In effetti la DAD procedure può essere disabilitata ma ciò è utile solo nel caso di reti di test o comunque *sicure;* nelle reti di *produzione* problemi quali lo *spoofing di indirizzi MAC* certamente minacciano la sicurezza delle comunicazioni. Per questi motivi le procedure di rilevazione di indirizzi duplicati sono indicate come MUST negli RFCs inerenti IPv6.

In secondo luogo, anche se la procedura *Eager Cell Switching* consente efficenti rilevazioni del movimento del nodo mobile, le performance complessive dell'handover potrebbero essere ancora migliorate se il cambiamento della posizione si basasse non più sulla ricezione degli Agent Advertisement ma direttamente dalla variazione della cella wireless (ESSID) seguita da una nuova è più efficente fase di Router Discovery.

Anche se non esistono standard sulla segnalazione di eventi L2 (in particolare 802.11x) verso L3 (IPv6) ignorare questa possibilità sarebbe sicuramente sconveniente.

A tal proposito due importanti draft sull'argomento, proprio in questi ultimi

mesi, sono divenuti RFC e stanno animando i relativi workgroup e mailing list:

- RFC 4068 “Fast Handovers for Mobile IPv6” (Luglio 2005)
- RFC 4140 “Hierarchical Mobile IPv6 Mobility Management” (Agosto 2005)

Il loro studio però esula dagli scopi di questa tesi.

9-Bibliografia

- [1] RFC 3775 D. Johnson, C. Perkins, and J. Arkko, Mobility Support in IPv6, 2004.
- [2] RFC 3776 D. Johnson, C. Perkins, and J. Arkko, Using IPsec to Protect Mobile IPv6 Signaling between Mobile Nodes and Home Agents
- [3] RFC 2460. S. Deering and R. Hinden, Internet protocol, Version 6 (IPv6) Specification, 1998
- [4] RFC 2461 Narten, T., Nordmark, E., Simpson, W., "Neighbor Discovery for IP Version 6 (IPv6)", 1998
- [5] RFC 4262. S. Thomson and T. Narten, IPv6 Stateless Address Autoconfiguration, 1998
- [6] RFC 2526 David B. Johnson and Stephen E. Deering, Reserved IPv6 Subnet Anycast Addresses, 1999
- [7] IEEE: 802.11: Wireless LAN Medium Access Control (MAC) and Physical Layer
- (PHY) Specification Arch. Rat. Mech. Anal. (1997)
- [8] RFC 2402. S. Kent ans E. Atkinson, IP Authentication Header. 1998
- [9] RFC 2406. S. Kent ans E. Atkinson, IP Encapsulation Security Payload. 1998
- [10] RFC-2544. Benchmarking Methodology for Network Interconnect Devices.
- [11] RFC-1242. Benchmarking Terminology for Network Interconnection Devices
- [12] Koodli, R., "Fast Handovers for Mobile IPv6", RFC 4068, July

- 2005.
- [13] Arkko, J., Kempf, J., Zill, B., and P. Nikander, "SEcure
- Neighbor Discovery (SEND)", RFC 3971, March 2005.
- [14] Helsinki University of Technology: MIPL Mobile IPv6 for Linux http://www.mobile-ipv6.org (2004)
- [15] Paolo Cavone, MIPv6-Analyzer: http://www.cavone.com/mipv6-analyzer (2005)
- [16] Paolo Cavone, Mobydik.tk: a TCL/TK wrapper around MIPv6 and wireless tools, http://www.mobydik.it (2005)