

Desarrollo y evaluación del protocolo “Fast Handovers for Mobile IPv6” en un entorno virtual

Development and Evaluation of the “Fast Handovers for Mobile IPv6” Protocol in a Virtual Environment

◆ A. Cabellos-Aparicio, L. Calafell, J. Núñez y J. Domingo-Pascual

Resumen

Mobile IPv6 es la versión para IPv6 del protocolo de movilidad en Internet. El tiempo empleado en cambiar de un enrutador de acceso a otro (handover) con este protocolo es de aproximadamente dos segundos. Fast Handover for Mobile IPv6 reduce este tiempo a unos milisegundos. Desarrollar dicho protocolo dentro del Kernel de Linux es una tarea difícil, es por ello que se optó por hacerlo en un entorno virtual. User-Mode-Linux permite ejecutar un sistema Linux como si de un programa se tratara encima de otro sistema Linux. Este artículo presenta un análisis de dicho desarrollo así como las posibilidades que ofrece el entorno virtual al evaluar la implementación.

Palabras clave: Movilidad, User-Mode-Linux, UML, Mobile IP, Fast Handovers, Kernel, Entorno Virtual.

Summary

Mobile IPv6 is the IPv6 version for the Internet mobility protocol. The time spent by such protocol to change from one access router to another (handover) is around 2 seconds. Fast Handover for Mobile IPv6 reduces this time to several milliseconds. Developing such protocol into the Linux Kernel is a hard task, that's why it was chosen to develop it in a virtual environment. User-Mode-Linux allows running an entire Linux system as a simple program into another Linux box. This paper presents an analysis of the development of the protocol using a virtual environment as well as the type of evaluation that such environment provides.

Keywords: Mobility, User-Mode-Linux, UML, Mobile IP, Fast Handovers, Kernel, Virtual Environment.

1.- Introducción

Las tecnologías inalámbricas se han desarrollado considerablemente en los últimos años. IEEE 802.11 [1] es probablemente la más utilizada proporcionando una conexión de hasta 54Mbps. La IETF diseñó Mobile IP que, juntamente con IEEE 802.11 proporciona movilidad a Internet, es decir, un usuario puede moverse y cambiar su punto de acceso a la red sin interrumpir sus conexiones de red. La parte más crítica de esta tecnología (IEEE 802.11 y Mobile IP) es el handover. El handover es el proceso por el que un dispositivo móvil cambia de un punto de acceso a otro. Durante este proceso algunos paquetes se pueden perder o retardar. El tiempo de handover para el protocolo Mobile IPv6 [2] es de aproximadamente 2 segundos [3], esto provoca que no sea viable, dicho protocolo, para tráfico con requerimientos de retardo o pérdidas, como VoIP. Para acelerar el proceso la IETF ha diseñado Fast Handovers for Mobile IPv6 (FMIPv6) [4] cuyos principales objetivos son reducir el tiempo de handover y reducir las pérdidas de paquetes a cero (retardando los paquetes en lugar de perdiéndolos).

El protocolo FMIPv6 es un *draft* de la IETF. Se está trabajando en su estandarización y hasta la fecha no existe ninguna implementación pública del protocolo. FMIPv6 es una extensión de Mobile IPv6 y, en caso de handover, FMIPv6 consigue sus objetivos permitiendo que el dispositivo móvil envíe paquetes de datos tan pronto como se detecte el nuevo enlace IEEE 802.11, asimismo, durante el handover, mientras el dispositivo móvil está desconectado, los enrutadores de acceso guardarán los paquetes destinados al mismo en un buffer. Tan pronto como se detecte la presencia del dispositivo móvil se le enviarán todos los paquetes guardados.



Mobile IPv6 es la versión para IPv6 del protocolo de movilidad en Internet



El handover es el proceso por el que un dispositivo móvil cambia de un punto de acceso a otro. Durante este proceso algunos paquetes se pueden perder o retardar



2.- Objetivos

Dentro del proyecto SAM [5] se decidió implementar el protocolo FMIPv6 en el Kernel de Linux usando la implementación de Mobile IPv6 MIPL [6]. Desarrollar un nuevo protocolo en el Kernel de Linux es una tarea costosa, es difícil depurar el código y a la hora de realizar pruebas con el nuevo un fallo implica a menudo reiniciar todo el sistema. Particularmente, FMIPv6 debe correr en varias entidades diferentes, es decir, su implementación estará dividida en varias máquinas diferentes: en los dispositivos móviles y en los puntos de acceso. De esta manera, implementarlo implica tener que hacerlo en diversas máquinas diferentes, complicando el proceso.

Con el objetivo de evaluar y medir diferentes protocolos de movilidad (entre ellos FMIPv6), en el proyecto SAM se desarrolló un escenario de pruebas basado en máquinas Linux, enrutadores Cisco y tarjetas IEEE 802.11 funcionando en modo infraestructura (figura 1). Por las razones antes expuestas, desarrollar el nuevo protocolo en el Kernel de Linux en dicho escenario es inviable. Es por ello que se optó por desarrollar el protocolo en otro entorno de mayor productividad. Linux ofrece la posibilidad de construir sobre el sistema (que interactúa directamente con el Hardware) una o varias máquinas virtuales usando User Mode Linux (UML) [7].



Desarrollar un nuevo protocolo en el Kernel de Linux es una tarea costosa

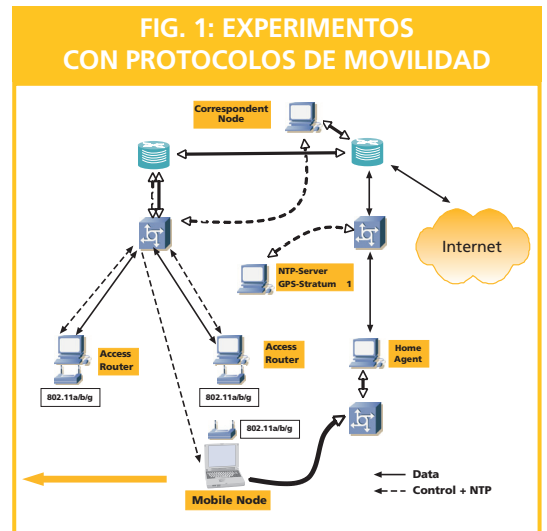


Para desarrollar el nuevo protocolo, se replicó el escenario en un entorno User Mode Linux, creando una máquina virtual por cada máquina física

3.- Entorno Virtual

UML permite “virtualizar” un sistema Linux, de manera que pueda correr como si de un programa se tratara, en una máquina física. Es más se pueden ejecutar diversos sistemas Linux independientes en una sola máquina física. Cada máquina UML puede usar un Kernel Linux o una distribución diferente, éstas, se pueden comunicar entre ellas a través de una emulación de Ethernet residente en la máquina física.

Para desarrollar el nuevo protocolo, se replicó el escenario en un entorno UML, creando una máquina virtual por cada máquina física. El escenario virtual UML es idéntico al real, se utilizan las mismas versiones de software y del Kernel de Linux. Se emuló una Ethernet en la máquina física que comunicaba las máquinas virtuales, dicha Ethernet servía de red troncal para comunicar las diferentes entidades del escenario de pruebas. Para emular la interfaz IEEE 802.11 que comunica el dispositivo móvil con los enrutadores de acceso se usó también una Ethernet, es decir, no se emuló el medio físico ni MAC de la red inalámbrica. Dicha emulación no es necesaria para una correcta evaluación del protocolo. FMIPv6 extiende las funcionalidades de MIPv6, y es independiente de la tecnología de nivel 2 utilizada. El handover en UML se emula desconectando la interfaz del enrutador de acceso y conectándola al siguiente.



Con tal de facilitar el proceso de desarrollo del nuevo protocolo, se diseñaron una serie de “scripts” que automatizaban la gestión del escenario virtual UML. A partir de la creación de una máquina UML genérica y usando una serie de “scripts”, se puede arrancarla y configurarla para que se comporte como un dispositivo móvil o cualquiera de las diferentes entidades necesarias. Es más, UML permite



instanciar más de una misma entidad, permitiendo por ejemplo, evaluar el protocolo usando múltiples dispositivos móviles y enrutadores de acceso. De este modo, el escenario virtual admite n dispositivos móviles moviéndose de una manera pseudos-aleatoria entre m enrutadores de acceso. Este tipo de pruebas son demasiado costosas de realizar en un escenario real.

Usando este sistema se pueden evaluar diferentes aspectos de la implementación. Se pueden realizar pruebas de carga del protocolo, es decir, estudiar cómo responde la implementación usando varios dispositivos móviles. También permite estudiar la cantidad de señalización que incorpora FMIPv6 respecto a MIPv6.

Una vez la implementación estuvo madura se migró al escenario real para poder realizar pruebas de prestaciones y compararla con otros protocolos como Mobile IPv6. Es importante resaltar que el mismo código que se usa en el entorno UML se puede ejecutar en el entorno real.

4.- Resultados

La figura 2 muestra una comparación entre los resultados obtenidos en el entorno UML y el real. Ambas figuras muestran el retardo sufrido por los paquetes durante un handover con FMIPv6. En el eje vertical aparece el retardo (ms) y en el eje horizontal se muestra el número de secuencia de los paquetes enviados, los números de secuencia son correlativos. En ambos casos, la gráfica describe el comportamiento del protocolo. Los paquetes fluyen hacia el dispositivo móvil con un retardo constante (inicio de la gráfica), cuando se inicia el handover (pico de retardo) los paquetes son guardados por uno de los enrutadores de acceso, justo en el momento en que el dispositivo móvil se reconecta al nuevo punto de acceso, el enrutador envía estos paquetes al dispositivo. Es por ello que los paquetes que se envían al dispositivo móvil mientras éste está desconectado (cambiando de un punto de acceso a otro) se retardan. Una vez que el dispositivo está conectado al nuevo punto de acceso, los paquetes fluyen con un retardo constante (final de la gráfica).

Una vez la implementación estuvo madura se migró al escenario real para poder realizar pruebas de prestaciones y compararla con otros protocolos como Mobile IPv6

FIGURA 2A: RETARDO DE LOS PAQUETES USANDO EL ESCENARIO REAL

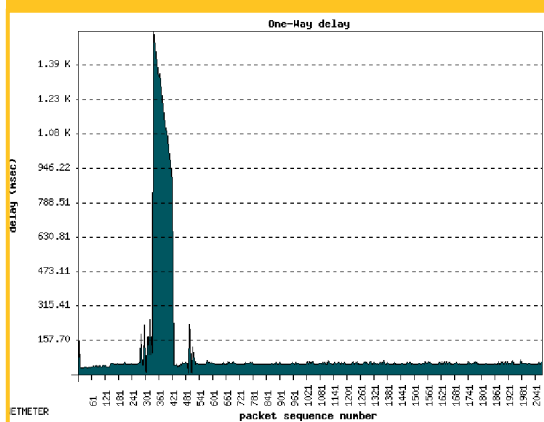
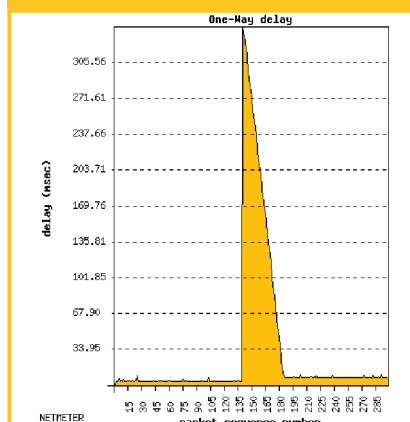


FIGURA 2B: RETARDO DE LOS PAQUETES USANDO UML



El retardo que sufren los paquetes que son transmitidos de una máquina virtual a otra, emulando una Ethernet, no tiene por qué corresponderse con la realidad

Tal y como demuestra la Figura 2, UML permite depurar el protocolo y estudiar su comportamiento, así como realizar las evaluaciones antes comentadas. Sin embargo no es útil para realizar pruebas de prestaciones. El retardo que sufren los paquetes que son transmitidos de una máquina virtual a otra, emulando una Ethernet, no tiene por qué corresponderse con la realidad. Las pruebas de prestaciones deben realizarse obligatoriamente en un escenario real.



◆
Para el desarrollo del protocolo, se creó un escenario típico de movilidad, que de hecho replicaba uno ya existente real

5.- Sumario

Desarrollar un nuevo protocolo en el Kernel de Linux es una tarea compleja, más cuando la implementación del protocolo está repartida entre diferentes máquinas. Es por ello que es necesario buscar un entorno que favorezca la productividad. Para desarrollar Fast Handovers for Mobile IPv6, una extensión del protocolo Mobile IPv6 que reduce el tiempo de handover y el número de paquetes perdidos, se ha optado por hacerlo en un entorno virtual. User-Mode-Linux permite ejecutar diferentes máquinas independientes, cada una pudiendo tener un Kernel y una distribución diferente. Así pues, para el desarrollo del protocolo, se creó un escenario típico de movilidad, que de hecho replicaba uno ya existente real. Sólo cuando el código estuvo suficientemente maduro se migró al escenario real.

El escenario virtual permite desarrollar el código con facilidad, depurarlo, evaluar su comportamiento y realizar pruebas de carga o cuantificación de la señalización, demasiado costosas para hacerse en un entorno real. Sin embargo, dicho escenario no permite evaluar las prestaciones de la implementación, no tienen por qué existir relaciones entre las prestaciones ofrecidas por UML y las de un sistema real.

Referencias

- [1] IEEE 802.11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) (1997)
- [2] D. Johnson, C. Perkins and J. Arkko: IP Mobility Support for IPv6 RFC 3775 (2004)
- [3] A. Cabellos-Aparicio, R. Serral-Gracià, L. Jakab, J. Domingo-Pascual, "Measurement Based Analysis of the Handover in a WLAN MIPv6 Scenario" Presentada en: *Passive and Active Measurements 2005, Boston (USA), Springer, LNCS 3431, pp 203-214 (ISBN 3-540-25520-6)*
- [4] Rajeev Koodli, "Fast Handovers for Mobile IPv6", draft-ietf-mipshop-fast-mipv6-03.txt (October 2004)
- [5] SAM-Advanced Mobile Services – Research Project MCyT (online) Consultado en: <http://sam.ccaba.upc.edu> (8/9/05)
- [6] Helsinki University of Technology: MIPL Mobile IPv6 for Linux (online) Consultado en: <http://www.mobile-ipv6.org/> (2004) (8/9/05)
- [7] UML – User-Mode-Linux (online) Consultado en: <http://user-mode-linux.sourceforge.net> (8/9/05)

Albert Cabellos-Aparicio,
(acabello@ac.upc.edu)
Lluís Calafell Liebanas,
(calafell@ac.upc.edu)
Jose Núñez-Martínez,
(jnunez@ac.upc.edu)
Jordi Domingo-Pascual
(jordid@ac.upc.edu)

Departament d'Arquitectura de Computadors
UPC