Resolución de alias para el cálculo de topologías

S. García, E. Magaña, M. Izal y D. Morató
Universidad Pública de Navarra
Departamento de Automática y Computación
Campus Arrosadia s/n, 31006 Pamplona
E-mail: {santiago.garcia, eduardo.magana, mikel.izal, daniel.morato}@unavarra.es

Abstract *The network topology is a fundamental parameter for managers and researchers. The traditional methodology for discovering the topology of a network is based on the tool* traceroute*, used from several vantage points in different subnetworks. The result is a set of sink trees where the nodes are the discovered IP addresses from the routers. However, few tools have faced the problem of identifying the nodes in different sink trees as interfaces in the same router. This paper shows a new methodology for this problem of* alias resolution*. It has been used in the european research network using the ETOMIC platform. It shows that the traditional methodologies are not effective in today's networking scenario but can be easily improved at least in a factor of 3 in the number of successes.*

1. Introducción

El modelado de la topología y arquitectura de redes IP como Internet son temas de estudio de importancia desde hace más de una década. Se trata de una red con centenares de miles de nodos interconectados sin un gestor central en la que por lo tanto es inviable un control único de su estructura y topología. Sin embargo, dicha topología representa una información imprescindible para cualquier administrador o gestor de red. Igualmente, la investigación centrada en numerosos temas sobre análisis de prestaciones, de retardos, congestión, encaminamiento, etc. requiere conocer la topología de red o al menos las características estructurales de las redes hoy en día (distribución del número de enlaces de los nodos, organización en clusters, etc)

En el caso de un administrador de red, se puede presuponer el acceso a dicha información, bien a través de documentación sobre la misma o mediante herramientas de gestión. Una vez fuera de su entorno de gestión (en el mejor caso limitado a su Sistema Autónomo - AS) dicha información no va a encontrarse disponible. Sin embargo, sería información muy útil a la hora por ejemplo de poner en práctica técnicas de ingeniería de tráfico, selección de rutas de salida del AS según las redes destino, decisión de rutas internas según el retardo que añada el tránsito por otros sistemas autónomos, verificación de parámetros de SLA (*Service Level Agreement*), selección de ubicación para servidores, detección de puntos críticos de fallo, etc.

En el ámbito de la investigación y desarrollo, el conocimiento de la topología de una red, generalmente Internet, o de una sección de ella, resulta fundamental para el cálculo y predicción de retardos entre nodos extremo [1], la localización geográfica de nodos [2], el diseño y prueba de protocolos de encaminamiento e ingeniería de tráfico [3], la evaluación de prestaciones de protocolos P2P [4], los mecanismos de recuperación de caminos ante fallos [5], la evaluación de algoritmos de construcción de árboles multicast [6] y en general, cualquier trabajo que requiera simulación sobre un escenario de red lo más similar posible a la realidad requerirá ejemplos reales o técnicas de generación sintética de topologías.

En este artículo se presentan mecanismos para el descubrimiento de la topología de una red mediante medidas activas desde diferentes nodos externos a la misma. En concreto el trabajo se centra en la problemática de la identificación de routers IP dadas las direcciones de sus diferentes interfaces de red. Lo que ha venido a llamarse como "resolución de alias" [7]. Se describen mejoras a las técnicas empleadas hasta el momento en la literatura que permiten la extracción de la topología con menor error en la identificación de nodos con múltiples interfaces.

El artículo se organiza comenzando en la sección 2 con la presentación de las herramientas ya existentes para el descubrimiento de la topología de una red IP así como las novedades propuestas en este trabajo. En la sección 3 se evalúan estos métodos ante un escenario de red controlado. A continuación en la sección 4 se introduce el escenario de medida real basado en la plataforma paneuropea ETOMIC creada dentro del Proyecto Integrado del VI Programa Marco "EVERGROW", para en el apartado 5 mostrar los resultados obtenidos para este escenario real. Finalmente, la sección 6 resume las conclusiones que se extraen de este artículo.

2. Metodología

2.1. Estado del arte

En cuestión de descubrimiento y descripción de la topología de una red la literatura aborda al menos cuatro niveles de detalle sobre la misma, en lo que se vendrá a llamar en este artículo:

Topología física: El interés se centra en la topología completa, incluyendo todos los equipos de intercone-

xión de redes así como los equipos de nivel de enlace (LAN o WAN) en cada una de las redes entre routers (conmutadores ethernet, ADMs SDH, conmutadores ATM, etc.). Las técnicas de descubrimiento de topologías con este detalle generalmente requieren el empleo de protocolos de gestión tipo SNMP [8].

Topología de red: El objetivo es averiguar tan solo la topología a nivel de red, incluyendo routers IP, enlaces router-a-router y enlaces router-a-subred, ignorando todas las tecnologías de nivel de enlace.

Topología efectiva de encaminamiento: Conocer la topología de interconexión de routers no implica conocer los caminos que emplearán los paquetes. Muchas de las técnicas de descubrimiento de topologías mediante mediciones activas se basan en realidad en descubrir los árboles de encaminamiento a los destinos. Tienen mayor utilidad a la hora de estudiar los caminos que seguirá el tráfico en la red pero el inconveniente de no descubrir los enlaces que en el momento del sondeo no están siendo empleados (por ejemplo enlaces de backup). Tampoco hay necesidad de relacionar los árboles calculados a diferentes destinos para reconocer los nodos que representan a la misma máquina.

Topología de ASes: Finalmente, algunos estudios no requieren conocer la red hasta el detalle de los interfaces IP de los routers sino que les es suficiente con el grado de interconexión de sistemas autónomos. Este tipo de topología puede obtenerse a partir de las mencionadas en los apartados anteriores, procediendo a la identificación del sistema autónomo al que pertenece cada nodo mediante bases de datos tipo WHOIS o empleo de los *AS Paths* de BGP [9][10][11][12].

Los dos primeros niveles de detalle descritos (topología física y topología de red) serán especialmente interesantes para trabajos que busquen por ejemplo calcular rutas alternativas dado que proveen de la información completa de enlaces disponibles. En general la topología física va a ser prácticamente imposible de conseguir debido a que requiere descubrir equipos que probablemente no incorporen nivel de red IP o el acceso a ellos esté muy controlado, así como caminos que no estén en uso e incluso enlaces que se establezcan *on-demand* ante determinadas situaciones (generalmente caminos que se activan para mantenimiento del servicio ante un fallo). Las topologías efectivas de encaminamiendo en cambio van a ser accesibles mediante técnicas activas de sondeo siempre que se disponga de máquinas distribuidas por gran número de redes desde las que iniciar dichas medidas.

La solución trivial al problema de descubrimiento de topologías se basa en el empleo de SNMP para, a partir de la consulta en MIBs de tablas de interfaces y rutas, ir descubriendo recursivamente toda la topología de la red. Sin embargo, esta técnica se enfrenta con claros obstáculos pues no es realista contar con que los agentes SNMP estén activos en todas las máquinas, mucho menos el tener acceso a las MIBs de encaminadores pertenecientes a sistemas autónomos ajenos. Para mayor dificultad, muchos fabricantes no se ajustan al diseño estándar de la MIB sino que emplean campos propietarios [8].

Las técnicas propuestas hasta el momento y utilizables en redes no controladas por el observador no cuentan con la colaboración de la misma. Los mecanismos más comunes se basan en la implementación de Jacobson en el programa *traceroute* [13]. Éste emplea datagramas UDP con campo TTL (*Time To Live*) empezando en 1 e incrementándose en una unidad para cada paquete enviado, con el objetivo de forzar la generación de mensajes ICMP de código *"time to live exceeded in transit"*. Estos mensajes desvelan al origen del datagrama UDP una de las direcciones IP del router que descartó el paquete. Mediante el empleo de la aplicación *traceroute* desde diferentes redes y a una gran cantidad de destinos se puede crear una imagen de los árboles de encaminamiento que se están empleando. Sin embargo, varias direcciones IP diferentes descubiertas por hosts en redes independientes pueden corresponder a interfaces del mismo router. Según [14] el mensaje ICMP que espera la aplicación *traceroute* debe ser enviado por cada router empleando como IP origen la del interfaz por el que lo enviará al destino (que es el host origen del datagrama UDP). En general, *traceroute* empleado desde nodos en diferentes redes podrá enviar mensajes que sean descartados por el mismo router pero a los que éste conteste desde diferente interfaz.

Sin la capacidad de reconocer que varias de las direcciones de routers corresponden a interfaces de la misma máquina se obtiene una topología con nodos duplicados con diferente nombre, mucho más compleja que la real y en la que no se descubre la existencia de numerosas interconexiones. Es el proceso que se viene a llamar de *identificación de routers* o *resolución de alias* el que va a permitir pasar del conjunto de árboles de expansión, correspondientes al encaminamiento, a un grafo de la topología de red, mediante el proceso de reducir todos los nodos que representan al mismo router a un solo nodo con todos sus interfaces. En el caso más simple (Fig. 1), permitirá reconocer las direcciones IP descubiertas de un router al usar *traceroute* entre dos máquinas en dos redes diferentes, dado que cada ejecución desvelará las direcciones de los interfaces de un lado de los encaminadores.

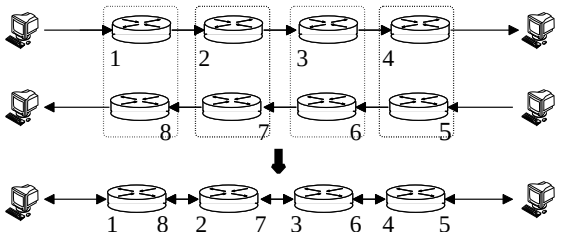

Figura 1: Resolución de alias en camino simétrico

Algunos trabajos ya han abordado esta problemática. El método propuesto en [6], implementado por CAIDA [15] en la herramienta *iffinder* [1] y empleado

[1] http://www.caida.org/tools/measurement/iffinder

en otros trabajos [12][16] propone enviar datagramas UDP desde una misma máquina a las direcciones IP que puedan pertenecer al mismo equipo. Si el puerto UDP destino no está en uso en ese router se esperaría recibir un mensaje ICMP de tipo *destination (port) unreachable*. Como se ha comentado, este tipo de mensajes parten del interfaz por el que se sigue el camino más corto al destino así que si todas las direcciones IP sondeadas pertenecen a la misma máquina entonces todas las respuestas vendrán de la misma dirección IP y se tendrá una identificación positiva. Este método se enfrenta hoy en día al filtrado ICMP realizado en muchos equipos que impide enviar estas notificaciones. También se han detectado en este trabajo equipos que pueden devolver estos mensajes ICMP desde diferentes interfaces según por qué interfaz reciban los mismos, aunque el destino no cambie, incumpliendo el requisito fundamental para que esta técnica sea efectiva.

Una alternativa planteada en [7] se basa en enviar el mismo tipo de mensajes UDP para provocar errores ICMP pero alternados a las diferentes direcciones y comparar en las respuestas el valor del campo *identificación* de la cabecera IP. Este campo es empleado en los procedimientos de fragmentación y reensamblado y permite diferenciar los datagramas IP del mismo flujo (entre los mismos hosts y protocolo). Muchas implementaciones de IP aseguran la diferencia entre los identificadores empleando un contador que se incrementa en una unidad por cada paquete que crea (independientemente del destino y protocolo y que no se ve afectado por paquetes reenviados). Eso hace que varios paquetes IP generados por la misma máquina y cercanos en el tiempo tengan valores cercanos en el campo identificación y su diferencia sólo se deba a otro tráfico intermedio generado por esa máquina. De esta forma la identificación se basaría en una cierta proximidad entre los valores de identificación de los mensajes ICMP de respuesta. En este artículo se muestra que existen implementaciones de routers que no cumplen con esta característica de incremento del valor de identificación. Además este método sigue siendo vulnerable a un filtrado de los mensajes ICMP.

Así pues, las ejecuciones de *traceroute* desde máquinas dispares descubren las direcciones IP de los interfaces de los routers pero no dan un mecanismo para reconocerlos como tales, es decir, para decidir si N direcciones IP corresponden a interfaces de un mismo router. Aquí entran en juego mecanismos para el *clustering* de dichos interfaces en lo que debería ser un router por cluster. Como se verá, las técnicas propuestas hasta el momento son muy optimistas respecto al comportamiento de los encaminadores de la red lo cual ocasiona que no sean muy efectivas.

2.2. Métodos analizados

Se describen a continuación los métodos de identificación de routers que se ponen en práctica en este trabajo, junto con las modificaciones, mejoras y nuevas propuestas:

PORT_UNREACH: basado en el envío de datagramas UDP a varias direcciones, a un puerto no utilizado, y la comparación de las direcciones IP origen de los mensajes ICMP de error de respuesta [6].

IP_IDs: Este método se basa en el presentado en [7], es decir, en la comparación de los valores del campo de identificación de los paquetes IP recibidos del router. En [7] se envían mensajes UDP a los interfaces y se espera recibir errores ICMP de los mismos. Se comparan los valores de identificación de los mensajes, cada uno de ellos proviniente de uno de los interfaces y se basa la identificación en la proximidad entre esos valores. La implementación original de este método quedará denominada como **IP_IDs (ALLY)**.

Asumiendo que el campo de identificación se incremente con cada datagrama IP que cree el router, esta técnica requiere elegir un umbral que permita reconocer a la máquina aunque entre ambos datagramas envíe otros que incrementen la secuencia, por ejemplo otros paquetes ICMP, mensajes de gestión, paquetes de protocolos de encaminamiento, etc. En la herramienta desarrollada en este trabajo lo que se propone es sondear alternativamente a las dos direcciones IP con varios paquetes y crear la serie discreta formada por los identificadores IP de cada uno de los paquetes de respuesta recibidos. En caso de que ambas direcciones pertenezcan al mismo equipo y éste emplee una estrategia de incremento secuencial del identificador se encontrará una secuencia creciente.

Dado que estos mensajes se encuentran filtrados en un gran número de routers se propone ampliar el abanico de posibilidades de recibir datagramas IP creados por el router. En este trabajo, además de mensajes UDP se envían también a cada interfaz mensajes ICMP de tipo *timestamp reply* y tipo *echo request*. Se provoca además el envío de un segmento TCP enviando a cada interfaz otro segmento TCP con el flag de SYN activo a un puerto que no esté empleando ningún servidor del router. Estas cuatro alternativas se denominarán **IP_IDs (UDP)**, **IP_IDs (TIME)**, **IP_IDs (ECHO)** e **IP_IDs (TCP)** respectivamente y se hará cada prueba para cada pareja de direcciones IP que se considere que puedan pertenecer al mismo equipo.

Se han detectado implementaciones de IP en routers que incumplen esta regla general para el valor del campo de identificación. Algunos equipos localizados en Internet devuelven valores de identificación pseudo-aleatorios, que no siguen una secuencia. Otros, al responder a un mensaje ICMP *echo request*, copian el campo de número de secuencia del mismo para el valor de identificación. Otros emplean siempre el mismo valor de identificación para todas las respuestas. Todas estas excepciones limitarán la aplicabilidad de esta técnica.

TSTAMP: Se obtendrá una nueva caracterización de las máquinas que poseen cada interfaz de red empleando varias funcionalidades de TCP/IP para la inclusión de marcas de tiempo (*timestamps*) en los pa-

quetes. Por un lado, en los mensajes TCP con el flag de RESET de la prueba **IP_IDs (TCP)**, con algunas implementaciones de TCP/IP se puede obtener la opción TCP *Timestamps*. El valor de dicha opción indica una marca de tiempo en el emisor que generalmente es el tiempo desde el arranque de la máquina, con una resolución al menos de segundos y siempre creciente [17]. Es posible reconocer mediante estas marcas de tiempo si varios interfaces corresponden a la misma máquina. Para ello se lleva a cabo de nuevo un sondeo alternativo a dos direcciones de interfaces que puedan corresponder a la misma máquina y se comprueba si la secuencia resultante es creciente. Este método se denominará **TSTAMP (TCP)**. De forma similar, los mensajes ICMP recibidos en la prueba **IP_IDs (TIME)** contienen una marca de tiempo del instante en que el router envió el mensaje. En caso de que dos interfaces contesten con esa opción se puede crear una nueva secuencia que permita decidir si pertenecen a la misma máquina.

3. Descripción y validación de resultados

El empleo de métodos de reconocimiento y análisis de topología en Internet se enfrenta habitualmente a la imposibilidad de verificar si los resultados obtenidos son correctos, es decir, no se dispone de acceso a los equipos para confirmar que la identificación de alias que se ha obtenido es válida. Por ello, para este trabajo se ha realizado un primer paso de validación de la metodología planteada empleando un entorno de red controlado dentro del Laboratorio de Telemática de la Universidad Pública de Navarra[2].

En Fig. 2 se muestra la topología de red configurada en el laboratorio. Los equipos R1 a R7 son routers Cisco mientras que R8 es un PC con sistema operativo Linux. Existen interfaces Serie, Ethernet, POS (*Packet Over SONET*) sobre STM-1, DOCSIS e interfaces ATM. No se muestra la topología de nivel de enlace pues no es relevante para este trabajo.

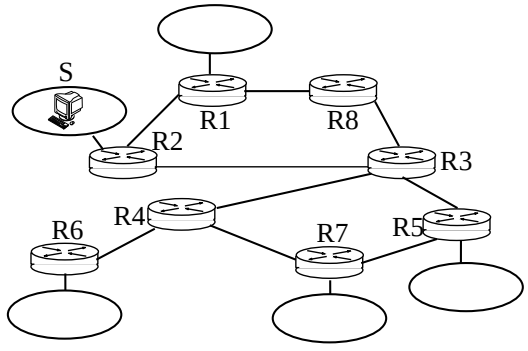

Figura 2: Topología de red controlada

La máquina S representa al ordenador-sonda desde el que se hacen las pruebas de resolución de alias. Normalmente las direcciones IP de los diferentes interfaces de los routers se descubrirían al emplear *traceroute* en ambos sentido de la comunicación entre cada pareja de sondas instaladas en las diferentes redes. Sin embargo, el objetivo principal en este trabajo es comprobar el correcto descubrimiento de alias con diferentes metodologías, proceso para el cual no es relevante la fase de descubrimiento de direcciones. Así pues se parte del conocimiento exógeno de las direcciones IP de todos los interfaces de los routers.

Se procede a continuación a intentar reconocer cada pareja de esas direcciones como el mismo router. Se comprueba para cada dirección IP descubierta si se puede emparejar con otra. Para ello, en la red controlada se hace cada prueba para cada pareja posible de direcciones IP. El tráfico introducido en la red por todas las pruebas para cada pareja de direcciones IP es inferior a 50KBytes pero crece con el cuadrado del número de nodos. En esta fase de la investigación se ha comenzado por la evaluación de las metodologías propuestas independientemente del coste. Una vez confirmado el correcto funcionamiento de las mismas se procederá en un trabajo posterior a la optimización de sus parámetros, entre ellos el ancho de banda consumido. La Tabla 1 muestra los resultados obtenidos con cada método. El tipo de información que se puede extraer de la ejecución de cada prueba para cada pareja de direcciones IP es diferente con cada método. Se describen a continuación los posibles resultados:

Emparejamiento Cierto: Un método podrá indicar con seguridad que una pareja de direcciones IP pertenece a un mismo equipo. De ser así se considerará cada una de esas direcciones como que ha dado lugar a un emparejamiento cierto. La columna *Cierto* de la Tabla 1 muestra el porcentaje de direcciones IP que han dado algún emparejamiento cierto. Las técnicas **TSTAMP** no pueden asegurar emparejamientos ciertos dado que se basan en el instante de tiempo actual marcado por el reloj de la máquina o del tiempo transcurrido, generalmente desde el arranque de la misma. Pueden existir máquinas diferentes cuyos relojes estén sincronizados (por ejemplo através de NTP) o que arrancaran en el mismo momento, lo cual daría lugar a falsos positivos que se ha preferido evitar.

Emparejamiento Falso: Se llamará así al resultado de un método que indica con seguridad que una pareja de direcciones no pertenecen a la misma máquina. Si se obtiene ese resultado para todos los posibles emparejamientos de una dirección se podrá decir que no hay otra dirección en la topología descubierta que pertenezca al mismo equipo. Se contará dicha dirección para el porcentaje de la columna *Falso*. Por ejemplo, si para dos direcciones IP se obtienen marcas de tiempo muy distantes, esto indicaría que pertenecen a máquinas con instantes de arranque diferentes y por lo tanto independientes. Sin embargo, si los tiempos son iguales, no se puede asegurar que sean la misma máquina. La técnica **PORT_UNREACH**, empleada en [6] y [15], no es fiable para dar emparejamientos falsos pues se han encontrado equipos cuyos diferentes interfaces

[2] https://www.tlm.unavarra.es

Tabla 1: Resultados de pruebas de validación

Método	*Cierto*	*Falso*	*Posible Falso*	*Error*	*No Concluyente*	*Ciertos acumulados*	*Falsos acumulados*
PORT_UNREACH	83.3	-	16.7	0	-	83.3	-
IP_IDs (ALLY)	100.0	0	-	0	0	100.0	-
IP_IDs (UDP)	37.5	0	-	62.5	0	100.0	0
IP_IDs (ECHO)	16.6	0	-	0	83.4	100.0	0
IP_IDs (TIME)	8.4	0	-	0	91.6	100.0	0
IP_IDs (TCP)	100.0	0	-	0	0	100.0	0
TSTAMP (TCP)	-	0	-	100.0	0	100.0	0
TSTAMP (TIME)	-	0	-	0	100.0	100.0	0
Acumulado	100.0	0			0		

contestan con errores ICMP a la misma máquina desde direcciones IP diferentes.

Emparejamientos Posiblemente Falsos: Se engloban en esta categoría resultados que generalmente se podrían considerar como emparejamientos falsos pero que se ha comprobado que en configuraciones inusuales dan falsos negativos. El caso registrado hasta el momento corresponde a los emparejamientos falsos dados por la prueba **PORT_UNREACH** que como se ha comentado no son completamente fiables.

Errores: Muchas pruebas son irrealizables con algunos equipos debido generalmente a filtrado de mensajes ICMP o a que la implementación de TCP/IP del router no soporte opciones de *timestamp*. Si todas las pruebas de emparejamiento de una dirección IP dan como resultado errores se clasifica la dirección como *Error*.

Resultados No Concluyentes: Algunas pruebas pueden otorgar resultados que no permitan concluir si la dirección es emparejable con otra o no. Tal es el caso por ejemplo de los resultados positivos en las pruebas basadas en *timestamps*. Empleando la prueba **TSTAMP (TCP)**, dos máquinas que se hayan arrancado en el mismo instante pueden devolver valores que hagan pensar erróneamente que son la misma. Se clasifica ese tipo de resultados "positivos" como *No Concluyentes*. La prueba **TSTAMP (TIME)**, basada en que las secuencias de marcas de tiempos estén ordenadas, de nuevo puede dar un falso positivo si los relojes están sincronizados. Finalmente, algunos equipos, para las pruebas **IP_IDs** devuelven en el campo de identificación de IP números que no están generados según una secuencia autoincremental. Esos resultados pueden dar falsos positivos debidos a dos máquinas con comportamientos análogos por lo que de nuevo, si no hay ningún resultado positivo ni errores en alguna prueba y hay alguno de estos comportamientos anómalos se clasificará a esa dirección IP como de resultado de emparejamiento *No Concluyente*.

La Tabla 1 muestra los resultados de todas las pruebas en la topología de red controlada, siguiendo la clasificación descrita. En cada fila las cinco primeras columnas de datos deben sumar 100 % pues cada dirección IP obtiene un resultado de clasificación con cada método (ha sido emparejada, no lo ha sido, ha dado error, etc). El objetivo de identificación es lograr que todas las direcciones IP, con algún método, aparezcan en la columna de *Ciertos* o de *Falsos* y que no haya inconsistencias entre los resultados (un método empareje la dirección con otra y un segundo método diga tajantemente que no tiene pareja posible). Hay que resaltar que no es imprescindible que todas las direcciones IP encuentren una pareja (100 % de *Ciertos*) ya que pueden existir algunas para las que no se haya descubierto otra dirección del mismo equipo. Si ese tipo de direcciones han sido clasificadas por algún método como *Falso* se habrá logrado el reconocimiento de sus alias (que en este caso es ninguno). Se ha comprobado que no se ha producido ningún resultado de inconsistencia entre las pruebas. Además, como se ve en la tabla, no hay ningún reconocimiento de *Falsos* en la topología controlada. Esto se debe a que se trabaja con todas las direcciones y hay al menos dos de cada equipo.

En la aplicación de los distintos métodos se repiten resultados de *Ciertos* y *Falsos*. Por ello en la Tabla 1 se muestran dos columnas adicionales donde se contabilizan como número acumulado (*ciertos acumulados* y *falsos acumulados*). De esta manera se puede comprobar cuántos nuevos resultados añade una técnica de la tabla frente a las anteriores. Además, la acumulación de emparejamientos falsos entre diferentes técnicas puede colaborar a la identificación de direcciones que no tengan pareja posible.

Como muestra la Tabla 1, tanto el método **IP_IDs (ALLY)** como el **IP_IDs (TCP)** logran la identificación de todas las parejas de interfaces de red. Sin embargo, los sistemas operativos de la mayoría de los routers de los que se dispone no emplean la opción de *timestamp* en los segmentos TCP (100 % de errores en la prueba **TSTAMP (TCP)**). Por otro lado todas las pruebas con el método **TSTAMP (TIME)** basado en los *timestamps* ICMP se han marcado como *No Concluyentes*. Esto se debe a que con este método no se aceptan resultados *Ciertos* ya que no se puede saber si se deben a la sincronización de relojes de diferentes máquinas. Sin embargo, en este escenario controlado se sabe que los relojes no están sincronizados. Teniendo esto en cuenta, el método sí reconocería el 100 % de los alias.

Se ha planteado un entorno muy optimista en el que prácticamente todos los equipos son del mismo fabri-

cante (lo cual limita las idiosincrasias) y con configuraciones de seguridad muy permisivas, con filtrados nulos de paquetes o los que haga por defecto el sistema operativo del equipo. Es interesante resaltar que en este escenario el método tradicional **PORT_UNREACH** resuelve solo el 83.3 % de los alias y que incluyendo el resto de propuestas de la literatura (**IP_IDs (ALLY)**) se alcanza el 100 %. Es decir, en un escenario sin filtrados en los nodos, con los métodos tradicionales se logra la resolución completa de alias.

4. ETOMIC como instrumento de medida

Una vez puesta a prueba la metodología en un entorno controlado donde la validación es factible se ofrecen resultados obtenidos directamente con Internet. Para ello se ha empleado la plataforma de monitorización ETOMIC [3]. Esta plataforma ha sido desarrollada dentro del Proyecto Integrado "EVERGROW" del VI Programa Marco de la Unión Europea. ETOMIC consiste en un sistema central de gestión y un conjunto de nodos de monitorización distribuidos por toda Europa [18]. En la actualidad se dispone de 17 nodos en localizaciones escogidas, principalmente en universidades, centros de investigación, operadores y empresas de telecomunicaciones (Fig. 3). ETOMIC ofrece una plataforma de monitorización abierta a la comunidad investigadora, capaz de realizar medidas activas, totalmente reconfigurable y que dispone de un hardware de generación y captura de tráfico de alta precisión, tanto en temporización como en sincronización, empleando para ello receptores GPS e interfaces de red *ad-hoc*.

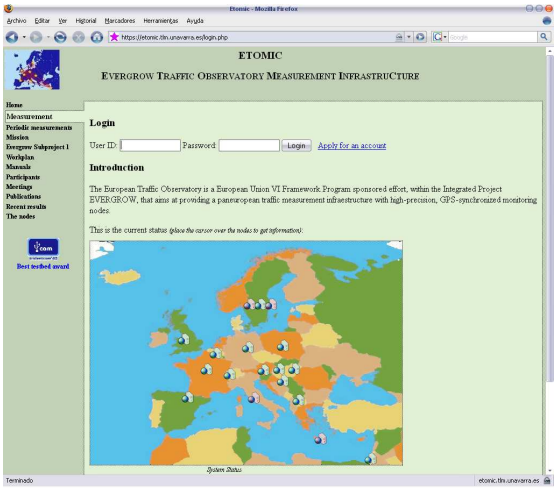

Figura 3: Interfaz de ETOMIC

Se han empleado los nodos de medida de esta plataforma para descubrir la topología de la red que los interconecta y reconocer los alias de los diferentes routers de la misma. La escala del problema en una situación paneuropea como la presentada es muy superior a la de la red de laboratorio controlado ya analizada. El número de nodos intermedios descubiertos mediante la utilidad *traceroute* en Marzo de 2007 es de 114, lo cual hace muy costosa (en tiempo y tráfico introducido en la red) la comprobación de todas las parejas con todos los métodos presentados. Por ello se van a limitar las parejas a comprobar a un subconjunto de parejas "potenciales".

Se comprueba cada dirección IP descubierta en un sentido ($SX \rightarrow SY$) con la que debería corresponder del *traceroute* en sentido contrario ($SY \rightarrow SX$) si el camino fuera simétrico. Como el camino puede no ser simétrico, se comprueba cada dirección con varias de las descubiertas en el sentido contrario; aquellas alrededor de la posición donde estaría el mismo router en caso de camino simétrico. En Fig. 4 se ve un ejemplo donde la dirección a comparar es la número 3 (rodeada con un círculo) del camino de izquierda a derecha. Ésta se compara con la de posición 4 del camino inverso junto con las adyacentes a distancia 1 salto (equipos marcados en negro). Para los resultados finales se comprueba cada dirección IP con 9 candidatas escogidas por proximidad (distancia 4 saltos) con la posición donde se esperaría encontrar ese router en caso de rutas simétricas.

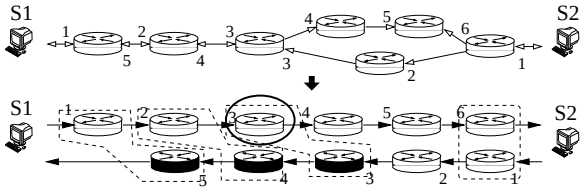

Figura 4: Emparejamientos potenciales

Los resultados se presentan a continuación siguiendo el mismo formato de la sección 3.

5. Resultados en escenario real

La Tabla 2 muestra los resultados obtenidos con la red europea, para los que de nuevo no se ha detectado ningúna inconsistencia. Lo primero que se observa es que el método clásico **PORT_UNREACH**, en un escenario real, tiene una tasa de descubrimientos muy baja, inferior a un 9 %. Esto se debe a que gran número de equipos en redes en producción tienen filtrado el envío de los mensajes que requiere esta metodología, lo cual produce un porcentaje de errores muy elevado (91.2 %).

La segunda técnica propuesta en la literatura, **IP_IDs (ALLY)**, amplía el porcentaje de reconocimientos hasta un 15 %. Este incremento proviene de interfaces que responden a los datagramas UDP pero que no habían sido reconocidos con el método anterior como de la misma máquina por no emplear la misma dirección origen. Con esta técnica se reconoce que son la misma y que la diferencia de dirección origen se debía a la implementación de IP.

La primera técnica nueva propuesta en este trabajo, **IP_IDs (UDP)**, es capaz de aportar un reducido núme-

[3] http://www.etomic.org

Tabla 2: Resultados de identificación por IP en la red que emplea ETOMIC

Método	*Cierto*	*Falso*	*Posible Falso*	*Error*	*No Concluyente*	*Ciertos acumulados*	*Falsos acumulados*	*Nodos*
PORT_UNREACH	8.8	-	0	91.2	-	8.8	-	108
IP_IDs (ALLY)	13.2	0	-	86.8	0	15.0	0	104
IP_IDs (UDP)	2.6	0	-	97.4	0	15.0	0	104
IP_IDs (ECHO)	21.2	4.4	-	29.2	45.2	36.2	5.3	90
IP_IDs (TIME)	17.6	0	-	41.5	40.9	36.2	5.3	90
IP_IDs (TCP)	25.6	0	-	74.4	0	47.7	5.3	83
TSTAMP (TCP)	-	0	-	100.0	0	47.7	6.1	83
TSTAMP (TIME)	-	0	-	35.3	64.7	47.7	6.1	83
Acumulado	47.7	6.1			46.2			83

ro de *Ciertos*. Esto se debe a que un gran número de interfaces no responden a suficientes mensajes UDP como para crear una secuencia de valores del campo de identificación que permita verificar con confianza la pertenencia a la misma máquina. Esto queda representado por la gran cantidad de errores contabilizados (para el 97.4 % de las direcciones). De los *Ciertos* obtenidos, ninguno es nuevo. Todos habían sido identificado con alguna de las técnicas anteriores.

La técnica de secuenciación de valores de identificación empleando mensajes ICMP *echo request*, **IP_IDs (ECHO)**, incrementa el porcentaje de identificaciones en un 21.2 %. Además este método ofrece un 4.4 % de direcciones que no tienen pareja entre las sondeadas. Si se añaden los emparejamientos falsos obtenidos con los métodos anteriores se alcanza un 5.3 % de falsos acumulados. La técnica **IP_IDs (TIME)** sin embargo no añade nuevas identificaciones, ni ciertas ni falsas, dado que los interfaces que identifica (un 17.6 % del total) ya lo estaban con alguno de los métodos anteriores. Ambas técnicas sí aportan datos sobre bastantes más parejas de direcciones pero no llegan a ser concluyentes.

La mejor tasa de resultados se obtiene con las máquinas que responden a segmentos TCP. Con el método **IP_IDs (TCP)** se logra un reconocimiento para un 25.6 % de las direcciones. Además resuelve un 11.5 % más de alias que no se habían identificado con los métodos anteriores.

Se observa que en general no se encuentran interfaces que respondan a los segmentos TCP con la opción *timestamp* y que no se logra extraer conclusiones de los datos obtenidos con la técnica **TSTAMP (TIME)**.

Atendiendo a la última fila de la Tabla 2 se observa como resultado global que para un 47.7 % de las direcciones IP descubiertas se ha encontrado otra que pertenece a la misma máquina. Esto representa una mejora en al menos un factor de 3 respecto de lo obtenido (un 15 %) con las técnicas anteriores. Aunque no se logre encontrar directamente todas las parejas de una dirección, el resultado puede mejorar aplicando transitividad (si A y B son interfaces de R1 y B y C son de R2 entonces R1=R2). Además un 6.1 % de las direcciones se sabe que no tienen ninguna pareja de entre las comprobadas. Para el 46.2 % restante se obtiene información, es decir, en general han contestado a alguna de las pruebas, pero no se han obtenido resultados concluyentes que permitan clasificarlas.

Finalmente, se ha incluido una columna adicional en la Tabla 2 que cuenta el número de nodos que se representarían en un grafo de la topología de red. Como se ha comentado con anterioridad, contando todas las direcciones descubiertas por la utilidad *traceroute* como nodos independientes se tiene un total de 114. Aplicando las técnicas tradicionales de identificación se reduce esta cifra a 104 nodos. Con la información aportada por los nuevos métodos incorporados se llega a una topología de 83 nodos gracias a la identificación de numerosas direcciones como pertenecientes a un número reducido de routers. Es decir, las técnicas tradicionales son capaces de eliminar 10 nodos inexistentes del grafo mientras que los procedimientos descritos en este artículo han permitido retirar 21 nodos *adicionales*. No se incluye una figura del cambio en el grafo de topología resultante debido a la complejidad de visualización de un grafo con 83 ó 114 nodos.

6. Conclusiones

En este artículo se ha descrito la problemática de identificación de los diferentes interfaces que pertenecen a un mismo router y cómo es determinante a la hora de describir el grafo de una topología de red. Se han analizado las técnicas existentes en la literatura para la resolución de este problema y se ha visto que en un escenario real actual de Internet tan solo son efectivas en el 15 % de las situaciones. La ampliación propuesta a estos métodos ha permitido incrementar dicha efectividad hasta un 47.7 % (un factor de 3) empleando nuevos sondeos a los equipos de red y procesados más sofisticados para los datos obtenidos.

Agradecimientos

Este trabajo ha sido financiado por el Proyecto Integrado Evergrow (contrato 001935) del Programa FP6/IST/FET de la Comisión Europea y parcialmente por los proyectos del Plan Nacional TEC2004-05622-C04-04 y TEC2004-06437-C05-03/TCM.

Referencias

[1] H.V. Madhyastha, T. Anderson, A. Krishnamurthy, N. Spring, and A. Venkataramani. A structural approach to latency prediction. In *Proc. USENIX Internet Measurement Conference*, 2006.

[2] E. Katz-Bassett, J.P. John, A. Krishnamurthy, D. Wetherall, T. Anderson, and Y. Chawathe. Towards IP geolocation using delay and topology measurements. In *Proc. USENIX Internet Measurement Conference*, 2006.

[3] T.J. Shi and G. Mohan. An efficient traffic engineering approach based on flow distribution and splitting in MPLS networks. *Computer Communications*, 29(9):1284–1291, May 2006.

[4] L. Garces-Erice, K.W. Ross, E.W. Biersack, P.A. Felber, and G. Urvoy-Keller. Topology-centric look-up service. In *Proc. COST264/ACM Fifth International Workshop on Networked Group Communications*, 2003.

[5] K. Jia, L. Mason, and Y. Qin. Two-layer restoration scheme for IP over optical networks with MPLS. In *Proc. the 8th International Conference on Communication Systems*, volume 2, pages 25–28, November 2002.

[6] J.-J. Pansiot and D. Grad. On routes and multicast trees in the Internet. *ACM SIGCOMM Computer Communication Review*, 28(1):41–50, January 1998.

[7] N. Spring, R. Mahajan, and D. Wetherall. Measuring ISP topologies with Rocketfuel. In *Proc. ACM SIGCOMM*, 2002.

[8] Y. Breitbart, M. Garofalakis, B. Jai, C. Martin, R. Rastogi, and A. Silberschatz. Topology discovery in heterogeneous IP networks: The NetInventory system. *IEEE/ACM Transactions on Networking*, 12(3):401–414, June 2004.

[9] R. Govindan and H. Tangmunarunkit. Heuristics for internet map discovery. In *Proc. IEEE INFOCOM*, 2000.

[10] N. Spring, R. Mahajan, and T. Anderson. Quantifying the causes of path inflation. In *Proc. SIGCOMM*, 2003.

[11] Y. Jiang, B. Fang, and M. Hu. Techniques in mapping router-level internet topology from multiple vantage points. In *LNCS 3320*, 2004.

[12] A. Broido and K.C. Claffy. Internet topology: Connectivity of ip graph. In *Proc. SPIE International Symposium on Convergence of IT and Communication*, August 2001.

[13] V. Jacobson. Traceroute ftp://ftp.ee.lbl.gov/traceroute.tar.gz, October 1989.

[14] F. Baker. Requirements for IP version 4 routers. RFC 1812, June 1995.

[15] B. Huffaker, D. Plummer, D. Moore, and K. Claffy. Topology discovery by active probing. In *Proc. the Symposium on Applications and the Internet (SAINT)*, January 2002.

[16] J. Leguay, M. Latapy, T. Friedman, and K. Salamatian. Describing and simulating internet routes. In *Proc. IFIP Networking*, May 2005.

[17] V. Jacobson, R. Braden, and D. Borman. TCP extensions for high performance. RFC 1323, May 1992.

[18] D. Morato, E. Magaña, M. Izal, J. Aracil, F.Ñaranjo, F. Astiz, U. Alonso, I. Csabai, P. Haga, G. Somin, J. Seger, and G. Vattay. The European Traffic Observatory InfraestruCture (ETOMIC): A testbed for universal active and passive measurements. In *Proc. TRIDENTCOM 2005*, pages 283–289, 2005.