Roaming mit gleich bleibender IP-Adresse

Laufend online

Mit Mobile IP behält ein Laptop seine IP-Adresse, wenn er in ein neues Netzwerk wechselt. Selbst beim Übergang von einem WLAN in ein Ethernet bleiben die TCP-Verbindungen bestehen. Bei TMip (Transparent Mobile IP) funktioniert das sogar ohne Eingriffe am Client. Ralf Spenneberg

Wer mit seinem Laptop mobil ist, hat meist verschiedene Anschlüsse, um sich ans Internet anzubinden. Je nach Ort stehen WLAN, drahtgebundenes Ethernet oder Modem zur Verfügung. Dabei wechselt auch jedes Mal die eigene IP-Adresse. Das ist ohne Zusatzaufwand auch nicht zu vermeiden, da jeder Zugang einen eigenen IP-Adressbereich verwendet.

Adresswechsel

Ändert sich die eigene Adresse, verlieren aber die Benutzer alle aktiven Netzwerkverbindungen. Mobile-IP-Techniken vermeiden dies: Der Client erhält eine gleich bleibende Heimatadresse, über die er immer erreichbar ist. Verbindungen über diese IP bleiben selbst nach einem Netzwechsel bestehen. Das erste offizielle RFC zu diesem Thema wurde bereits 1996 veröffentlicht. Die aktuelle Fassung in RFC 3344 stammt vom August 2002 und beschreibt die Mobile-IP-Unterstützung in IPv4 [7]. Eine sehr interessante Alternative ist TMip (Transparent Mobile IP) [1], dazu später mehr.

Eine mobile Maschine wird bei Mobile IP immer durch ihre Heimat-IP-Adresse identifiziert. Diese Adresse ist unabhängig von ihrem aktuellen Aufenthaltsort. Befindet sich der Rechner in seinem Heimatnetzwerk, kommuniziert er ganz normal mit anderen Computern.

Befindet sich die mobile Station in einem fremden Netzwerk, benötigt sie für die Kommunikation einen Mittelsmann, den Home-Agent. Dieser Agent kennt den tatsächlichen Aufenthaltsort des mobilen Rechners und leitet die Pakete über einen Tunnel an ihn weiter. Die Gegenstelle des Tunnels ist ein weiterer Mittelsmann: der Foreign-Agent. Er befindet sich in dem Netzwerk, in dem sich der mobile Rechner aufhält.

Sobald der Client erkennt, dass er sich in einem fremden Netzwerk befindet, registriert er sich über den Foreign-Agent bei seinem Home-Agent. Der Home-Agent speichert die IP-Adresse des Foreign-Agent als so genannte Care-of Address. Er entscheidet auch, ob die Registrierung überhaupt erlaubt ist. Wenn ja, arbeitet der Home-Agent als Mittelsmann. Hierzu fängt er alle für den Client bestimmten Pakete ab und tunnelt sie an die Care-of-Adresse, also an den Foreign-Agent. Der nimmt die Pakete entgegen und leitet sie dann an den mobilen Client weiter.

Für die Funktion des Mobile-IP-Protokolls sind daher drei Komponenten erforderlich: Ein Home- und ein Foreign-Agent sowie ein Mobile-Client, der selbstständig erkennt, in welchem Netzwerk er sich befindet.

Implementierungen

Es gibt seit einigen Jahren mehrere Implementierungen des Mobile-IP-Protokolls für Linux. Ein Großteil dieser Programme wird jedoch nicht aktiv weiterentwickelt. Einen guten Überblick über die einzelnen Varianten gibt Jean Tourrilhes [4]. Auch er hat seine Mobile-IP-Implementierung seit 1997 nicht mehr weiterentwickelt.

Ein bekanntes Beispiel ist Mosquitonet [2], es unterstützt aber lediglich die Linux-Kernel 2.0 und 2.2. Etwas weiter ist Dynamics HUT Mobile IP [3] der Helsinki-Universität, diese Implementierung unterstützt wenigstens Kernel 2.2 und 2.4. Die Software läuft sogar auf Microsoft-Betriebssystemen, wenn die Cygwin-DLLs installiert sind. Leider hat die Helsinki-Universität im Oktober 2001 die Entwicklung eingestellt.

Transparent Mobile IP

Eine neue, aber nicht standardkonforme Entwicklung ist Transparent Mobile IP (TMip, [1]). Das unter der BSD-Lizenz stehende Projekt ist auf kleinere lokale Bereiche ausgerichtet, in denen die mobilen Clients häufig zwischen verschiedenen WLANs und drahtgebundenen Netzwerken wechseln. Auf dem mobilen Client sind für TMip keinerlei Eingriffe nötig: Es realisiert die Mobile-IP-Funktionalität vollständig durch zentrale

Dienste. Am einfachsten ist die Konfiguration mit DHCP-fähigen Clients (Dynamic Host Configuration Protocol).

Ein TMip-Netzwerk besteht aus drei Komponenten: MLR (Mobile Location Register), CN (Correspondent Nodes) und MS (mobile Stationen). Das MLR speichert zu jeder Zeit den Ort aller MS. Sobald sich eine neue mobile Station in einem Netz befindet, bemerkt das der CN dieses Netzwerks und registriert die MS im zentralen MLR.

Das MLR dient als Gedächtnis des TMip-Netzwerks. Es verteilt die IP-Adressen per DHCP, speichert den Aufenthaltsort der mobilen Stationen und reagiert auf die Wanderung einer MS von einem CN zu einem anderen. Da ein Ausfall des MLR für die Funktion des Netzwerks fatal ist, unterstützt MLR einen Carbon-Copy-Modus. Damit überträgt das primäre MLR seine Datenbank automatisch auf ein sekundäres MLR. Fällt das primäre Register aus, kann das sekundäre seine Aufgabe übernehmen. Nach dem Neustart holt sich das primäre MLR die Daten wieder zurück.

Mobile Location Register

Um TMip in Betrieb zu nehmen, ist etwas Installations- und Konfigurationsarbeit zu leisten. Voraussetzung für TMip ist die Libpcap, ist sie nicht installiert, bricht Make wegen fehlender Header ab. Zuerst ist das MLR an der Reihe. Der Aufruf »make« im Verzeichnis »mlrd« erzeugt den MLR-Daemon »mlrd«, den man manuell an eine geeignete Stelle kopieren muss (etwa »/usr/local/sbin«). Ein »make install« ist leider noch nicht implementiert.

Nun muss der Administrator die Konfigurationsdatei »mlrd.rc« für den MLR-Daemon anlegen. Listing 1 zeigt ein Beispiel für das sekundäre MLR, Listing 2 die Variante für das primäre MLR. Mit diesen Beispielen können primäres und sekundäres MLR auf derselben Maschine laufen, da sie unterschiedliche Portnummern verwenden. Die »grant«-Direktive in Listing 1 erlaubt es der primären MLR, die Datenbank der sekundären MLR zu verändern. Die »cc_mlr«-Anweisung in Listing 2 nutzt das: Sie sorgt dafür, dass der Secondary-Daemon auf dem aktuellen Stand bleibt.

Ausfallsicher durch Secondary Server

Der Primary-Daemon darf erst starten, wenn der Secondary bereits läuft. Welche Konfigurationsdatei der Daemon benutzt, legt die Option »-f« fest:

```
./mlrd -f mlrd-secondary.rc
./mlrd -f mlrd-primary.rc
```

Gibt einer der MLR-Server eine Fehlermeldung aus, sind häufig fehlende Zugriffsrechte im Logging-Verzeichnis die Ursache. In der Protokolldatei zeigt sich

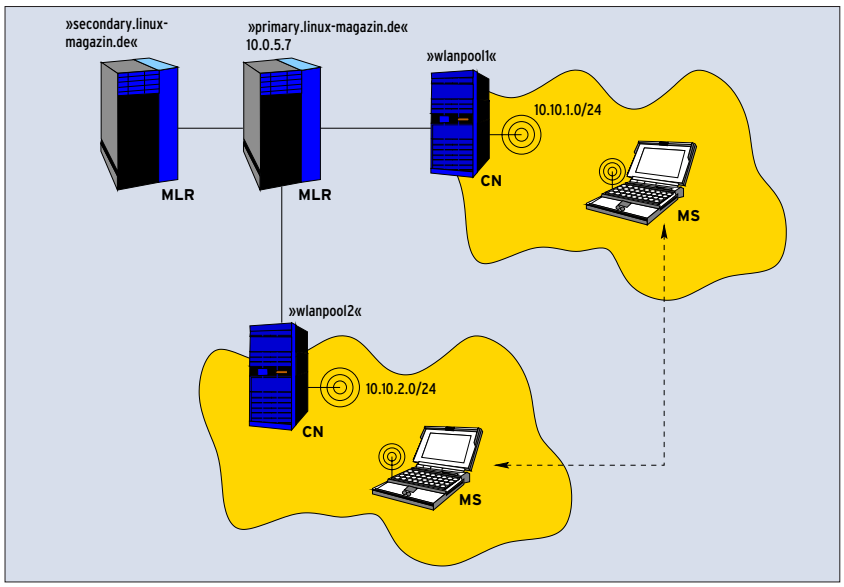

Abbildung 1: TMip erlaubt es der MS (Mobile Station), mit gleich bleibender IP-Adresse zwischen Netzwerken zu wechseln. MLR (Mobile Location Register) und CN (Correspondent Nodes) kümmern sich um die Details.

unter anderem, ob der Carbon-Copy-Modus funktioniert. Ist während der Laufzeit eine Änderung der Konfiguration nötig, reagiert das MLR auf ein Sighup-Signal und liest das Konfig-File neu.

Correspondent Nodes

Während das MLR die Zentrale für alle Netze darstellt, zwischen denen die mobilen Clients wandern, ist ein CN (Correspondent Node) für jeweils ein Netz zuständig. Er stellt die Verbindung der mobilen Stationen mit dem Netzwerk her. Dazu benötigt er üblicherweise zwei Netzwerkkarten: Eine ist mit dem Backbone verbunden (CN-Seite), während die zweite Karte die mobilen Stationen versorgt. Sie kann drahtgebunden oder drahtlos arbeiten (WLAN).

Das folgende Beispiel geht von zwei CNs aus (siehe Abbildung 1). Ein »make«-Aufruf im »tmipd«-Verzeichnis übersetzt den CN-Daemon. Die ausführbare Datei »tmipd« muss der Admin an eine geeignete Stelle kopieren, etwa »/usr/local/sbin«. Listing 3 zeigt eine passende Konfigurationsdatei: »cn_if« benennt das Interface des CN, über diesen Weg sind die anderen CNs und das MLR zu erreichen. Das Interface zu den mobilen Stationen ist in »mobile_if« angegeben.

Listing 1: Sekundäres MLR

```
01 # mlrd-secondary.rc
02 network_name linux-magazin
03 port 5555
04 foreground false
05 log_file /var/log/mlrd.log
06 status_file /var/log/mlrd.status
07 log true
08 grant primary.linux-magazin.de
```

Listing 2: Primäres MLR

```
01 # mlrd-primary.rc
02 network_name linux-magazin
03 port 6554
04 foreground false
05 log_file /var/log/mlrd.log
06 status_file /var/log/mlrd.status
07 log true
08 cc_mlr secondary.linux-magazin.de:5555
```

Listing 3: Konfiguration des CN

```
01 # tmipd.rc
02 mlr              primary.linux-magazin.de
03 cn_name          wlanpool1.linux-magazin
04 cn_if            eth0
05 mobile_if        wlan0
06 network_name     linux-magazin
07 addr_pool        wlan0 * *
08 dns_server       10.0.5.7
09 log_file         /var/log/tmipd.log
10 status_file      /var/log/tmipd.status
```

Listing 4: Roaming einer MS

```
01  -> Mobile station detected in my cell
    [00:20:E0:6C:72:1E] (via IP or ARP activity)
02   + Establishing host's address allocation
03     + Success -> Restored from MLR
04     + Notifying MLR of host's new status
05   + Attempting mobile host handover [migrated from
    10.10.1.1] parent 10.10.1.1
06     + Contacting transaction participants
07     + Requesting tunnel between parent CN and local CN
08       + Agreed to use tunnel type [MLRP_TUN_4IN4]
09     + All OK, going for commit
10     + Using 10.10.1.9 on 10.10.2.0/255.255.255.0
    gw:10.10.2.1
11     + Committing changes to MLR: Done.
12
13  -> Mobile host [00:01:f4:ee:90:44] has arrived in
    this cell
```

DHCP inklusive

Nach dem Start per »tmipd -f tmipd.rc« aktiviert der TMip-Daemon auch seinen DHCP-Server. Jede mobile Station, die nun eine DHCP-Anfrage an das »wlan0«-Interface des CN schickt, erhält von ihm eine IP-Adresse aus »addr_pool« (IP-Bereich dieses Interface). Der DHCP-Server teilt den Clients auch die Adresse des DNS-Servers mit, sie ist mit der Option »dns_server« konfiguriert.

Die Konnektivität des CN lässt sich mit der Option »-E« (Netzwerk-Evaluation) prüfen, Abbildung 2 zeigt einen erfolgreichen Ablauf. Nach der Prüfung beendet sich der »tmipd« wieder. Wichtig ist, dass der CN das MLR auf dem TCP-Port 6554 und die weiteren CNs auf Port 5554 erreicht und dass er IPIP-Tunnel zu den weiteren CNs aufbauen darf. Dazu sind möglicherweise Firewallregeln anzupassen und auf den CN-Rechnern muss auch der IPIP-Support aktiviert sein (Abbildung 3).

Zudem kann das Anti-Spoofing Probleme bereiten. Es ist sinnvoll, die entsprechende Option im Kernel zu deaktivieren:

```
for i in /proc/sys/net/ipv4/conf/*/rp_filter; do echo "0" > $i; done
```

Im Fehlerfall liefert die Protokolldatei meist sehr ausführliche Hinweise für die Fehlersuche. Genügt das nicht, kann der Admin »tmipd« und »mlrpd« mit der Option »-F« starten. Dann verbleiben beide Anwendungen im Vordergrund und schreiben ihre Fehlermeldungen auf dem Bildschirm. In diesem Modus lassen sich beide Programme sogar mit Befehlen direkt steuern, beispielsweise sorgt »debug 4« für ausführlichere Protokolle. Die Tastenkombination [Strg] + [D] beendet die Applikationen.

Roaming

Auf der mobilen Station muss der DHCP-Client aktiviert sein. Der Rechner erhält vom CN seines aktuellen Netzes eine lokale IP-Adresse. Solange sich die MS im Bereich des CN aufhält (drahtgebunden oder drahtlos), verhält sich TMip wie ein gewöhnlicher DHCP-Daemon. Spannend wird es, wenn die mobile Station in ein anderes Netz wandert.

Im neuen Netz soll die Station ihre IP-Adresse behalten und ohne Unterbrechung weiter kommunizieren. Der CN-Daemon bemerkt ihre MAC-Adresse und informiert das MLR. Von dort erfährt der CN, dass die Station bereits bekannt ist. Da sie im neuen Netz mit ihrer alten IP nicht direkt kommunizieren kann, öffnet der CN einen IPIP-Tunnel in das Heimatnetz und leitet die Paket dorthin weiter. Die Gegenstelle im Heimatnetz schickt Pakete, die für die MS bestimmt sind, ebenfalls durch den Tunnel in das aktuelle Netz. Listing 4 zeigt das Protokoll eines solchen Roaming-Vorgangs aus Sicht des CN im neuen Netz.

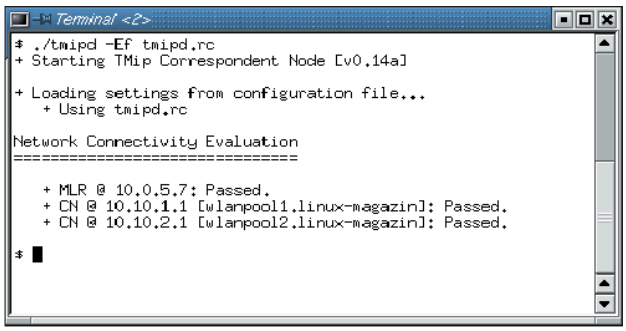

Abbildung 2: Mit der Option »-E« überprüft ein CN, ob er das MLR und die anderen CNs erreicht. Hier waren alle Verbindungen erfolgreich.

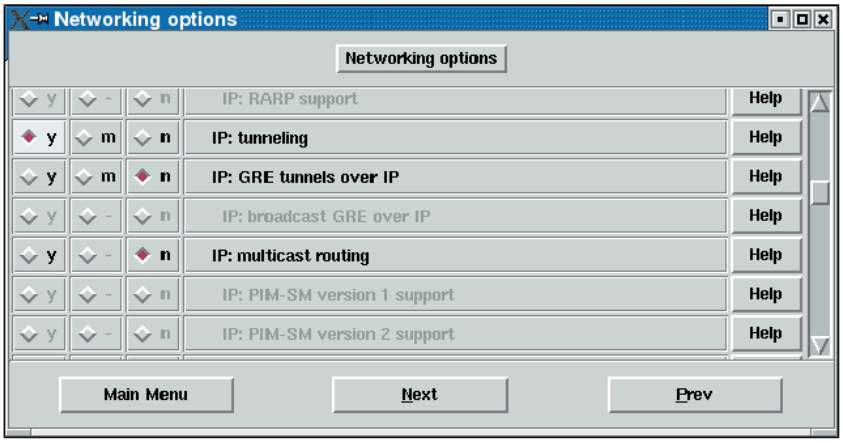

Abbildung 3: Für die Kommunikation zwischen den CNs muss der Kernel IPIP unterstützen. Die passende Option versteckt sich unter den »Networking Options« und heißt »IP: tunneling«.

Findet beim Roaming auch ein Medienwechsel statt (etwa zwischen Ethernet und WLAN), dann kann es auf der mobilen Station leider zu einem Problem kommen: Ethernet- und Wireless-Netzwerkkarte müssen dieselbe IP-Adresse verwenden. Damit das funktioniert, muss der Client selbst feststellen, welche Karte gerade aktiv ist, und über diese seine Pakete versenden.

Das Problem können Laptop-Benutzer mit PCMCIA-Karten umgehen. Sie tauschen die Karten je nach Bedarf einfach aus. Allerdings müssen sich beide Karten mit identischen MAC-Adressen melden, sodass sie per DHCP identische IP-Adressen erhalten. In der Regel ist es kein Problem, als MAC-Adresse der WLAN-Karte die des Ethernet-Interface zu setzen:

```
ifconfig wlan0 hw ether MAC-Adresse
```

Ohne Medienwechsel funktioniert TMip aber hervorragend.

Fazit

Seit langem wollen Laptop-Benutzer beim Roaming die bereits aufgebauten Verbindungen weiter benutzen. Mobile IP leistet dies. Leider benötigt das Protokoll Anpassungen auf den mobilen Stationen, die bisher für die wenigsten Betriebssysteme zur Verfügung stehen. Auf lange Sicht wird Mobile IP vermutlich den Weg in die meisten IP-Protokollstacks finden. Wer transparentes Roaming heute schon ohne Änderung an den mobilen Stationen einsetzen will, findet in TMip ein passendes Produkt.

Transparent Mobile IP stellt ähnliche Dienste bereit – aber ohne den Mobile-IP-Standard selbst zu unterstützen. Das System erkennt selbstständig und automatisch die Migration einer mobilen Station von einem ins andere Netzwerk. Leider implementiert TMip keine Verschlüsselung für übertragene Daten. Das lässt sich aber mit herkömmlichen VPN-Lösungen wie Freeswan [5] oder OpenVPN [6] nachrüsten. *(fjl)*

Infos

[1] Transparent Mobile IP: [http://www.slyware.com/projects_tmip.shtml]

[2] Mosquitonet Mobile IP: [http://gunpowder.stanford.edu/mip/]

[3] Dynamics HUT Mobile IP: [http://www.cs.hut.fi/Research/Dynamics/]

[4] Jean Tourrilhes Mobile IP: [http://www.hpl.hp.com/personal/Jean_Tourrilhes/MobileIP/mip.html]

[5] Ralf Spenneberg, „Standard-Tunnel – VPN mit Linux 2.4 und Freeswan 2.01“: Linux-Magazin 10/03, S. 24

[6] Achim Leitner, „Offener Tunnel – VPN ohne Kernel-Modifikation: OpenVPN“: Linux-Magazin 10/03, S. 29

[7] RFC 3344, „P Mobility Support for IPv4“: [http://www.ietf.org/rfc/rfc3344.txt]

Der Autor

Ralf Spenneberg arbeitet als freier Unix/Linux-Trainer und Autor. Er veröffentlichte 2002 sein erstes Buch „Intrusion Detection für Linux-Server“. Vor wenigen Wochen erschien sein zweites Buch „VPN mit Linux“.