

IPv6 mit festen Adressen: So nutzt man SixXS-Tunnel

SixXS bietet eine kostenlose IPv6-Anbindung an das Internet. ZDNet erklärt Schritt für Schritt, wie man diesen Dienst von jedem festen oder mobilen Anschluss nutzt, und wie sich eine Fritzbox oder ein Unix-Rechner als IPv6-Router einsetzen lässt.

von [Christoph H. Hochstätter](#) am [6. November 2009](#), 16:29 Uhr

Am neuen Internet-Protokoll in der Version 6 (IPv6) wird man bereits in wenigen Jahren nicht mehr vorbei kommen. Der Grund dafür ist recht einfach: Die bisherigen IPv4-Adressen werden langsam knapp. Verschiedenen Schätzungen zufolge wird es auch in Europa bereits im Jahr 2012 nicht mehr ohne weiteres möglich sein, IPv4-Adressen zu bekommen.

In anderen Regionen der Welt, beispielsweise Asien, herrscht bereits Knappheit an IPv4-Adressen. Anders als man es in Europa gewohnt ist, bekommen private Internet-Nutzer dort keine öffentliche IPv4-Adresse mehr. Sie müssen sich mit einer privaten Adresse meist aus dem Bereich 10.0.0.0/8 begnügen und erhalten den Internet-Zugang nur via NAT-Routing.

Diese Praxis hat sich in Deutschland auch bei vielen UMTS-Anbietern etabliert. T-Online und O2 vergeben grundsätzlich nur noch private IPv4-Adressen. Bei Vodafone ist die Vergabe einer privaten oder öffentlichen IPv4-Adresse vom Tarif abhängig. E-Plus vergibt ausschließlich öffentliche IPv4-Adressen.

DSL-Nutzer erhalten in Deutschland meist genau eine IPv4-Adresse, die bei den großen Anbietern dynamisch vergeben wird. Feste IPv4-Adressen gibt es nur bei einigen regionalen Anbietern, etwa [M-Net](#), sowie bei vielen überregionalen Anbietern, die allerdings einen Telekom-Anschluss voraussetzen. Dazu gehören unter anderem [Tal.de](#), [Manitu](#) und [Snafu](#).

Wer mehrere Rechner an einem DSL-Anschluss betreiben möchte, kann das mit einem NAT-Router. Da der eigene NAT-Router jedoch eine öffentliche IPv4-Adresse besitzt, lässt sich die Erreichbarkeit einzelner Dienste mittels DynDNS und Portforwarding realisieren. Bekommt man bereits eine private Adresse vom Provider, ist die Erreichbarkeit eines Rechners ausgeschlossen. Mit IPv6 entfallen alle diese Beschränkungen. Eine IPv6-Adresse ist 128 Bit lang. Jeder private Anwender kann dabei ein eigenes /64-Netz erhalten. Kunden, die mehrere Netze benötigen, weil Routing erforderlich ist, werden mit einem /48-Netzblock versorgt. Somit ist es möglich, dass jeder Rechner in einem privaten Netzwerk eine eigene öffentliche IPv6-Adresse erhält, die weltweit erreichbar ist. Darüber hinaus gibt es keinen technischen Grund, weiterhin ausschließlich dynamische IP-Adressen für Endanwender bereitzustellen.

Eine einfache technische und zudem kostenlose Möglichkeit, bereits heute IPv6 zu nutzen, besteht darin, einen [6to4-Tunnel](#) einzusetzen. Das ist mit jedem DSL- oder Kabelanbieter möglich. Voraussetzung ist allerdings, dass der 6to4-Router die öffentliche IPv4-Adresse des Anbieters bekommt oder der NAT-Router in der Lage ist, das IP-Protokoll 41 einem dedizierten 6to4-Router zur Verfügung zu stellen. Das können die meisten Router nicht.

Für die Fritzbox 7270 gibt es eine [Laborfirmware](#), die 6to4 beherrscht. Somit ist es möglich, eine [6to4-Anbindung mittels einer Fritzbox 7270](#) sehr leicht zu realisieren. So bekommen alle Rechner im Heimnetzwerk IPv6-Connectivity ins Internet.

Die Freude über eine IPv6-Anbindung kann jedoch schnell getrübt werden, wenn man von seinem Anbieter eine dynamische IPv4-Adresse bekommt. Da die IPv4-Adresse in der IPv6-Adresse kodiert ist, ändern sich die IPv6-Adressen jedes Mal, wenn sich die IPv4-Adresse ändert.

Wer beispielsweise die IPv4-Adresse 1.2.3.4 bekommt, erhält den IPv6-Adressraum 2002:0102:0304::/48. Hat man 24 Stunden später die IPv4-Adresse 5.6.7.8, so ändert sich der IPv6-Adressraum auf 2002:0506:0708::/48. Da es für IPv6 noch keine DynDNS-Lösungen gibt, lässt sich auf diese Weise keine dauerhafte Erreichbarkeit der eigenen Rechner realisieren.

Wenn man feste IPv6-Adressen nutzen möchte, dann ist der Tunnel-Broker [SixXS](#) eine ideale Lösung. SixXS vergibt Einzeladressen, die man beispielweise für einen Laptop nutzen kann, und / 48-Netzwerke, mit denen sich das Heim- und Firmennetzwerke per IPv6 an das Internet anbinden lassen.

Das funktioniert mit jedem Internetzugang, auch wenn man nur eine private IPv4-Adresse zugewiesen bekommt. Wer häufig mit seinem Laptop unterwegs ist und verschiedene Internet-Zugänge wie UMTS oder WLAN-Hotspots nutzt, kann sich so leicht einen direkten IPv6-Zugang zu seinem Heim- oder Firmennetz schaffen und hat dann eine feste IPv6-Adresse.

Dasselbe lässt sich auch mit einer [VPN-Lösung](#) erzielen, jedoch ist eine IPv6-Anbindung eine Lösung, die man als getrost als zukunftsicher betrachten kann. Außerdem ist dies eine Gelegenheit, sich mit dem Internet-Protokoll der nicht mehr allzu fernen Zukunft vertraut zu machen.

Zudem ist man mit IPv6 äußerst flexibel. Viele VPN-Lösungen sind entweder auf den mobilen Zugang von Mitarbeitern oder auf die Verbindung von Unternehmensstandorten spezialisiert. Mit dem großen Adressraum von IPv6 ist sichergestellt, dass sich alle Rechner jederzeit erreichen können.

SixXS akzeptiert [Anmeldungen](#) von Privatpersonen, Firmen und ISPs. Bei der Anmeldung muss man jedoch eine Begründung angeben. Sie kann allerdings recht knapp gefasst sein. So reicht es beispielsweise aus, "I want to use IPv6 to connect my home network and my laptop computer which I use from mobile locations frequently" zu schreiben.

Wer bei Xing oder LinkedIn registriert ist, sollte in der Begründung für die Anmeldung einen Link zu seinem Profil angeben. SixXS arbeitet mit einem Punktesystem. Für jeden IPv6-Tunnel oder jedes Subnetz, das man anfordert, werden Punkte abgezogen. Wer seine Tunnel technisch sauber aufrecht erhält, bekommt wöchentlich Punkte gutgeschrieben.

Bei der Anmeldung erhält man 25 Punkte, die [IP SixXS Kredit](#) (ISK) genannt werden. Wer sein Xing- oder LinkedIn-Profil mitschickt, bekommt weitere 50 ISKs, da die Profile von SixXS als Identitätsnachweis anerkannt werden. Das Punktesystem dient dazu, dass man nicht hunderte von öffentlichen IPv6-Netzen anfordern kann, die ungenutzt bleiben, und helfen, den Dienst kostenlos zu halten.

SixXS bietet drei verschiedene Tunnelarten an. Die robusteste Tunnelvariante ist ein AYIYA-Tunnel, der das [Anything-in-anything-Protokoll](#) nutzt. AYIYA-Tunnel können von einem Rechner mit privater IPv4-Adresse aufgebaut werden, also hinter einem NAT-Router oder von einem UMTS-Zugang mit privater IPv4-Adresse. Sie haben aber auch den größten Protokoll-Overhead. Sie sind vor allem geeignet, Einzelrechner mit IPv6 zu versorgen. Sinnvoll ist das für Laptops, mit denen man häufig unterwegs ist.

Den geringsten Protokoll-Overhead haben statische Tunnel. Sie nutzen das 6in4-Protokoll, das dem 6to4-Protokoll sehr ähnlich ist. Der einzige Unterschied besteht darin, dass man keine 6to4-Adresse bekommt, sondern eine von einem SixXS-POP. Die technischen Voraussetzungen sind identisch mit denen von 6to4. Da 6to4 heutzutage von jedermann ohne Anmeldung genutzt werden kann, gibt es kaum praktische Anwendungsfälle für diesen Tunneltyp.

Ebenfalls auf dem 6in4-Protokoll setzen die sogenannten Heartbeat-Tunnel auf. Sie sind sinnvoll, wenn man über ein Heimnetzwerk verfügt, das über eine dynamische IPv4-Adresse verfügt, und der NAT-Router SixXS-Heartbeat-Tunnel verwalten kann. Über den Heartbeat wird die IPv4-Adresse regelmäßig überprüft. Wenn sie sich ändert, wird der IPv6-Tunnel auf die neue IPv4-Adresse geroutet.

Die [Fritzbox 7270 mit aktueller Laborfirmware](#) und alle Router mit [OpenWRT-Software](#) können als SixXS-Router dienen. Um die Fritzbox 7270 zu nutzen, muss man einen Heartbeat-Tunnel und ein dazugehöriges Subnetz bei SixXS anfordern. Die eigentliche Konfiguration der Fritzbox ist sehr

einfach. Es ist lediglich der Username, das Passwort und die Tunnel-ID einzutragen, die von SixXS zugeteilt wird, siehe [Bild 1](#). Die genaue Vorgehensweise gibt es auf der [AVM-Website](#).

Wenn die Konfiguration erfolgreich war, bekommen alle Rechner im Heimnetz automatisch eine öffentliche IPv6-Adresse zugewiesen. Allerdings hat die aktuelle Laborfirmware der Fritzbox 7270 eine "Zwangsfirewall" eingebaut, die verhindert, dass von außen IPv6-Verbindungen zu internen Rechnern aufgebaut werden.

Man kann zwar von intern Webseiten wie [ipv6.google.com](#) aufrufen, jedoch lässt sich kein Zugang, etwa von einem mobilen Notebook, zu den heimischen Rechner erreichen. Insofern muss die aktuelle Fritzbox-Labor-Firmware als unvollständig betrachtet werden, da man sie bestenfalls zum Kennenlernen der IPv6-Technologie nutzen kann.

Wer keinen Router mit IPv6-Unterstützung für SixXS-Tunnel hat, dem bieten sich zwei Möglichkeiten: Entweder bindet man jeden Rechner im Heimnetz mit einem eigenen AYIYA-Tunnel an IPv6 an oder versorgt sein Heimnetzwerk mit einem Rechner als Router.

Zum Aufbau des Tunnels dient die Software [AICCU](#). Sie ist verfügbar für Windows sowie die meisten gängigen Unix-Varianten wie Linux, Mac OS, FreeBSD, Solaris und AIX. Die Windows-Version hat allerdings einen Bug, der den Einsatz eines Windows-Rechners als IPv6-Router verhindert.

Als Voraussetzung für AICCU muss ein virtueller Tunnel-Netzwerkadapter auf dem System vorhanden sein. Alle gängigen Linux-Distributionen haben den notwendigen Treiber bereits installiert. Für Mac OS steht ein solcher Treiber zum [Download](#) bereit. Kompliziert wird es mit Windows: Es gibt eine recht komfortable AICCU-Version mit einem GUI, die AICCU als Windows-Dienst installiert. Sie ist allerdings aus dem Jahr 2006 und kommt nur mit den tun/tap-Treibern in der Version 8.01 zurecht, die bis einschließlich Windows XP laufen.

Darüber hinaus lässt sich eine direkte Portierung der Unix-Konsolen-Version nutzen, die auch mit den Treibern der Version 9 zurechtkommt. Diese neuen Treiber laufen auch mit Vista und Windows 7. Die 32-Bit-Version des Treiber kann man direkt [herunterladen](#). Die 64-Bit-Version befindet sich derzeit im Release-Candidate-Stadium. Sie gibt es nur im Bundle mit [OpenVPN](#). Um den Treiber allein zu installieren, lädt man OpenVPN [herunter](#) und startet den Installer. Bei der Auswahl der Komponenten wählt man ausschließlich den Treiber aus, siehe [Bild 2](#).

Installation von AICCU unter Windows

Zunächst lädt man die Konsolen-Version von AICCU [herunter](#). Diese Datei kann in ein beliebiges Verzeichnis, etwa C:AICCU, kopiert werden. Zweckmäßigerweise benennt sie in AICCU.EXE um. Die Config-Datei heißt aiccu.conf und gehört in das Verzeichnis C:WINDOWS. Sie sollte folgenden Inhalt haben:

```
username XXXXX
password YYYYY
ipv6_interface SixXS
tunnel_id ZZZZZ
verbose true
daemonize false
automatic true
requiretls false
```

Anzupassen sind lediglich die Werte für username, password und tunnel_id, die man von SixXS bekommt. Danach startet man den Client mit dem Kommandozeilenbefehl `aiccu start`, siehe [Bild 3](#). Das Kommandozeilenfenster muss zunächst unverändert bleiben. Zum Test der IPv6-Verbindung öffnet man ein neues Kommandozeilenfenster. Mit dem Befehl `netsh interface ipv6 show addresses sixxs` lässt sich ermitteln, ob dem virtuellen Netzwerkadapter die IPv6-Adresse des Tunnels zugewiesen wurde. [Bild 4](#) zeigt, dass neben der linklokalen Adresse aus dem Bereich `fe80::/10` auch eine öffentliche IPv6-Adresse zugeteilt wurde.

Mittels Traceroutes ist leicht festzustellen, ob die IPv6-Verbindung steht. Dazu nutzt man beispielsweise den Befehl `tracert -6 ipv6.google.com`, siehe [Bild 5](#). Ein weiterer beliebter Test ist der Aufruf der Website www.kame.net. Erreicht man die Seite mit IPv6, dann "tanzt" die abgebildete Schildkröte, siehe [Bild 6](#).

Wenn die Tests erfolgreich sind, ist es sinnvoll, AICCU als Windows-Dienst einzurichten. Dazu eignet beispielsweise SrvAny aus dem [Windows 2003 Resource Kit](#). Die im Kit enthaltenen Programme InstSrv und SrvAny laufen mit jeder Version ab Windows 2000 bis einschließlich Windows 7. Eine Anleitung hat [Microsoft](#) auf seinen [TechNet-Webseiten](#) veröffentlicht. So lässt sich der IPv6-Tunnel beim Einschalten des Rechners automatisch hochfahren.

Für viele Linux-Distributionen, etwa Fedora, Debian, Ubuntu oder OpenSuse, gibt es fertige Pakete auf der [AICCU-Download-Seite](#). Für nicht unterstützte Distributionen holt man sich den [Sourcecode](#) und kompiliert selbst. Dabei ist zu beachten, dass die Header und Libraries von gnutls benötigt werden, die in jeder Distribution zwar vorhanden sind, jedoch in der Regel manuell installiert werden müssen.

Unter Linux steht die Konfigurationsdatei aiccu.conf im Verzeichnis `/etc`. Der Aufbau ist im wesentlichen identisch mit der Windows-Version, jedoch muss `daemonize` in der Regel auf `true` gesetzt werden, es sei denn, die Readme-Datei des Distributionspaket schreibt etwas anderes vor.

Mit dem Befehl `aiccu start` lässt sich der IPv6-Tunnel testen. Der Befehl `ifconfig` zeigt an, ob das Interface `sixxs` die öffentliche IPv6-Adresse bekommen hat, siehe [Bild 7](#). Wenn man ein fertiges Paket einer Distribution heruntergeladen hat, lässt sich der Autostart nach dem Hochfahren des Rechner mit dem Runlevel-Editor der Distribution konfigurieren.

Installation von AICCU unter Mac OS X

Für Mac OS gibt es ein [fertig kompiliertes Binary](#). Das lässt sich mit dem Terminal durch die Eingabe folgender Befehle downloaden und installieren:

```
sudo -s
curl http://www.sixxs.net/archive/sixxs/aiccu/mac/aiccu-mac-i386.exe >/usr/sbin/aiccu
chown root:wheel /usr/sbin/aiccu
chmod 755 /usr/sbin/aiccu
```

Wie unter Linux muss die Config-Datei `/etc/aiccu.conf` heißen. Man kann sie mit `vi` oder `nano` editieren. Die Konfiguration wird wie bei Windows vorgenommen. `daemonize` muss also auf `false` stehen. Da in der Datei auch das Zugangspasswort zu SixXS gespeichert ist, sichert man die Datei sinnvollerweise mit den Befehlen

```
chown root:wheel /etc/aiccu.conf
chmod 600 /etc/aiccu.conf
```

gegen unbefugte Einsicht und Veränderungen ab. Den Testlauf startet man mit `/usr/sbin/aiccu start`. Wie unter Linux lässt sich mit `ifconfig` überprüfen, ob die IPv6-Adresse zugewiesen wurde. Allerdings heißt der virtuelle Adapter unabhängig von der Einstellung in der Konfigurationsdatei `/etc/aiccu.conf` immer `tunx`, in der Regel `tun0`, siehe [Bild 8](#).

Um AICCU beim Einschalten des Rechners automatisch zu starten, legt man die Datei `/System/Library/LaunchDaemons/net.sixxs.aiccu.plist` an, die folgenden Inhalt hat:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>Disabled</key>
  <false/>
  <key>EnableTransactions</key>
  <true/>
  <key>Label</key>
  <string>net.sixxs.aiccu</string>
  <key>OnDemand</key>
  <false/>
  <key>ProgramArguments</key>
  <array>
    <string>/usr/sbin/aiccu</string>
    <string>start</string>
  </array>
  <key>ServiceIPC</key>
  <false/>
</dict>
</plist>
```

Wer bei SixXS ein /48-Subnetz ordert, kann allen seinen Rechnern eine IPv6-Anbindung an das Internet bieten. Dabei muss nur der Router konfiguriert werden. Alle anderen Rechner im Netz werden vollautomatisch mit IPv6 versorgt.

Da die Windows-Version von AICCU den genannten Bug hat, kommt als Router nur ein Rechner mit einer Unix-Variante wie Linux oder Mac OS X in Frage. Zunächst ist AICCU unter Linux oder

Mac OS X wie beschrieben zu installieren. Sobald SixXS dem Tunnel ein Subnetz zugewiesen hat, kann man mit der Konfiguration des Routers beginnen.

Wenn AICCU gestartet ist, lassen sich alle IPv6-Routing-Möglichkeiten des Betriebssystems ausnutzen. Mit den folgenden Anleitungen für Linux und Mac OS kann man schnell einen Erfolg erzielen. Wer ausgefeiltere Routing-Optionen nutzen möchte, etwa nur das Erreichen bestimmter Rechner im Heimnetz, kommt nicht umhin, sich ausführlich mit IPv6-Routing unter Linux und Mac OS auseinanderzusetzen.

Unter Linux legt man am besten ein bash-Skript an, das man beispielsweise `/usr/local/bin/sixxs-route-start` nennt. Wenn man von SixXS das Subnetz `2001:db8:1::/48` zugewiesen bekommen hat und das zu versorgende Heimnetz am Ethernet-Interface `eth0` hängt, dann sollte das Skript wie folgt aussehen:

```
#!/bin/bash
echo 1 >/proc/sys/net/ipv6/conf/all/forwarding
ip -6 addr add 2001:db8:1::1 dev eth0
ip -6 route add 2001:db8:1::/64 dev eth0
ip -6 route add 2001:db8:1::/48 dev lo
```

Damit ist die Routerkonfiguration abgeschlossen. Der Router hat nun im internen Netz die IPv6-Adresse `2001:db8:1::1`. Es lässt aber auch jede beliebige andere Adresse verwenden, etwa `2001:db8:1::1234:5678:9ABC:DEF0`.

Damit alle Rechner im Heimnetz direkt nach dem Einschalten wissen, dass sie diesen Router benutzen können, lässt sich der Router "advertisen". Anders als bei IPv4 ist kein DHCP erforderlich. Stattdessen installiert und startet man den Router-Advertising-Daemon `radvd`, der in allen gängigen Linux-Distributionen vorhanden ist.

Die Datei `/etc/radvd.conf` sieht dabei wie folgt aus:

```
interface eth0
{
    AdvSendAdvert on;
    AdvHomeAgentFlag off;

    prefix 2001:db8:1::/64
    {
    };
};
```

`radvd` lässt sich über den Runlevel-Editor der jeweiligen Distribution für einen Autostart konfigurieren. Da mittlerweile nahezu jede Distribution ihren eigenen Init-Mechanismus verwendet, sei auf die Dokumentation der Distribution verwiesen, um herauszufinden, wie man das Skript `/usr/local/bin/sixxs-route-start` dazu bringt, beim Hochfahren des Rechners zu starten.

Für Distributionen, die ein Standard-SysV-Init nutzen oder Kompatibilität dazu bieten, kann das folgende Skript /etc/init.d/sixxs genutzt werden:

```
#!/bin/sh

if [ `id -u` != 0 ]; then
    echo „Must be root to use this script“
    exit 5
fi

case „$1“ in
    start)
        /usr/local/bin/sixxs-route-start
        ;;
    stop)
        echo 0 >/proc/sys/net/ipv6/conf/all/forwarding
        ip -6 addr del 2001:db8:1::1 dev eth0
        ip -6 route del 2001:db8:1::/64 dev eth0
        ip -6 route del 2001:db8:1::/48 dev lo
        ;;
    restart)
        $0 stop
        $0 start
        ;;
    *)
        echo „Usage: $0 {start|stop|restart}“
        exit 1
esac
exit 0
```

Mac OS X als IPv6-Router einsetzen

Als erstes weist man über die Systemeinstellungen der Netzwerkkarte die manuelle IPv6-Adresse 2100:db8:1::1 zu, sofern SixXS das Subnetz 2100:db8:1::/48 zugeteilt hat. Als Router wird ::1 und als Präfix-Länge 64 eingetragen, siehe [Bild 9](#). Danach öffnet man ein Terminal, verschafft sich mit `sudo -s` Root-Rechte und editiert die Datei `/etc/rtdvd.conf`, so dass sie folgenden Inhalt hat:

```
en0:
:addr#1:addr="2001:db8:1::":prefixlen#64:
```

Anschließend gibt nacheinander folgende Befehle ein:

```
sysctl -w net.inet6.ip6.forwarding=1
sysctl -w net.inet6.ip6.redirect=1
rtdvd -s en0
```

Danach sollten alle anderen Rechner im lokalen Netz Internetzugang via IPv6 haben.

Der Tunnelbroker-Dienst [SixXS](#) bietet heute jedermann eine IPv6-Anbindung mit festen IPv6-Adressen, und zwar auch dann, wenn der eigene DSL-Provider nur eine Anbindung mit einer dynamischen IPv4-Adresse zulässt, die täglich wechselt. Das ist ein Vorteil gegenüber einer [6to4-Anbindung](#), die zwar ohne Anmeldung von jedem DSL-Anschluss genutzt werden kann, deren IPv6-Adressen sich jedoch jedes Mal ändern, wenn der Provider eine neue IPv4-Adresse zuteilt. Mit einer Fritzbox 7270, die man mit der [aktuellen Laborfirmware](#) aufrüstet, ist diese Anbindung sehr einfach zu realisieren. Ebenso einfach kann man ein mobil genutztes Gerät wie ein Netbook oder einen Laptop mit einem IPv6-Tunnel versorgen.

Praktischen Nutzen erzielt man vor allem, wenn man seine Rechner im Heim- oder Firmennetzwerk von unterwegs ohne [VPN](#) erreichen will. Öffentliche Dienste im Internet, die nur über IPv6 angeboten werden, gibt es kaum. Zwar kann man bei einer IPv6-Anbindung die Webseite [ipv6.google.com](#) erreichen, wird jedoch schnell feststellen, dass es sich um exakt dieselbe Seite wie [www.google.com](#) handelt.

Wer seine Rechner im Heimnetz erreichbar macht, sollte sich darüber im Klaren sein, dass die angebotenen Rechner mit einer öffentlichen IPv6-Adresse im Internet stehen. Sie müssen daher gut abgesichert werden. Die aktuelle Laborfirmware der Fritzbox 7270 löst dieses Problem mit der Holzhammermethode. Sie hat eine "Zwangsfirewall" implementiert, die den Zugriff von außen auf die Rechner im Heimnetz verhindert.

Als Alternative bietet es sich an, Linux oder Mac OS X als Router zu benutzen. Aufgrund eines Bugs in der Tunnel-Client-Software AICCU kann man Windows nicht als Router einsetzen. Um das zu realisieren, ist eine gewisse Unix-Erfahrung erforderlich. So kann man beispielsweise bestimmte Rechner des Heimnetzes von außen erreichbar machen, während man andere vom Routing ausschließt.

Wem es nur darauf ankommt, eine einfache Anbindung von mobilen Geräten an das eigene Netz zu schaffen oder sich mit Freunden in einem virtuellen LAN zu vernetzen, der sollte eine Software wie [Hamachi](#) nutzen. Eine Lösung mit IPv6 bietet jedoch die ideale Gelegenheit, sich mit dem Internetprotokoll der Zukunft vertraut zu machen und sich einen Wissensvorteil zu verschaffen.