

Transparent Proxy for IPv6 traffic under Linux

Asked 15 years, 9 months ago Modified 11 years, 11 months ago

Viewed 9k times



9



When maintaining networks, it is often an expedient thing to do to run a transparent proxy. By transparent proxy I mean a proxy that 'hijacks' outgoing connections and runs them through a local service. Specifically I run a linux firewall with squid configured so that all tcp/ip connections forwarded on port 80 are proxied by squid.

This is achieved using the iptables 'nat' table, using IPv4.

But iptables for IPv6 does not have a 'nat' table, so I cannot use the same implementation. What is a technique I can use to transparently proxy traffic for IPv6 connections?

linux

ipv6

nat

Share

edited Oct 12, 2010 at 10:24

Improve this question

Follow

asked Mar 9, 2009 at 6:04



Jerub

42.6k ● 15 ● 75 ● 90

6 Answers

Sorted by:

Highest score (default)



8



A viable way to do this is with the TPROXY rule in iptables, documentation is available here:

- http://wiki.squid-cache.org/Features/Tproxy4#IPv6_Support
- <http://www.mjmwired.net/kernel/Documentation/networking/tproxy.txt>

This should be supported Squid (\geq version 3.2). Using `-enable-linux-netfilter` and the `iptables -t mangle -j TPROXY` rule.

Share Improve this answer

edited Jan 2, 2013 at 9:24

Follow

answered Sep 9, 2010 at 4:10



Jerub

42.6k ● 15 ● 75 ● 90



2

iptables has a QUEUE target, which you can use to deliver packets to userspace. I am not sure, but perhaps something could be implemented there.



Past that, you could take a stab at adding something to the kernel to do redirection.



Share Improve this answer

answered Mar 9, 2009 at 6:38

Follow



[Zoredache](#)

39.5k ● 9 ● 49 ● 62



2

You can't. Quoting from squid-cache.org:

NAT simply does not exist in IPv6. By Design.



Given that transparency/interception is actually a feature gained by secretly twisting NAT routes inside out and back on themselves. It's quite logical that a protocol without NAT cannot do transparency and interception that way.



Share Improve this answer

answered Mar 25, 2009 at 11:35

Follow



[innaM](#)

47.8k ● 5 ● 71 ● 88



2

Here's an implementation:

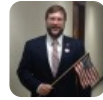
<http://www.suse.de/~krahmer/ip6nat/>



Share Improve this answer

answered Jun 24, 2009 at 21:06

Follow



[dwcarder](#)

21 ● 1



0



Another sort of ugly hack:

- MARK all traffic with iptables (seems, there is CONNMARK target for IPv6)
- route all marked traffic to tun device
- do user-space NAT in the daemon listening at tun device
- ...

Share Improve this answer

answered Mar 10, 2009 at 18:00

Follow



[darkk](#)

824 ● 7 ● 13

1 Is this actually possible? Any idea where I'd start with a tun implementation with ipv6 support? – [Jerub](#) Mar 10, 2009 at 23:00

IMHO, it SHOULD be possible. Linux tun/tap driver seems to support IPv6. Try tap driver (virtual ethernet) instead of tun if I'm wrong and IPv6 is not supported. I don't know what may be done with iptables `QUEUE`, but tun-based solution should work, though it may be unsuitable for highload. – [darkk](#) Mar 16, 2009 at 12:52



Write your own implementation of NAT in IPv6 stack.

-3

Share Improve this answer

edited Apr 9, 2012 at 11:39

Follow



Chuck Norris

15.2k ● 15 ● 95 ● 127



answered Mar 25, 2009 at 12:04



Kazimieras Aliulis

1,551 ● 3 ● 13 ● 25
