strongSwan - Issue #817

IKEv2 IPv6 Router Advertisement

11.01.2015 20:57 - Sam Wong

Status:	Feedback	
Priority:	Normal	
Assignee:		
Category:	libcharon	
Affected version:		**Resolution:**

Description

I have a working IKEv2 server setup on a Linux box 'moon' (Ubuntu 14.04), with one IPv4 address and a globally routable /48 IPv6 subnet.

On Windows 7/8 'roadwarrior', the connection can be established, and all IPv4 Internet traffic goes through 'moon' as planned. However, IPv6 is not accessible even though an virtual IP has been seen assigned on the interface correctly.

Only if I manually added a route on 'roadwarrior'

```
route -6 add ::0/0 2001:123:19:d81:1::2  # This is the Virtual IP address
```

Everythings works - and all IPv4 and v6 traffic goes through the tunnel.

I guess what's missing is the router advertisement.
I tried running radvd on the Linux but it doesn't work.
I tried kernel-libipsec such that there is a ipsec0 TUN for tcpdump/radvd to be configured with - although I can see RA solicit request packet coming from Windows, no v6 traffic seems be able to cross the tunnel though. Even IPv6 pinging from 'moon' to 'roadwarrior' does not work. (I don't know how to wiretap the 'roadwarrior' though)

My configurations:

```
$ ipsec --version
Linux strongSwan U5.1.2/K3.13.0-24-generic

$ uname -a
Linux niceboat 3.13.0-24-generic #47-Ubuntu SMP Fri May 2 23:30:00 UTC 2014 x86_64 x86_64 x86_64 GNU/Linux

config setup
    uniqueids=never

conn %default
    keyexchange=ikev2
    ike=aes256-sha1-modp1024!
    esp=aes256-sha1!
    dpdaction=clear
    dpddelay=300s

conn win7
    left=%any
    leftsubnet=0.0.0.0/0,::/0
    leftauth=pubkey
    leftcert=niceboatProdCert.der
    leftid=@niceboat.hellosam.net
    right=%any
    rightsourceip=2001:123:19:d81:1::/80,172.31.0.0/21
    rightauth=eap-radius
    eap_identity=%any
    auto=add
```

History

#1 - 17.02.2015 17:10 - Tobias Brunner

- Target version deleted (5.3.0)

#2 - 17.02.2015 17:36 - Tobias Brunner

- Tracker changed from Feature to Issue

- Status changed from New to Feedback

> Only if I manually added a route on 'roadwarrior'
> [...]

Hm, this sounds like a Windows bug to me. Shouldn't the native VPN client install such a route automatically?

> I guess what's missing is the router advertisement.

I don't really see why NDP would be needed. But have a look at this Gist, looks like at least one other user ran into this issue.

> I don't know how to wiretap the 'roadwarrior' though

Wireshark?

#3 - 07.06.2015 11:38 - Conrad Kostecki

I can confirm this Problem.
My Windows 8.1 and Windows Phone 8.1 are showing the same behaviour.
I have to add manually an IPv6 route or IPv6 won't work.

IPv4 is working perfectly fine.

Conrad

#4 - 20.10.2015 21:35 - ValdikSS ValdikSS

Here is a plugin for strongSwan by Richard Laager which adds remote traffic selector for the corresponding link local IP.
https://www.mail-archive.com/users%40lists.strongswan.org/msg09241.html

To use it, you should:
1) Apply patch and compile strongSwan with --enable-link-local-ts
2) Enable plugin in charon.conf
3) Configure any router advertisement daemon to answer router solicitation, prefix is not needed. Radvd should work.
4) Add fe80::/64 address to the ipsec interface.

I'd like to see this plugin in the mainline. It is possible?

#5 - 12.01.2016 16:50 - Kilian Krause

Any thoughts from upstream on whether or not this will be pulled into any future upstream release (or at least a feature equivalent)?

#6 - 18.11.2019 09:08 - Christian Ehrhardt

Hi,
I just got into contact to this by the Author of the initial ML thread that was already linked here [1].
They are maintaining a PPA with this added on top [2] and from a few discussions around the internet it seems to be a common enough issue that I wanted to probe for an update here again just as Kilian did 4 years ago.

Is there any particular way this should be resolved in a different way?

Or would you want/need the old submission rebased. AFAIK Richard does so anyway and could provide code that matches todays git head if that would help.

To sum it up my questions are:
- could this become one of the many plugins that strongswans already has?
- if not, what is the alternative?
- if yes then what is needed to get it done?

[1]: https://www.mail-archive.com/users%40lists.strongswan.org/msg09241.html
[2]: https://launchpad.net/~wiktel/+archive/ubuntu/ppa/+packages

#7 - 18.11.2019 10:14 - Richard Laager

As the original author of this patch, let me summarize the situation from my perspective...

At my day job, we want a VPN that uses modern IKEv2+IPsec security, is usable by the *stock* Windows VPN client, and works with *both* IPv4 and IPv6 on the *inside* of the VPN.

Windows is, IMHO, a bit weird in how it handles IPv6 inside of an IKEv2 VPN. It treats it like an Ethernet interface and expects to do a router solicitation and receive a router advertisement in response. (Additionally, at least when we tested this extensively with Windows 7, it would only send one RS at the start of the VPN session, so the server needs to send RAs *unsolicited* before they expire.)

We have had this working for several years. It requires strongswan, my patch to strongswan to add a link_local_ts plugin, radvd, and my strongswan _updown script to update the radvd.conf file. With every release of Ubuntu, I have to re-patch the strongswan package to add this super trivial plugin. I've been doing this for years now and this has been very stable across multiple Ubuntu and Windows releases. It'd be really nice if this could be included in strongswan directly.

The key ask here is for strongswan to ship the link_local_ts plugin. The updown script is a straightforward Perl hook script that, being interpreted not compiled, doesn't *have* to be shipped with strongswan (though it would be nice if it was).

Is this acceptable in principle? If so, I'll submit a refreshed version of the patch.

#8 - 19.11.2019 19:05 - Tobias Brunner

This reliance on Neighbor Discover is weird. No idea where it comes from or is specified/documented (I haven't found anything in Microsoft's open specs). It definitely isn't compliant with the IPv4-like mechanism specified in RFC 7296 for IKEv2. Section 3.15.3 explicitly states:

> In particular, IPv6 stateless autoconfiguration or router advertisement messages are not used, neither is neighbor discovery.

While that approach does have some limitations (one of which is the lack of a link-local address, which some protocols might need), it's simple and usually works fine.

There actually exists a possible alternative defined in RFC 5739 (currently not implemented by strongSwan), which assigns unique prefixes to clients and includes the link-local address based on the client's link ID in the traffic selector. But as far as I can tell, Windows doesn't send an INTERNAL_IP6_LINK attribute in IKE_AUTH. The use of ND, in particular RD, would not be in compliance with RFC 5739 anyway, as that explicitly excludes using most of ND, **because all the stuff is already negotiated with IKEv2**. That is, with neither approach is the use of Neighbor Discovery like Windows does necessary or compliant. So Windows users should probably go and complain to Microsoft about this.

To install routes that work, without having to rely on ND and any special treatment on the server, they may be configured for individual VPN connections on the client using the Add-VpnConnectionRoute PowerShell cmdlet (which might have to be used anyway for split-tunneling, at least for IPv4). To route everything via VPN use the following (::/0 is not accepted):

```
Add-VpnConnectionRoute <name> -DestinationPrefix ::/1
Add-VpnConnectionRoute <name> -DestinationPrefix 8000::/1
```

Now regarding a plugin that adds link-local addresses to the remote traffic selector. I guess we could add something like that if it helps, but it's obviously not a complete solution (and I personally prefer the client-based approach above). I pushed something to the *link-local-ts* branch. There is a potential problem with conflicting link-local addresses if multiple address pools are used concurrently (the pools must be different in the last 64 bits to avoid that). I also wonder if it might be necessary to add the link-local address(es) of the server to its traffic selector in case split tunneling via narrowing on the server is used. Learning these addresses is currently not easily possible, though, as they are not enumerable via the kernel-net implementations. We also don't know what the original traffic selectors were, we only see the narrowed ones in the listener (we can only legally add such a TS if it was contained in the proposed TS). However, since this is for Windows clients, which always propose ::/0 (any routes configured with Add-VpnConnectionRoute don't matter), this check would probably not be necessary.

#9 - 20.11.2019 18:39 - Tobias Brunner

By the way, using the PowerShell on the client might actually be necessary anyway. Because even with the latest Windows 10 release the default IKEv2 proposal still only includes *modp1024*. So most users probably want to change that using the Set-VpnConnectionIPsecConfiguration cmdlet.

#10 - 14.12.2019 05:28 - Richard Laager

Sorry for the delay. I'm not sure why I didn't get an email with your comments. I've now clicked "watch" on this, which might help.

Set-VpnConnectionIpsecConfiguration sounds like a one-time thing when the VPN is setup. That's extra hassle, but not too bad.

I'm concerned that Add-VpnConnectionRoute has to be run on every VPN turn-up, which would mean that the VPN must be turned up by a script, not the direct built-in mechanisms. I think we have one person using that, in a script, for split tunneling. Everyone else uses a default route. I'll have to do some testing to confirm.

With regards to using router discovery... I'm not looking to defend Microsoft's implementation choices. I'm just looking to interoperate with it, in its existing state.

Perhaps I'm doing this wrong. Do you know anything about Microsoft's implementation of L2TP? Is that something that could help me here?

#11 - 16.12.2019 10:32 - Tobias Brunner

> I'm concerned that Add-VpnConnectionRoute has to be run on every VPN turn-up

It doesn't, same semantics as Set-VpnConnectionIPsecConfiguration. The configured routes can be retrieved via routes property of the object returned by Get-VpnConnection and removed with Remove-VpnConnectionRoute.

> Do you know anything about Microsoft's implementation of L2TP? Is that something that could help me here?

No idea. (I only know it's limited to IKEv1, so I wouldn't recommend using it.)

#12 - 27.12.2019 03:30 - Richard Laager

I'm currently having my colleagues at work do additional testing on my proposed changes, but tentatively, we are going to go the PowerShell route. If we have to run a PowerShell script to configure (today's) strong encryption, we can easily use Add-VpnConnectionRoute at the same time. That would allow us to eliminate this patch, the custom _updown script, and radvd (plus the configuration to prevent leakage of RAs out the regular interface).

#13 - 27.03.2020 17:14 - Malcolm Scott

Richard Laager wrote:

> Windows is, IMHO, a bit weird in how it handles IPv6 inside of an IKEv2 VPN. It treats it like an Ethernet interface and expects to do a router solicitation and receive a router advertisement in response. (Additionally, at least when we tested this extensively with Windows 7, it would only send one RS at the start of the VPN session, so the server needs to send RAs *unsolicited* before they expire.)
>
> We have had this working for several years. It requires strongswan, my patch to strongswan to add a link_local_ts plugin, radvd, and my strongswan _updown script to update the radvd.conf file.

I came across this whilst trying to solve this problem myself, and have managed to make your router advertisement method work without using your link_local_ts plugin.

It turns out Windows (10) accepts unicast router advertisements directed at the virtual address of an IKEv2 interface, and doesn't need a link local address for this to work.

So, I just have an updown script like yours add the client's virtual IPv6 address to a clients{} block in /etc/radvd.conf, using stock (Ubuntu) strongSwan on the server.