strongSwan - Feature #268

support for ssh keypairs in strongswan network-manager plugin

20.12.2012 15:27 - Yves-Alexis Perez

Status:	Feedback	**Start date:**	20.12.2012
Priority:	Normal	**Due date:**	
Assignee:	Tobias Brunner	**Estimated time:**	0.00 hour
Category:			
Target version:			
Resolution:			

Description

Hi,

I'm trying to setup a strongswan client authentication using my OpenPGP smartcard. I'm currently able to use it for ssh authentication (gpg-agent being used as an ssh-agent) so I thought it would be possible to do the same using the 'agent' plugin in strongswan.

As far as I can tell, 'agent' plugin only makes sense when used with network-manager plugin since something has to pass strongswan the path to the unix socket where the agent can be reached.

So I've tried to install network-manager and the network-manager-strongswan plugin. I can configure the thing, but right now the “agent” part only supports certificates (I guess those are SSH certificates and not X509 certificates, but I'm not completely sure).

I didn't find much documentation on how to setup strongswan with SSH certificates, and I think it would be nice to have some sort of simpler setup where one could directly use ssh keypairs to authenticate (one would need to put the public part on the server, like for ssh connections, so it doesn't scale really well, but is really useful for small, home setups).

If you need any more information, please ask.

History

#1 - 20.12.2012 16:44 - Martin Willi

> so I thought it would be possible to do the same using the 'agent' plugin in strongswan.

If this card doesn't have a full PKCS#11 interface, using the agent plugin might be an option.

> As far as I can tell, 'agent' plugin only makes sense when used with network-manager plugin since something has to pass strongswan the path to the unix socket where the agent can be reached.

Yes, it is used by NM only, but extending ipsec.secrets to support agent configuration shouldn't be too hard.

> I think it would be nice to have some sort of simpler setup where one could directly use ssh keypairs to authenticate

I haven't tested it recently, but I think everything is there on the server side. Have a look at the left/rightrsasigkey options; they allow you to define a trusted RSA public key.

On the client, we'd need an extension to ipsec.secrets when using a ipsec.conf based setup. Or, when using NM, we'd have to add support for RAW keys in addition to X.509 certificates.

#2 - 20.12.2012 17:05 - Yves-Alexis Perez

Martin Willi wrote:

> > so I thought it would be possible to do the same using the 'agent' plugin in strongswan.
>
> If this card doesn't have a full PKCS#11 interface, using the agent plugin might be an option.

Yes, OpenPGP smartcard is completely unrelated to PKCS#11. There have been attempts to exhibit a PKCS#11 interface but afaiui it's a bit hackish. And in my case, I prefer to rely on PGP stuff anyway.

> > As far as I can tell, 'agent' plugin only makes sense when used with network-manager plugin since something has to pass strongswan the path to the unix socket where the agent can be reached.
>
> Yes, it is used by NM only, but extending ipsec.secrets to support agent configuration shouldn't be too hard.

But “something” running in the user session has to give strongswan the socket path, so that'll still require an user-util.

> > I think it would be nice to have some sort of simpler setup where one could directly use ssh keypairs to authenticate
>
> I haven't tested it recently, but I think everything is there on the server side. Have a look at the left/rightrsasigkey options; they allow you to define a trusted RSA public key.

Even better would be to accept an ssh pubkey format, but I guess some of the format exported by ssh-keygen should work?

> On the client, we'd need an extension to ipsec.secrets when using a ipsec.conf based setup. Or, when using NM, we'd have to add support for RAW keys in addition to X.509 certificates.

I guess so.

#3 - 25.07.2013 10:27 - Tobias Brunner

- Status changed from New to Feedback

- Assignee set to Tobias Brunner

The 5.1.0 release will bring some of the requested features. For instance, SSH public keys can be used to authenticate clients. Also, the charon-cmd VPN client supports user authentication via *ssh-agent*.

#4 - 25.07.2013 22:34 - Yves-Alexis Perez

Thanks for the update, will try to test it but I'll be away for the first half of august so it might take some time. And I'll need to check if it's possible to feed charon/charon-cmd the socket path for ssh-agent.

#5 - 26.08.2013 13:39 - Yves-Alexis Perez

Ok so I've updated to strongSwan 5.1, and the sshkey plugin has been built. Is there some documentation on how to plug everything together?

#6 - 30.09.2013 23:49 - Yves-Alexis Perez

Yves-Alexis Perez wrote:

> Ok so I've updated to strongSwan 5.1, and the sshkey plugin has been built. Is there some documentation on how to plug everything together?

Actually I'm unsure what to use as identity when using pubkey authentication, and especially how to give the pubkey to the charon responder. Any pointer?

#7 - 02.10.2013 12:15 - Tobias Brunner

The identity can be chosen arbitrarily (but obviously has to be the same on both ends). And the public key on the responder can be configured via *rightsigkey* in ipsec.conf.

For SSH public keys you could copy-n-paste the base64 portion from the id_rsa.pub file like this:

```
rightsigkey=ssh:0sAAAAB3NzaC1yc2EA...
```

The *sshkey* plugin in the current master branch also supports rightsigkey=/path/to/id_rsa.pub.

#8 - 19.11.2013 21:40 - Yves-Alexis Perez

I've finally had the time to test with to 5.1 charon. Problem is that, on the client, I'd like to use my ssh-agent. If running charon-cmd as root, it'll drop its privileges (and only keep CAP_NET_ADMIN for network setup) and thus won't be able to connect() to the ssh-agent socket.

I'm not too sure what to do here, maybe it could connect() the socket before dropping caps (although I'm not completely sure of all the consequences).

I guess the most sensible thing to do would be to use some kind of wrapper like the network-manager plugin or something like that, but it open a whole new kind of worm which I'm not sure I'm prepared to face.

#9 - 19.02.2014 15:13 - Yves-Alexis Perez

So, the charon-cmd part is working fine (provided we use sudo -E to keep $SSH_AUTH_SOCK variable and charon-cmd doesn't drop CAP_DAC_OVERRIDE to access it).

There's still an issue for Network Manager setup.

As far as I can tell, the current version(1.3.0) can only use the following authentication methods:

- certificate/private key (using a certificate file and a private key file)
- certificate/ssh-agent (using a certificate file and the private key in ssh agent?)
- smartcard (using pkcs11)
- EAP

certificate/ssh-agent looks like what I'm looking for, except that in case of ssh-agent, I don't have any certificate associated to the public key. On the responder side, the public key is configured using rightsigkey=ssh:0sAAAA (until I can use the sshkey plugin with support for a path).

Would it be possible to add to the plugin a pure public key authentication, a bit like charon-cmd when using --agent?

(I can open a separate issue if you want, but as the initial title of this one is exactly about that, I thought we could keep it, even though the body is more about charon-cmd).