

Re: [Full-Disclosure] SSH vs. TLS

Source: <http://www.derkeiler.com/Mailing-Lists/Full-Disclosure/2004-06/1007.html>

Valdis.Kletnieks_at_vt.edu

Date: 06/29/04

To: dante@forethought.net

Date: Tue, 29 Jun 2004 13:35:45 -0400

On Tue, 29 Jun 2004 09:20:11 MDT, dante@forethought.net said:

- > - *SSH is not an IETF standard.*
- >
- > *The documents that make up the SSH2 protocol are still at the*
- > *Internet-Draft stage. I don't know how long they've been at this stage,*
- > *but the comment from security was that it's been at this stage for a while*
- > *and doesn't appear to be moving forward.*

Hmm... Which RFC specifies *TLS* Telnet? I'm drawing a blank on it at the current http://www.ietf.org/iesg/1rfc_index.txt.

DES, 3DES, IDEA-128 yes Kerberos yes. TLS Telnet? Umm... I'm drawing a blank here, even as an I-D...

Oh wait... RFC2839 "Internet Kermit Service" discusses it. But that's an 'informational'....

The various revisions of draft-ietf-tn3270e-telnet-tls-NN.txt all expired, apparently without making it to RFC status. Which is better, an active I-D, or an expired one? ;)

- > - *SSH allows tunneling other protocols, circumventing firewall policies.*
- >
- > *While most admins consider this to be a desirable feature, it's generally*
- > *frowned upon by network ops and security. Port forwarding can be turned*
- > *off within the sshd config file. However, the security group has no way of*
- > *making sure it stays off. Essentially, it's a trust issue: Can the*
- > *security group trust the admin group not to turn it back on?*

RFC2840 seems to specify kermit-over-telnet... sounds like tunneling to me.

And if your security group doesn't have a way to *audit* these systems to make sure the admin group is following the policies, you have a *MUCH* *MUCH* *MUCH* *BIGGER* problem than ssh-vs-telnet.

Full-Disclosure: Re: [Full-Disclosure] SSH vs. TLS

And there's a bad case of "pot", "kettle", and "black" here – if SSH is evil because it can tunnel stuff, this guy is recommending the use of TLS *to tunnel telnet* traffic? (Hint – the only reason tunneling is "evil" is because you can't have a firewall examining the traffic – which is *just* as true for tls-telnet....)..

- > *In addition, there are several requirements for key management. I think*
- > *kerberos will address all of these, maybe I'm wrong.*
- >
- > *– There must be a secure means by which all server keys are distributed to*
- > *appropriate ssh clients.*

Also true for TLS (hint – what happens if you hack a server and replace that cert signed by your CA, or intercept/replace it en route?)

- > *– There must be a secure means by which all necessary client keys are*
- > *distributed to appropriate servers.*

Also true for TLS. (hint – why does ALMOST NOBODY use certificate-based web browser client authentication? ;)

- > *– There must be a mechanism to expire keys. Keys will not be valid for*
- > *more than 365 days. This feature should be an integral part of the key*
- > *management infrastructure. It must technically prevent either clients or*
- > *servers from using expired keys.*

```
echo >> /var/spool/cron/crontabs/root
0 * * * * find /etc/ssh -name 'ssh_host*key' -mtime +365 -exec rm { } \; && /etc/init.d/sshd restart
```

That should suffice. ;)

(Also – notice that you now have to balance "exposure due to old key" against "exposure due to all users having to get the new keys". Personally, I suspect that the only sites that *really* have to worry about a year-old host key are the kind of places that use pressurized conduits with alarm circuits...)

- > *– There must be a mechanism in place to allow a trusted third party to*
- > *revoke either a client or a server key. Revocation must technically*
- > *prevent either clients or servers from using revoked keys.*

Quick – does the TLS his Telnet uses support *PROPER* handling of a CRL? If not, he's blowing smoke...

- > *– There must be a mechanism to integrate both client and server keys into*
- > *LDAP.*

This may be a good thing from a data-management perspective, but I'm totally failing to see why it's a Good Thing for security..

- > *So, what do you all think? Is SSH really that bad or are these*
- > *requirements unreasonable? Is it really worth implementing TLS Telnet?*

Re: [Full-Disclosure] SSH vs. TLS

Full-Disclosure: Re: [Full-Disclosure] SSH vs. TLS

Well... you have your options:

<http://www.openssh.org/>

<http://www.openssh.org/users.html>

<http://www.openssh.org/windows.html>

<http://www.openssh.org/macos.html> (OSX includes OpenSSH by default)

Remember – "actually deployable to end users" almost always trumps any "my security appendage looks more like a white ivory tower than yours" analysis.

Lots of different clients. He's welcome to come up with that extensive a list of well-tested clients to cover the entire range of systems for TLS Telnet...(googling for '+tls +telnet' gets some 43,000 hits, while googling for openssh gets 954,000.).

He's then invited to read Schneier, on any of the *many* places where he points out that a crypto system that's heavily beat on is more trustable than one that's not heavily tested. Yes, OpenSSH has had its share of bugs. ON the other hand, by now that code is *REALLY* heavily audited (quite possibly some of the best-audited code in existence ;), while his TLS-Telnet client probably still has tons of low-hanging-fruit bugs left in it.....

Bottom line – whatever theoretical advantages TLS-Telnet may have over OpenSSH are vastly outweighed by the fact that the current OpenSSH software is widely deployed, usable, and well-tested....

Full-Disclosure – We believe in it.

Charter: <http://lists.netsys.com/full-disclosure-charter.html>

- application/pgp-signature attachment: [stored](#)