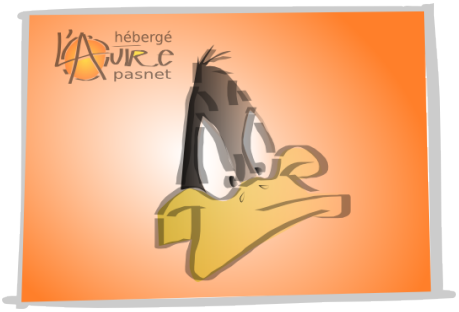
http://www.coincoin.fr.eu.org/?There-is-a-surprising-amount-of



There is a surprising amount of old SSL code in running MTAs

- 6- Webographie -

Date de mise en ligne : jeudi 24 décembre 2015

SSL has a number of old, dark corners. One of them is 'SSL2 handshake compatibility' , where clients that want to be able to talk to a SSLv2 server must start their conversation out with a (special) SSLv2 ClientHello. Of course this means that a SSL v3 or TLS server that wants to be able to talk to such clients needs to accept the SSLv2 ClientHello and then upgrade the connection to SSLv3 or TLS afterwards, in a coordinated dance with the client. Now, SSLv2 has been dead for a long time now , but of course SSL libraries don't like removing features so SSLv2 client support lingered on for years and years after SSLv2 servers had probably all vanished, and along with it lingered SSL2 handshake compatibility. All of this is generally invisible in server logs, which will just reflect that SSLv3 or more likely TLS was negotiated in the end.

Except if you run a Go-based TLS server, that is, because Go doesn't support SSL2 handshake compatibility . As Filippo Valsorda notes, the browsers and browser software that supports this is now really, depressingly old and most people won't care about them anyways. Unfortunately, HTTPS is not the only TLS based protocol in the world and browsers (and browser libraries) are not the only things making TLS connections. In particular, SMTP supports TLS, which means that s thus talk TLS.

I have a Go based SMTP server with TLS support and it logs TLS setup errors, so I actually have an opportunity to see how many sending MTAs try to do SSL2 handshake compatibility. It turns out that there were more than I expected, although the good news is that the number of such servers has fallen over time and that some of the really eyebrow raising parties seem to have stopped doing this.

(When I started my sinkhole SMTP server in the spring and summer of 2014, some of Yahoo's mail servers were still doing SSL2 handshake compatibility for outgoing SSL connections. That seems to have changed since then.)

I've done various casual checks on the machines trying to do this with my server, and they've turned up a wide variety of apparent sending MTAs, many of them running on Unix machines (unless someone has ported eg qmail to Windows). At least some of them are running MTA code that seems very old by now (eg), although this doesn't say anything definitive about how old their SSL libraries are.

At one level most of this is not really surprising. We've always known that there are very old servers out there and old servers mean old SSL libraries with old behavior, which this very definitely is. On another level I am surprised by just how common this seems to be. My little SMTP sinkhole is not really a hotspot of activity, yet practically the day I recently started logging these messages again I was getting some sending MTAs that were trying to do this.

Unfortunately this also means that if you care about maximal SSL support for MTAs and similar server side software, you probably want to consider still supporting SSL2 handshake compatibility. Users clearly update browsers, IMAP clients, and so on far more frequently than people update server side software like MTAs and OS SSL libraries (and OSes, for that matter).

(I also suspect that visible programs like browsers, IMAP clients, and so on are somewhat more likely to explicitly tell their SSL libraries to turn off now obsolete features like SSL2 support, SSL2 handshake compatibility, and so on. I expect that much MTA code just goes with the SSL library defaults, which means that if the library is conservative about what it deprecates the MTA goes along with that.)