Authentication

Authentication is a big part of TLS. But authentication is a slippery concept. In some contexts it is very complicated, in other contexts it is extremely simple.

There is a continuum in terms of what we know about another party. We may know absolutely nothing at all, as for example when we visit a web site for the very first time. In this case it is entirely appropriate to go to extreme lengths to establish that we can trust the other party, to the extent that we may trust them with information like our credit card details. And this is exactly what TLS does for us, by its use of X509 based PKI authentication.

At first glance all that PKI authentication requires is the possession of a certificate chain that is anchored in a trusted certificate authority. But there is of course more to it than that. For example the certificate may have been revoked. So basic PKI has been bolstered in the context of TLS by other mechanisms like certificate stapling, and certificate transparency. These important mechanisms effectively establish and reinforce our trust in this most challenging of settings.

PAKEs

Password authenticated key exchange is one of the wonders of modern cryptography. Two parties can authenticate, not by sharing a massive secret that cannot be brute-forced, but by sharing a simple 4-digit PIN number. Unless an attacker is able to guess the correct PIN first time, the two party connection remains secure and fully mutually authenticated. Which suggests that maybe authentication should not be such a big deal.

What PAKEs teach us is that once we share even a small amount of mutual context with the other party, our authentication requirements diminish surprisingly rapidly.

And what we can take away from this is that the most stringent requirements of first-contact authentication need not apply to subsequent communications. Trust builds rapidly. We don’t need to react to the other party like we have no information about them, that we don’t have a shared experience that we can leverage on a second and subsequent contact.

The well known Signal protocol with its famous double ratchet makes full use of this. Once first contact is made, and trust established, the mutual shared context can be used from then on to maintain trust indefinitely.

Internet of Things

In the IoT setting there is, we contend, almost always already some built-in shared context right from the start. Which is why many vendors insist on support for raw public keys in TLS. Now on the face of it raw public keys completely undermine PKI, and purists probably weep at the thought of it. But raw keys are entirely appropriate in a setting where all parties may share a common manufacturing ancestry.

Therefore in TiigerTLS, while supporting x509 authentication, we don’t go to extremes with it. We do just enough to be compatible with internet servers that support TLS1.3. As we have shown mechanisms other than full blown X509 PKI are possible (like IBE, raw keys, resumption tickets) and are particularly appropriate in the IoT setting.