TLS, November 5, 2013
Chairs: Eric Rescorla
Joseph Salowey
Scribe: Ted Hardie

Note Well reviewed.

Agenda reviewed (slides: http://www.ietf.org/proceedings/88/slides/slides-88-tls-3.pdf)
---
1. Administrivia (5 min)

Blue Sheets distributed
Jabber scribe and note takers assigned.

2. Document Status and TLS related work (10 Min)
The chairs reviewed the work going on in other groups:
- DICE
- HTTPbis & ALPN
- Apps Area TLS BCP

3. ALPN (15 Min)
- draft-ietf-tls-applayerprotoneg

Steve Friedl reviewed the discussion from the HTTPBis working group, which raised no issues. Notes that an issue has since been raised on the list. Brian Smith has raised an issue related to f5 and BigIP. Andre Popov notes their older firmware cannot take client hellos over a certain size. This is not ALPN specific, but to any large extension. This has to get resolved in any case; it is a question of rolling them out. So it is a deployment issue.

Sean asked whether 3 per cent of web servers is a really big deal (hum yes). Can this be delayed while we wait for this get resolved? Speaking as Chair, Eric asked for more data, as it would be very, very helpful. Wan-teh notes that he has a co-worker has collected a set of hostnames. These are big sites, using F5 hardware. Will Chan notes that when Adam ran those tests, there was a very long tail of sites who might be affected. Sean Turner asked whether other application protocols have been tested, e.g. SMTP. Andre says that other protocols can be added to the registry when it is set up, and then SMTP can be added then. Paul Hoffman notes that the negative effect seen here would not happen when NPN was used, and he feels that should be part of the discussion. He is hearing as well that the firmware push is going very slowly. Brian Smith from Mozilla got up to clarify: there is 255 byte issue with F5 and some other software (some Sun software). Can we rely on the patches getting out in a timely fashion? There is no reason to be optimistic about that; as a web browser maker, he feels he has a hard limit of 255 bytes. Ekr's recent draft adds more to the client hello; if we do npn and that work, we might be under 255 bytes, but that won't be the case if we do alpn and

one round trip handshake. He personally feels the benefit of alpn is not enough to justify that; in fact, he sees alpn as worse. He feels that the theme of today is “how to encrypt more”, but switching from npn to alpn puts more in plain text--which feels counter to our aims. Chairs cut the line. Andre answers the question: length of the client hello: 1.3 is a different issue, progress may involve additional issue in the client hello, and it may raise the same issue. So the F5 issue just has to be resolved, so we can move forward. It’s also not a “how much do we encrypt” versus “how much security are we getting”, and that the application layer identifier is not the most critical piece. In the next rev, we’re going to tackle that systematically. Hasan Khalil notes that no one is saying that any of these is going to stop the deployment of 1.3; we can choose today either alpn or client hello for the time period until these upgrades go forward--for the short term, we can’t do both. He then asks what kind of measurements do you want? Eric: I will talk to you offline/send to the list. Patrick McManus would like to deploy 2.0 in part because it will increase the amount of TLS in the web. There are a whole slew of tokens which will get used--you can expect the frame to be dozens of bytes long because of the semantics people want to jam in here. We do not want this to get turned off for long hostnames in order to deal with this. Sees this as silly with NPN. Scott from F5 notes that they are aware of the issue and are doing all they can; at the mercy of the customers and in some cases by those customers’ procedures for orderly upgrade. Martin Thomson: when we put this together we weren’t aware of this constraint, so we used generously sized strings. I want to know how close to the wind are we sailing? How much space do we have left with a 25 character hostname? Could we use stopgap strings that are very short? Don’t want to overblow the problem. Jon Mattson: same comment: why not make the identifiers shorter? It would not effect the protocol, only the wireshark imp.

Sean Turner: I was about to issue an IETF last call two minutes before the email came in. Now I am not comfortable doing so without more data. Breaking 3 per cent of the internet is bad. Eric: We will get together with folks and work out a plan. Do I understand from the HTTPbis chair that a plan before London would be within time? Mark: Yes, and even that isn’t a hard deadline.

Action Item: TLS chairs will work out a plan for moving forward.

4. TLS BCP (15 min)
- draft-sheffer-tls-bcp (slides: http://www.ietf.org/proceedings/88/slides/slides-88-tls-0.pdf)

Presented by Paul Hoffman, who is not an author and doesn’t really agree with the draft. He describes the motivation, which is “a good solid set of suggestions”, not mandatory requirements. Recommends a single ciphersuite or possibly two that you should accept as a server and propose as a client, certificate size, disabling fallback, etc. As people adopt the BCP, you get better interoperability and security.

Default is TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256. Requires 1.2. Eric notes that he and Joe just sent a mail saying that they are not comfortable with selecting a specific curve

within the working group. We will get with the AD on how to develop an IETF consensus on this.

Joe notes that there is a different doc on TLS for applications, what coordination is needed? Paul: A lot. Two sorts are needed: how to get to TLS-protected, and how to determine when the TLS community has a general agreement on a set of practices.

The jabber scribe asks about some of the recommendations which are for future practices. Paul says that's true, and a common category issue, may mean a change of status. Franck Martin wonders about the relationship of TLS document and particular SMTP issues. Paul wants to stress that belongs elsewhere. He also suggests that the document avoid and possibly recommend against language like "weak", since that is changed over time.

Mark Nottingham, suffering from severe jet lag, and he's raising a devil's advocate question: what kind of opportunity cost versus shipping 1.3? HTTPbis wants to reference most advanced form of TLS available. Eric says "let's find out". Paul notes that it is better to get 1.3 right. Eric sees that they are parallel discussions. Stephen Farrell, it would be a stupid idea to do a best future practice. Why would you do it? Paul answers, and there is a digression into the meaning of BCP. Stephen wants the goal to be to describe the best stuff you can turn on today. Sean--don't worry about BCP or standards track, I can make it work either way, don't worry about it. We also want to get aligned--we don't want server vendors to do it and browsers not. Andre Popov notes that security is a very agile area, and an RFC may not be the best way to maintain a document of this type. One TLS version and one ciphersuite may not be best. Wan-Teh notes that this document has very similar goals to a NIST document (cited). Ekr thinks this is an interest effort, but has similar concerns to Andre's.

The chairs take a hum on adoption, wait for a revision, don't know. No clarity between first and second, but no general opposition to adoption.

Decision: get another revision before adopting.

5. Updating Cipher Model (20 min)
- Encrypt-Then-Mac - draft-gutmann-tls-encrypt-then-mac
- AEAD

Joe summarises the Encrypt-then-MAC approach and notes questions: fallback and MAC length. He then summarized AEAD, noting two issues: previous version support, and plaintext length for MAC computation.

Mike St. Johns, notes that associated data occurs two places, which makes it hard for hardware implementations, please steer them toward a single place. Andre says that best approach is to deprecate cipher suites that have these problems; changing the way cipher suites work needs extensive analysis. Brian Smith, notes that they have additional work about adding new suites using things like cha-cha; do we have a budget for adding ciphersuites that web browsers are going to use. Are the new suites which are not CBC mode a better use of our time? Eric notes

Brian wrote a document of ciphersuites and an order; Brian says yes, he did, partly based on Google work. **Eric asks him to cut and paste it into a draft.**

Questions from the Chair: who thinks we should do only encrypt then mac; who thinks we should do AEAD, who thinks both? No one for the first? weak hum for the second? weak hum for the third? (nothing?) weak hum? Sean notes we have to do something here because it is a real attack--if we don't fix this what are we here for? **The chairs judgement is that there is more enthusiasm for AEAD, so they will solicit an actual draft, and then ask for adoption.**

6. Stream Ciphers (20 min)
- Salsa - draft-josefsson-salsa20-tls
- ChaCHa - draft-agl-tls-chacha20poly1305

Wan-teh presents on ChaCha20 stream cipher; please see the slides: http://www.ietf.org/proceedings/88/slides/slides-88-tls-1.pdf . Chairs remind people in the WG that there has previously been a presentation on Salsa; they asked for a report. They asked the CFRG, and they preferred ChaCha-20 over Salsa. Poly1305 was discussed, but there was no strong reason to shift. Mike St. Johns asked if they had numbers against CCM instead of GCM? No. Richard Barnes asked about the conflation of the sequence number and the nonce. Have you thought through this argument? Wan-teh replied that it is not required in AES to use the sequence number as the nonce, but it is allowed. The reason to do it is to assure yourself is that the nonce is not repeated. Andre notes that these are relatively new ciphers, so he has some general concerns. On the performance aspects, he feels the hardware support for AES makes the performance story weak. Eric Rescorla notes that he's happy to see something other than AES, and it sounds like the CFRG would prefer they adopt this instead of salsa. So he favors adopting this draft. He asks Wan-teh, should we specifying GCM cipher suites in addition? Wan-teh says maybe there could be work on a truncated GCM.

Kevin again; he says that the man years of research that went into AES are much better than Salsa or 1305, but it has had years of work trying to scratch its paint and nothing has so far. Kevin notes that it is his job to keep his fellow mathematicians employed. Robin Wilton asks if we can draw any inferences on its relationship to SHA3? Kevin answers that the relationship is present, but that the direct work is more salient.

A hum on adoption: **strong hum for adoption, no opposition.**

7. Hardware Considerations for TLS Key Generation (5 Min)
(slides: http://www.ietf.org/proceedings/88/slides/slides-88-tls-2.pptx)
Threat model: Insider has access to HSM token credentials, can reuse the keys in the token, but can't necessarily extract them. This is about preventing back door extraction of TLS session keys.

Dan Harkins asks if the length is mixed in with the key derivation, doesn't that solve the problem

you pose? Eric replies that the key stays in the tamper boundary. Eric: can we trust you to pay attention to the ongoing work and give us comments? Mike replies that it wasn't something he was following until he tried to do it in hardware. **Eric asks if it something we should fix in 1.3? Strong hum for yes.**

8. TLS 1.3 (60 min)
(slides: http://www.ietf.org/proceedings/88/slides/slides-88-tls-4.pdf)

Eric Rescorla apologizes for the delay in posting the material. During the presentation of the objectives, Paul Hoffman notes that most of the discussion in the past week has been about crypto changes. He asks, should there be a bullet on that? Eric agrees that there should be a new crypto bullet.

After reviewing fast track and its antecedents, Eric notes that he is not asking for adoption of it per se, but he believes that this is the idiom that the working group should adopt for 1.3. No issues on this were raised when he asked for questions.

On the reduced RT handshake with privacy topic, Joe asks whether you might also want to provide privacy for the client side of the exchange? Eric replies that this is the best you can do without snapstart, and it does leave the ClientHello in the clear. Martin Thomson asks about authentication. Eric replies that there is a tradeoff here between how fast things can get and how private you can get; TLS has a massive number of use modes, and we can't change them too much. In the reduced RT handshake with privacy, Eric notes that in the fallback case, the client offers to do X, the server says you have to do Y and it hands him new keying for Y. But the situation creates up to three clientHellos. Paul Hoffman asks him to compare this with the previous slide--essentially, this is what happens when the client gets this wrong? Yes, this is the fallback, when the initial client guess is wrong. Some of this, though, is because of the requirement for privacy for some aspects of the data in the ClientHello. Eric's discussion with Will Chan and others convinces him that some people do want this. Gabriel Montenegro: there will be a lot of incentives to get one round trip, and that means guessing right; how does that mesh with the need for agility? Eric responds that the data from previous servers creates a general histogram of what's generally acceptable. Joe notes that there can be other channels for this data. Eric agrees. Joe asks about the cached info structure; Eric describes the splitting up of the data, so you can keep some of it cache. Paul notes that he prefers this to previous approaches, as the failure protects stuff from the server to the client that previous approaches did not. Linus asks whether you could send application data before seeing a finished from the server? Cryptographically, this is sound, but he would like to see some analysis with it, and there may be interoperability problem where intermediaries are present.

Eric notes that any Zero RT handshake is a replay from the server's perspective; the server has to keep some memory of the situation. This also does not work with PFS, as you need a stable Diffie-Hellman key. Eric asks how many fallback options we should have? The potential is 0RTT resumed→ 0RTT non-resumed → 1RTT Fast Track → Full handshake; this seems pretty

complicated. Will Chan asks whether he has thought about the interaction with TCP fast open. Eric agrees that this is worth considering.

Resumption doesn't provide PFS; resumed-handshake also doesn't provide it, because you need a static key. You can use a temporary security window (hourly key?) here, for some mixed state. You can also do a rehandshake. There is also a handwaving possibility of a two-phased handshake, but it seems very complicated and fragile. Paul notes that PFS may be very important to either end.

Eric notes again that this is a quick review of the handshake possibilities and he will circle back with implementers as soon as possible.

Marc asks that if he has user that does the initial TLS session, then a follow-on user of the same browser, will he get the same credentials? The browser has to do something about that--profiles with different buckets of session credentials.

Wes is torn, he is looking at all of these variations and it conflicts with his desire to deploy the new shiny stuff. He encourages everyone to review savagely and early. Joe asks if he has thought about an upgrade path to 1.3 from 1.2? Yes, there is text in the draft. Stuff it all in extension, and keep it small so they don't choke. They can also change configurations to prefer 1.3. Yoav Nir notes that the only reason people really renegotiate is to move from server authenticated to mutually authenticated. He also notes that what we mean by PFS and what IPSEC means by it are different. Here what is in danger is the long term private key.

Brian Smith highlights that there are so many features here that it will be a long time before we can deploy them all. Can we do a cost/benefit trade-off here? So we have a realistic view of what can get done in small time. Eric says NSS team can work on then--Brian agrees, but wants to use the WG mailing list to communicate with a larger community. Sean Turner believes that is a great idea. He wants early, but constructive criticism--let's get alternative text and concrete proposals. Peter St. Andre notes we need to get clearer on what the trade-offs are. We don't know how to weigh those yet and we need to work out. Question from the floor about the "finished" message meaning--when you receive that you verify that everything has matched up. What happens if the client starts sending application data before that? Are you enabling MiTM? Server's AlmostFinished is designed to avoid that (reference to "Reduced handshake with privacy") for the data from the client; the Server also can't send before the {Finished} message. Stephen Farrell agrees that implementers are important, but we shouldn't be driven by that.

Does anyone want to stop support RSA? Chorus of No. Brian sees a lot of appeal in doing as much as we can to encourage ephemeral key exchange. Stephen confirms that we're not talking about the algorithm, but no RSA key exchange. **No consensus on this point.**