LAMPS Virtual Interim on 20 April 2022

Chairs: Russ Housley and Tim Hollebeek

Notes by: Jonathan Hammell

Chair Introduction

Slides

Chairs reminded the participants about the NOTE WELL.

The focus of this virtual interim is PQC in Certificates and S/MIME.

Proposed agenda:

```
1)  draft-turner-lamps-nist-pqc-kem-certificates (Sean, Panos, Jake, Bas)
2)  draft-perret-prat-lamps-cms-pq-kem (Ludovic, Julien, Mike)
3)  Hybrid Composite Certificates (Mike, Max, John, Serge)
    --  draft-ounsworth-pq-composite-encryption
    --  draft-ounsworth-pq-composite-keys
    --  draft-ounsworth-pq-composite-sigs
    --  draft-ounsworth-pq-explicit-composite-keys
4)  draft-becker-guthrie-cert-binding-for-multi-auth (Alison, Rebecca, Mike)
5)  draft-becker-guthrie-noncomposite-hybrid-auth (Alison, Rebecca, Mike)
```

Agenda bash: The first two items on the agenda were swapped.

draft-perret-prat-lamps-cms-pq-kem

Slides

Uri Blumenthal: The draft seems to have a lot of overhead. Can use XOR instead of AES-KEYWRAP. Will put together a slide to explain.

Mike Ounsworth: A XOR wrap likely still requires a KDF since it needs a whitened key.

Uri: The NIST Round 3 candidate PQ KEM algorithms already incorporate a KDF.

Mike O: Most PQ KEMs likely generate properly whitened keys, but other KEMs might not.

Uri: An external KDF could be used whenever the quality of the whitened key output is in doubt.

Julien Prat: We cannot assume the shared secret size provided by the KEM will be proper, so the external KDF will align it as necessary.

Quynh Dang: The NIST candidate PQ KEMs generate a 256-bit shared secret. This is sufficient for AES encryption. The recipient's KEM private key will be sufficient to generate the shared secret to perform decryption. Use of AES-128 could be done by truncating the shared secret.

Mike O: Is the j-invariant with SIKE sufficient?

Quynh: SIKE is not a NIST Round 3 candidate, so my memory of the details has faded.

Julien: To encrypt with CMS for multiple recipients, we have to be able to communicate a fixed content-encryption key (CEK). A KEM generates a unique shared-secret for each recipient. This draft explains how to handle this situation.

Uri: Slide describes the XOR instead or AES-KEYWRAP. The sender randomly-generates CEK, then XOR it with the KEM-derived shared-secret for each recipient to produce the wrapped key. The KEM ciphertext and the wrapped CEK are sent to the recipient.

Mike O: This assumes that the shared secret and the CEK are the same length.

Quynh: Looks okay, but need more time to study.

Uri: I hope that the final draft will provide "identity wrap", as I've proposed. This is not a new idea. It has been extensively analyzed.

Tim H: I agree with Uri that this is fine in a perfect world. However, when used with a poorly defined protocol, it may have issues.

Uri: From a cryptographic point of view, it doesn't matter what type of wrap you use if your KEM is secure and the CEK is random. Doing something more complex will not add any security.

Mike O: May have concerns about protocol composition attacks.

Quynh: XOR is losing half the security.

Uri: I disagree.

Russ H: We need to discuss this further on the mailing list.

draft-turner-lamps-nist-pqc-kem-certificates

Slides

Mike O: On key usages, Sean may have made some opinionated decisions.

Uri: I am uncomfortable with the NIST PQC API; it practically requires KEM as opposed to Diffie-Hellman type of Key Agreement. I hope that after Round 3, NIST might consider standardizing something like SIDH/CSIDH, even if only for ephemeral. We should allow key agreement, even if NIST selects a PQ key agreement algorithm in the future (after Round 3).

Uri: I am uncomfortable with the NIST process being the only way of choosing PQ algorithms. We should allow key agreement, even if NIST does not select a PQ key agreement algorithm.

Russ H: Our charter says that we will define structures using evaluated algorithms. The NIST process is one way, and the CFRG is another way.

Russ H: We will wait to adopt until NIST selects algorithms, which is expected very soon.

Hybrid Composite Certificates

Slides

Composite Keys

Russ H: How is a custom combiner different than replacing the top-level OID? For the receiver, it would be better to recognize incompatibility at the top-level instead of finding an unknown OID in a parameter.

Mike O: Likely the same.

Mike Jenkins: It seems odd to specify the policy in signer's key structure, it should be recipient policy that determines the acceptance.

Mike O: It is a good question who decides on the policy for signatures. This proposal is driven by the CMS encryption case.

Panos Kampanakis: Adding these combiners, such as K of N, adds a lot of complexity to the public key and signature. Please don't do that.

Mike O: Happy to modify however the community decides. Please start a mailing list thread on this topic.

Uri: Signature verification is completely in the hands of the verifier. I agree with others it is not the signer who should dictate the policy.

Tadahiko Ito: Policy may change over time. It may be difficult to specify K of N at time of signing.

Mike O: Let's discuss further on the mailing list. There are many considerations that the authors have discussed, but would be open to other input.

Jan Klaussner: As a signer, I want to know under what situations my signature is valid, which may be a reason to include it here.

Uri: What if the recipient doesn't care about one of the signatures provided in K of N combiner?

Jan: Similar to not trusting a signature in current model.

Uri: In 2 of 3, if the verifier doesn't trust one algorithm and one is broken, what should the verifier do?

Mike O: A verifier that doesn't trust the algorithms the signer has chosen is outside the scope of this document.

Scott Fluhrer: I don't see the point of omitting a signature in verification because we don't trust it anymore. It is more of a operational issue.

Sebastian Vogt: How would we put these three signatures in a certificate?

Mike O: That is defined in a different I-D.

Florence D: This adds an awful lot of complexity, which could lead to implementation errors. Tying it to the public key makes it hard to understand what the security. It would be easier to update the certificate than change the verifiers.

Jan: Could be moved to the signature I-D. Let's discuss on the mailing list.

Yoav Nir: I agree with the concerns about complexity. You just need on good algorithm. K of N doesn't care whether one is classical and one is post-quantum.

Mike O: Sounds like for signatures, it doesn't make sense. For encryption, it maybe shouldn't be in the public key structure.

Serge Mister: There may be a requirement for the signer-defined policy when enforcing non-repudiation.

Mike O: Is that an argument for including in the public key or can it just be in the signature?

Serge: The CA can include the policy in the public key, but cannot enforce the signer's actions.

Composite KEM

Russ H: Synchronizing with TLS WG hybrid KEMs is interesting. Doing so will ensure that we get more eyes on this.

Terminology

Florence D: Plan to get something out on terminology before next IETF meeting. No particular opinion on direction since all options are overloaded. Anyone with opinions, please get it contact.

Ludovic Perret: I agree that "hybrid" is overloaded, but it is now widely used and may be too late to change.

Florence D: It is fine if we've decided that "hybrid" is the best choice. But we need to write it down to stop debating it.

draft-becker-guthrie-cert-binding-for-multi-auth and draft-becker-guthrie-noncomposite-hybrid-auth

(no slides)

Rebecca Guthrie: Non-composite is an alternative to composite solutions. Currently organized into a technical draft and a non-technical informational draft.

Mike O: This work is good. It is complimentary to the composite solution. Sent a big list of comments on the mailing list yesterday.

Russ: The drafts were posted during IETF 113 meeting. I encourage people to take a look.

Uri: Support this work. The use of composite should be a policy decision.

Mike O: Non-composite (aka multi-cert) is fine and needs to be defined. The binding mechanism has value. But there are a lot of edge cases in the binding; see comments from Ryan Sleevi and myself. Need to remove the sharp edges.

Uri: Absolutely support non-composite, but not sure how much value binding provides. The verifier should determine whether to treat them together or not.

Rebecca: The binding is not required to perform non-composite authentication, but it is an option.

Yoav Nir: The sharp edges that Mike pointed out were likely always there. We need to decide whether to handle them here or in the protocol (e.g. IKE and TLS).

Rebecca: Agree. There is another draft that discusses how to do non-composite authentication in IKEv2.

Tadahiko Ito: Is this non-composite use defined per protocol? It is not defined for a key that would be used in multiple protocols, right? Cross-protocol attacks should likely be out of scope.

Rebecca: Agree that it is outside of the scope.

Russ H: Certificates often use extended key usage to restrict to which protocol it should be used with. Expect that to continue here.

Uri: I don't think a cross-protocol attack is practical in this context.

Mike O: I've seen certificates with multiple extended key usages.

Uri: I can use the same private key to sign email messages and PDF file. What is the attack?