CFRG (Crypto Forum Research Group)

IETF 113 in Vienna

- Date: Thursday, March 24, 2022 (2 hours)
- Time: 14:30-16:30 (UTC + 1)
- Meetecho: https://meetings.conf.meetecho.com/ietf113/?group=cfrg&short=&item=1
- Onsite tool: https://meetings.conf.meetecho.com/onsite113/?group=cfrg&short=&item=1
- Jabber: cfrg@jabber.ietf.org
- Notes: https://notes.ietf.org/notes-ietf-113-cfrg

RG Chairs:

- Nick Sullivan nick@cloudflare.com
- Stanislav Smyshlyaev smyshsv@gmail.com
- Alexey Melnikov alexey.melnikov@isode.com

Note taker

- Christopher Patton, Rich Salz

Minutes for CFRG at IETF 113

14:30 - Chairs' update.

- Per Chris, move his doc-formatting discussion to end of session.

Document status

- Published!! HPKE
- MISSREF: SPAKE2
- Lots of active docs
    - RGLC: Please provide feedback on KangarooTwelve, chairs need more RG feedback to decide what to do with the document.
    - Chris W.: Latest on ristretto draft? (Number of docs have dependencies on it.)

        - Ready for RGLC?

Other business

- Crypto review panel
    - Purpose: review docs coming through CFRG
    - New members! (Thank you.)
    - René Struik: Review requests take too long to get through (i.e., months). Could this be faster?
        - Chris Patton: How many reviews does the panel get?
            - 2-3 requests/month on average, but varies.
        - René: CFRG mailing list not getting a lot of discussion about documents.
            - Chair: Let's handle on a case-by-case basis
    - René raised concern mailing list not very technical, decisions sometimes made just on basis of one crypto-panel review
- New secretary: Chris Wood! (Thank you.)

Chris Wood, "Key Blinding for Signature Schemes" (15 mins)

```
https://github.com/chris-wood/draft-dew-cfrg-signature-key-blinding
```

- Motivation: In some applications you don't want the signature to leak information about the prover to the verifier. (E.g., "Rate Limiting" issuance protocol in Privacy Pass WG).
- Another setting: Multiple provers
    - Conventional signature schemes don't work for this setting
- Requirements:
    - Per-message public keys are independent from long-term public keys
    - Per-message signatures are unlinkable to the signer
- Proposal: Signature scheme with "key blinding"
- Draft specifies a couple instantiations:
    - PureEdDSA-based variant (RFC 8032); no changes so analysis seems simple
    - ECDSA-based variant; changes complicate analysis, see Q&A
    - Analysis is underway

Chris Patton: Changes to ECDSA?
A: might multiply the keys; body of work on related-key attacks; naive way lends itself to forgeries
Rather than maintain algebraic relationship between input blinding key and output

blinding public key -> hash input key to a scalar; hard to produce collision in this hash to produce a forgery

~ 14:59 Stephen Farrell, “Signatures: deterministic vs randomized” (10+10 mins)

- LAKE WG is working on a protocol called EDHOC for AKE for “small” devices
- EDHOC has some agility: supports ECDSA, EdDSA, …
- Q: Which signatures should be part of mandatory-to-implement (MTI) ciphersuites
- Attack context: “small” devices susceptible to fault-injection attacks
    - Several examples of attacks on signature schemes
    - Stephen’s conclusion: Basically any signature is potentially vulnerable
- Discusson on list: Determnism of EdDSA might make fault injection eaasier (randomized signautres considered less dangerous?)
- Questions for the CFRG:
    - Which of RSA, ECDSA, or EdDSA should be MTI for LAKE; think other WGs want answers too.
    - Perhaps there’s something better?

Discussion

- Rene: can’t avoid side-channels without knowing something about what you’re doing.
- John Preuss Mattson: Large support for a draft that may help with this problem (IPR might be a blocker). Should we make another call for adoption?
    - Alexey: Run the adoption call with IPR disclosure; fall through before
- Phillip Hallam-Baker: Threshold signatures look non-deterministic anyway
    - Signatures should always allow for a “deterministic component”
    - Signature-based protocols are bad because they’re not repudiable
- Bart Preneel: SPA attacks (simple power attacks) attacks should also be taken into account, not only DPA attacks (differential power attacks).
- Thom Wiggers: Fault attacks are much wider than just the crypto: essentially they often allow skipping arbitrary instructions which may allow e.g. skipping a RNG call or the verify operation altogether.
- Guilin: What does MTI mean?
    - A: Mandatory to implement: Goal is to increase interopability, since different implementations are required to implement
- Stephen: cTLS (compact TLS) may have the same problem, for what it’s worth. Agreed to Alexey’s suggestion, will start a thread on the list; hope this presentation made things more visible

~15:17 Chris Patton, “Update on the VDAF (Verifiable Distributed Aggregation Functions) draft” (10+5 mins)

```
https://cjpatton.github.io/vdaf/draft-patton-cfrg-vdaf.html
```

- This topic is related to the work in PPM (Privacy Preserving Measurement); it is the core crypto algorithm for it (currently). Seeking CFRG adoption.
- See slides for details, but: clients send measurement “shards” to multiple aggregators, aggregators collect data from all their clients, all the aggregators send their output to a (single) collector to be combined into a final aggregate result.
- Prio(https://crypto.stanford.edu/prio/paper.pdf) and Poplar (https://eprint.iacr.org/2021/017) are examples. Prio is simpler; Poplar runs multiple collections with different bit-string prefixes to get “heavy hitters” on the desired info (see the referenced IACR eprint for a description of the “heavy hitter” problem).
- Next steps: completion of Poplar, interop definitions, security analysis, hopefully more cryptographic research on new VDAFs.

Chris Wood: Support adoption.
Stephen Farrell: Also support, want CFRG to do this for PPM WG.
Alexey: will call for adoption on list.

~15:30 *Joachim Fabini*, Alexander Hartl, “AES GCM exploit” (10+5 mins)

- Malware is getting more stealthy, wants to “hide” its command messages (e.g., IP headers/flags!)
- Zero-days are inevitable => use CKMD (“Crypto Key Management Device”) that are “uncompromiseable” via zero-days.
- When using a CKMD, AES-GCM deterministic IVs are likely managed by the requesting device.
- Observation: GCM allows decryping by encrypting with the same IV.
- Attack:
    - Uses authentication tag to transfer “hidden information”
    - Circumvents CKDM authentication using weaknesses in GCM
- Mitigations
    - AES-GCM-SIV
    - Have CKMDs genearte IVs (not perfect)

Discussion:

- Jonathan Hoyland: Maybe a compromised receiver can be identified by changing a valid AuthTag on the path. If the compromised receiver does not complain, then he's likely compromised.
- Scott Fluhrer: Why is this specific to GCM, especially since the attacker controls both sides?
    - A: More general in theory, but we were working on a specific attack.

~ 15:49 Nimrod Aviram, "A dual-PRF construction" (10+5 mins)

- Modern protocols derive symmetric keys from shared secret using a KDF
- KDF needs to be a PRF (HMAC is most often used in practice)
- What if you need to derive more than one symmetric key?
    - TLS 1.3
    - Hybrid key exchange
    - Signal's "double ratchet"
- KDF with two keys
    - In many settings, attacker may control one of the keys but not the other
- Can HMAC be used as a dual PRF? *No*, but it wasn't designed for that.
- Proposal (eprint 2022/065)
    - Based on standard primitives: HMAC, a hash function, etc.
    - New "expanding function", based on a hash funcction.
    - Main factor for controlling security level: the "expansion factor"
        - the expanation factor is repeatedly hashing the blocks with different prefixes, and then combining. Current recommendation is 2 or 3.
- Key combiners in practice
    - Lots of protocols use dual-PRFs implicitly – standardization is a good idea.

Discussion

- Chris W.:
    - Claim that HMAC is not a dual-PRF: Many proofs make this assumpion. What's the impact of this claim on existing security analysis?
        - A: Extensive discussion in the paper, let's take this to the list
    - Regarding a draft: It might be nice to generalize this to multiple keys, e.g., for MLS. (There may be a draft kicking around in the IETF somewhere.)

        - A: Construciton is generalizable to this case
    - Chris Patten: Why compare to asymmetric crypto, isn't this like HKDF?
        - A: Slower than HKDF, but we only do it once.
    - Phillip Hallam-Baker: I'd like to see KMAC recommended in the standard (as an alternative to HMAC)
        - A: Maybe; I don't know enough about KMAC to say

~16:07 Bart Preneel, "The AEGIS family of authenticated encryption algorithms" (5+5 mins)

```
https://jedisct1.github.io/draft-aegis-aead/draft-denis-aegis-aead.html
```

- Twice as fast as AES-GCM
- Implementation in Linux!
- Design:
    - Create stream cipher from MAC
    - Building block is the AES round function, rather than the full blockcipher
- Security
    - Targets 128-bit security level
    - Key committing (unlike AES-GCM)
    - More options for nonce length compared to AES-GCM
    - NOT misuse resistant
    - NOT …
    - History
        - Submitted to CAESAR competition, which ran from 2014 to 2018.
        - Cryptanalysis: Attacks for reduced round variants, but nothing below 128 security level for proposal
- Performance
    - Much faster than AES-GCM on modern hardware

Discussion
* Chris Patton: no question, this is a beautiful construction, want to see a draft
* In response to Armando, this is not lightweight, it's fast.
* Chris wood: explain nonce issues? A: Don't re-use nonce, but not as catastrophic as GCM if you do

~16:18 Dan Harkins, "Deterministic Nonce-less Hybrid Public Key Encryption" (5+5 mins)

- HPKE uses uncompressed serialization, propose to go for compact

    - Easy to address
- Out-of-order delivery is incompatible with HPKE
    - Use deterministic AEAD, like Rogaway-Shrimpton-2014 or RFC 5297
    - Use existing AEAD mode and do rolling replay ala RFC 2401
- Draft: draft-harkins-cfrg-dnhpke-01; seeking CFRG adoption
- Code! github.com/danharkins/hpke-wrap

Discussion

- Stephen Farrell
    - Lots of ciphersuites already
    - Are these considerations for the IETF (not CFRG)?
    - Supportive, but maybe elsewhere
- Chris Wood
    - Supportive of codepoints compressed KEM public keys
    - Less supportive of determnistic AEAD
    - Process question: Do we need to assign experts to consider these changes?
        - Chris W.: Follow-up

Chris Wood, “Discussion of pseudocode in CFRG drafts” (15 mins)

No time for the talk itself. Chris Wood’s quick summary:

- Reflection on documents produced by CFRG and how they’re used in the IETF
- Number of things to improve here, especially with resepct to pseudocode.

Meeting ends