From: Peralta, Rene C. (Fed)
To: Apon, Daniel C. (Fed)
Subject: Re: this week - PQC report
Date: Monday, June 22, 2020 3:01:21 PM

? So compiler optimization has to be fully turned off ?

Rene.

From: Apon, Daniel C. (Fed) <daniel.apon@nist.gov>
Sent: Monday, June 22, 2020 2:57 PM
To: Alagic, Gorjan (Assoc) <gorjan.alagic@nist.gov>; Moody, Dustin (Fed) <dustin.moody@nist.gov>; internal-pqc <internal-pqc@nist.gov>
Subject: RE: this week - PQC report

Actually, just to illustrate the trickiness of properly coding crypto algorithms..

I wrote

```
            if (diff != 0)
                res = diff;
```

but really it should be like

```
            if (diff != 0)
                res = diff;
            else
                diff = diff; //a dummy assignment
```

Generally, I think our role (agreeing with Gorjan) is to point out that every line of code, and every function call, should be properly verified as being constant-time. (This is intentionally handing off the responsibility to crypto software engineers at big tech companies who make 5x our salary to do this boring job.)

From: Alagic, Gorjan (Assoc) <gorjan.alagic@nist.gov>
Sent: Monday, June 22, 2020 2:52 PM
To: Apon, Daniel C. (Fed) <daniel.apon@nist.gov>; Moody, Dustin (Fed) <dustin.moody@nist.gov>; internal-pqc <internal-pqc@nist.gov>
Subject: Re: this week - PQC report

Ok, thanks Daniel! It indeed sounds like this was not a particularly deep insight, and probably not worth more research attention. If everyone is already aware of the problem and the fix, then perhaps a mention in the report is not needed.

About your question: I'm not sure what role we should play with regard to verifying

implementations. Certainly that seems like a bigger step than just occasionally using our megaphone to make sure a big oops like this one is avoided. It's also a subject I know essentially nothing about... beyond basic things like the ability to understand why it's bad to terminate a string comparison early.

From: Apon, Daniel C. (Fed) <daniel.apon@nist.gov>
Sent: Monday, June 22, 2020 2:44 PM
To: Moody, Dustin (Fed) <dustin.moody@nist.gov>; Alagic, Gorjan (Assoc) <gorjan.alagic@nist.gov>; internal-pqc <internal-pqc@nist.gov>
Subject: RE: this week - PQC report

Anyway, here's an image of the offending line of code's use of memcmp (just to verify directly – this is from nist.gov/pqcrypto).

The logic of memcmp is as:

```
int memcmp(const unsigned char *m1, const unsigned char *m1, size_t n) {
    size_t i;
    for (i = 0; i < n; ++i ) {
        int diff = m1[i] - m2[i];
        if (diff != 0)
            return (diff < 0) ? -1 : +1;
    }
    return 0;
}
```

However, one could happily use something like:

```
int cst_time_memcmp(const unsigned char *m1, const unsigned char *m1, size_t n) {
    int res = 0, diff;
    if (n > 0) {
        do {
            --n;
            diff = m1[n] - m2[n];
            if (diff != 0)
                res = diff;
        } while (n != 0);
    }
    return (res > 0) - (res < 0);
}
```

with no loss in runtime

From: Apon, Daniel C. (Fed) <daniel.apon@nist.gov>
Sent: Monday, June 22, 2020 2:35 PM
To: Moody, Dustin (Fed) <dustin.moody@nist.gov>; Alagic, Gorjan (Assoc) <gorjan.alagic@nist.gov>; internal-pqc <internal-pqc@nist.gov>
Subject: RE: this week - PQC report

A less terse response..

I read this paper with great joy when it was posted to ePrint, and then – with great sadness – realized it was just about the use of memcmp.

I subsequently discussed the paper in a research meeting call with Jon Katz + student at UMD, and we decided there was nothing worth following up on in terms of research

---

From: Apon, Daniel C. (Fed)
Sent: Monday, June 22, 2020 2:27 PM
To: Moody, Dustin (Fed) <dustin.moody@nist.gov>; Alagic, Gorjan (Assoc) <gorjan.alagic@nist.gov>; internal-pqc <internal-pqc@nist.gov>
Subject: RE: this week - PQC report

I already read it

From: Moody, Dustin (Fed) <dustin.moody@nist.gov>
Sent: Monday, June 22, 2020 2:10 PM
To: Alagic, Gorjan (Assoc) <gorjan.alagic@nist.gov>; Apon, Daniel C. (Fed) <daniel.apon@nist.gov>; internal-pqc <internal-pqc@nist.gov>
Subject: Re: this week - PQC report

I think it would be good to hear more. Can somebody volunteer to read it?

From: Alagic, Gorjan (Assoc) <gorjan.alagic@nist.gov>
Sent: Monday, June 22, 2020 1:34 PM
To: Apon, Daniel C. (Fed) <daniel.apon@nist.gov>; Moody, Dustin (Fed) <dustin.moody@nist.gov>; internal-pqc <internal-pqc@nist.gov>
Subject: Re: this week - PQC report

Well, that may be a good summary of the takeaway of the paper, but this does not mean that the paper is unimportant or that we shouldn't amplify this message in our report. (If this was a triviality, why did they get a CRYPTO paper? Why did the Frodo implementers not notice the problem and claimed constant-time implementation? Is it obvious what to use in place of *memcmp*? Etc.)

If we have good reason to be 100% sure that all the implementers know this and will address

it, then I guess I'm satisfied. If not, I think we should mention it somewhere.

From: Apon, Daniel C. (Fed) <daniel.apon@nist.gov>
Sent: Monday, June 22, 2020 11:22 AM
To: Alagic, Gorjan (Assoc) <gorjan.alagic@nist.gov>; Moody, Dustin (Fed) <dustin.moody@nist.gov>; internal-pqc <internal-pqc@nist.gov>
Subject: RE: this week - PQC report

This paper boils down to one sentence:

"By the way, don't use a standard C call like *memcmp* that is variable time in your implementation."

From: Alagic, Gorjan (Assoc) <gorjan.alagic@nist.gov>
Sent: Monday, June 22, 2020 11:21 AM
To: Moody, Dustin (Fed) <dustin.moody@nist.gov>; internal-pqc <internal-pqc@nist.gov>
Subject: Re: this week - PQC report

Just saw that this appeared on the eprint: https://eprint.iacr.org/2020/743

It's in Crypto this year. Looks like a timing attack on a step in the FO xform which might apply to a lot of our schemes. Looks like they ran it against Frodo and it worked quite well.

Anyone know this paper? Is it worth mentioning in the report, as something we want implementers to address?

From: Moody, Dustin (Fed) <dustin.moody@nist.gov>
Sent: Monday, June 22, 2020 10:20 AM
To: internal-pqc <internal-pqc@nist.gov>
Subject: this week - PQC report

Everyone,

At our last meeting we cleared up several of the comments. Please take a look at the few remaining ones. It looks like we are getting close to done!

A few last assignments:

- Daniel Apon - please check NTRU. You have a comment about wanting to make sure the text is more positively phrased.
- Ray - please check New Hope. There is a comment there.
- Angela - please see if you can re-word the one sentence Lily commented about.
- Daniel ST - please check Rainbow. We want to make sure the write-up presents the case for it being a finalist.
- Everybody - John reworked the write up for SPHINCS+ in a few places. We need to review his changes.

I don't see a need for a meeting tomorrow, but if anybody would like to meet we could. Please send me an email if you'd like to have a meeting.

Thanks!

Dustin