From: Bo Lin <bolinsco@gmail.com> via pqc-forum@list.nist.gov
To: pqc-forum <pqc-forum@list.nist.gov>
Subject: [pqc-forum] Implementation Security in NIST Standardization Process
Date: Friday, June 24, 2022 06:33:29 AM ET

Hi, all,
Hope everyone's well!
The other thread "HertzBleed : power side channel attacks on SIKE" is going very long multiple related but slightly different topics were discussed.

I'd like to create this thread for those who are interested in if and how implementation security should be part of NIST Standardization Process.

My comment on the HertzBleed paper was "In general, for **ephemeral keys**, only the "single trace" attack is valid because if an adversary is not able to figure out a secret with a single trace, they need to try again." CCA does not apply. The SIKE spec, as I read it, uses ephemeral keys - public keys are generated in each transaction.

If **static keys** are used, "... no matter what countermeasures are implemented, implementation security of side-channel and fault injection must be assessed. It applied to any cryptosystems to be deployed, especially, in regulated markets."

My reason for **not** including implementation security in a standard is that it is highly related to / coupled with hardware, especially semiconductor technology and the precision of detection tools, as I gave the following scenario: "... if a part does not leak, then what a side channel adversary can do? (I know, it depends on detection tools, just for an example here.) On the other hand, let me take the time-invariant implementation of the three-point Montgomery ladder as an example here, which is in hot discussion now in this thread. Say, if it is time-invariant, i.e., no time information leakage, but the skA scanning was implemented in if-statement which leaks, then the time-invariant implementation doesn't matter because an adversary can just target the leaky if-statement."

The above scenario (hardware security strength) also apply to fault injection.

It would be very difficult and unnecessary to decide security countermeasure, in terms of implementation security, in a standard because the tight-coupling with the final built product.

Also, although my above comments were started with SIKE, the same apply to other schemes because when a scheme is implemented on a particular part, side-channel attacks and fault injection attacks from a particular angle, depending on the hardware features or adversary's

detection tools, may apply. In this forum, just very recently, a thread posted "Study of Side-Channel and Fault Injection Attacks over Lattice-based Schemes: Survey and New Results".

Regards,

Bo

--

From: Sydney Antonov <ska84@protonmail.com> via pqc-forum <pqc-forum@list.nist.gov>
To: Bo Lin <bolinsco@gmail.com>
CC: pqc-forum <pqc-forum@list.nist.gov>
Subject: Re: [pqc-forum] Implementation Security in NIST Standardization Process
Date: Friday, June 24, 2022 10:28:39 AM ET

I think it's important to distinguish between what I'll call digital and physical side-channels.

Digital side-channels (e.g. cache-based timing attacks) can be exploited remotely and implementations can (and, in my opinion, should) be made immune to them relatively easily (but immune implementations may be less efficient).

Physical side-channels depend on the adversary's physical capabilities to collect data and it's difficult, if not impossible, to make implementations fully immune to them.

I think that when implementation efficiency is being considered, only the efficiency of implementations immune to digital side-channels should be considered.

If even the AES implementations in Intel CPUs are vulnerable to Turbo Boost side-channels I think it's pretty clear that Turbo Boost must be disabled on Intel CPUs to make them immune to digital side-channels. This means implementation efficiency on Intel CPUs should be measured with Turbo Boost disabled.

Sydney

From: Doge Protocol <dogeprotocol1@gmail.com> via pqc-forum@list.nist.gov
To: pqc-forum <pqc-forum@list.nist.gov>
CC: Sydney Antonov <ska84@protonmail.com>, pqc-forum <pqc-forum@list.nist.gov>, boli...@gmail.com <bolinsco@gmail.com>, Sydney Antonov <ska84@protonmail.com>
Subject: Re: [pqc-forum] Implementation Security in NIST Standardization Process
Date: Friday, June 24, 2022 02:05:58 PM ET

In general, it may be preferrable to decouple implementation security from the standardization process, especially with respect to such side channel attacks. However since performance characterestics play a role in standardization, only implementations that are immune/resistant to common side channel attacks should be used to evaluate performance.

There is a big catch with this approach though; not enough scrutiny may have been done on all cryptoschemes for side channel attacks like HertzBleed. So it may not be level playing grounds to evaluate, for example, the first pq scheme (SIKE) on an implementation that slows it down becauae of addressing HertzBleed. Other cryptoschemes may have similar or worse performance degradation as well, but just that not enough research may have been done on this front.

Besides this, hardware keeps evolving, new hardware may expose a newer category of hithereto unknown side channel attacks. What will happen if say, this happens after standardization, resulting in a large perf degradation?

On Friday, June 24, 2022 at 7:28:34 AM UTC-7 Sydney Antonov wrote:

> I think it's important to distinguish between what I'll call digital
> and physical side-channels.
>
> Digital side-channels (e.g. cache-based timing attacks) can be exploited
> remotely and implementations can (and, in my opinion, should) be made
> immune to them relatively easily (but immune implementations may be
> less efficient).
>
> Physical side-channels depend on the adversary's physical capabilities
> to collect data and it's difficult, if not impossible, to make
> implementations fully immune to them.

> I think that when implementation efficiency is being considered, only the efficiency of implementations immune to digital side-channels should be considered.
>
> If even the AES implementations in Intel CPUs are vulnerable to Turbo Boost side-channels I think it's pretty clear that Turbo Boost must be disabled on Intel CPUs to make them immune to digital side-channels. This means implementation efficiency on Intel CPUs should be measured with Turbo Boost disabled.
>
> Sydney

--

From: Doge Protocol <dogeprotocol1@gmail.com> via pqc-forum@list.nist.gov
To: pqc-forum <pqc-forum@list.nist.gov>
CC: Doge Protocol <dogeprotocol1@gmail.com>, Sydney Antonov <ska84@protonmail.com>, pqc-forum <pqc-forum@list.nist.gov>, boli...@gmail.com <bolinsco@gmail.com>
Subject: Re: [pqc-forum] Implementation Security in NIST Standardization Process
Date: Friday, June 24, 2022 03:05:06 PM ET

In above message, making a correction:
Replacing "common side channel attacks" with "easily exploitable side channel attacks that impact a large percentage of devices with commonly used settings".

On Friday, June 24, 2022 at 11:05:46 AM UTC-7 Doge Protocol wrote:

> In general, it may be preferrable to decouple implementation security from the standardization process, especially with respect to such side channel attacks. However since performance characterestics play a role in standardization, only implementations that are immune/resistant to common side channel attacks should be used to evaluate performance.
>
> There is a big catch with this approach though; not enough scrutiny may have been done on all cryptoschemes for side channel attacks like HertzBleed. So it may not be level playing grounds to evaluate, for example, the first pq scheme (SIKE) on an implementation that slows it down becauae of addressing HertzBleed. Other cryptoschemes may have similar or worse performance degradation as well, but just that not enough research may have been done on this front.
>
> Besides this, hardware keeps evolving, new hardware may expose a newer category of hithereto unknown side channel attacks. What will happen if say, this happens after standardization, resulting in a large perf degradation?
>
> On Friday, June 24, 2022 at 7:28:34 AM UTC-7 Sydney Antonov wrote:
>
> > I think it's important to distinguish between what I'll call digital
> > and physical side-channels.
> >
> > Digital side-channels (e.g. cache-based timing attacks) can be exploited
> > remotely and implementations can (and, in my opinion, should) be made
> > immune to them relatively easily (but immune implementations may be

> less efficient).
>
> Physical side-channels depend on the adversary's physical capabilities to collect data and it's difficult, if not impossible, to make implementations fully immune to them.
>
> I think that when implementation efficiency is being considered, only the efficiency of implementations immune to digital side-channels should be considered.
>
> If even the AES implementations in Intel CPUs are vulnerable to Turbo Boost side-channels I think it's pretty clear that Turbo Boost must be disabled on Intel CPUs to make them immune to digital side-channels. This means implementation efficiency on Intel CPUs should be measured with Turbo Boost disabled.
>
> Sydney

--

From: Billy Brumley <bbrumley@gmail.com> via pqc-forum@list.nist.gov
To: pqc-forum <pqc-forum@list.nist.gov>
Subject: Re: [pqc-forum] Implementation Security in NIST Standardization Process
Date: Friday, June 24, 2022 04:11:16 PM ET

Anyone suggesting to decouple side-channel aspects from
standardization should be summarily ignored.

This kind of incompetence is a regression to late 90s, early 2000s
(now legacy) ECC standardization.

It is tragic that I have to remind security experts of this in public.
It's like saying "let's design this system now, and worry about
security later." Design and standardization decisions you make now
affect implementation aspects later. Again, I feel embarrassed for my
field as I forcibly type that out on my keyboard.

Hyvää juhannusta!

BBB

PS Don't interpret my message to mean that HertzBleed is somehow
especially applicable to SIKE. It's not.

On Fri, Jun 24, 2022 at 10:04 PM Doge Protocol <dogeprotocol1@gmail.com> wrote:
>
> In above message, making a correction:
>
> Replacing "common side channel attacks" with "easily exploitable side channel
attacks that impact a large percentage of devices with commonly used settings".
>
> On Friday, June 24, 2022 at 11:05:46 AM UTC-7 Doge Protocol wrote:
>>
>>

>> In general, it may be preferrable to decouple implementation security from the standardization process, especially with respect to such side channel attacks. However since performance characterestics play a role in standardization, only implementations that are immune/resistant to common side channel attacks should be used to evaluate performance.
>>
>> There is a big catch with this approach though; not enough scrutiny may have been done on all cryptoschemes for side channel attacks like HertzBleed. So it may not be level playing grounds to evaluate, for example, the first pq scheme (SIKE) on an implementation that slows it down becauae of addressing HertzBleed. Other cryptoschemes may have similar or worse performance degradation as well, but just that not enough research may have been done on this front.
>>
>> Besides this, hardware keeps evolving, new hardware may expose a newer category of hithereto unknown side channel attacks. What will happen if say, this happens after standardization, resulting in a large perf degradation?
>> On Friday, June 24, 2022 at 7:28:34 AM UTC-7 Sydney Antonov wrote:
>>>
>>> I think it's important to distinguish between what I'll call digital
>>> and physical side-channels.
>>>
>>> Digital side-channels (e.g. cache-based timing attacks) can be exploited
>>> remotely and implementations can (and, in my opinion, should) be made
>>> immune to them relatively easily (but immune implementations may be
>>> less efficient).
>>>
>>> Physical side-channels depend on the adversary's physical capabilities
>>> to collect data and it's difficult, if not impossible, to make
>>> implementations fully immune to them.
>>>
>>> I think that when implementation efficiency is being considered, only
>>> the efficiency of implementations immune to digital side-channels should
>>> be considered.
>>>
>>> If even the AES implementations in Intel CPUs are vulnerable to Turbo
>>> Boost side-channels I think it's pretty clear that Turbo Boost must be
>>> disabled on Intel CPUs to make them immune to digital side-channels. This
>>> means implementation efficiency on Intel CPUs should be measured with

>>> Turbo Boost disabled.

>>>

>>> Sydney

>

> --

> You received this message because you are subscribed to the Google Groups "pqc-forum" group.

> To unsubscribe from this group and stop receiving emails from it, send an email to pqc-forum+unsubscribe@list.nist.gov.

> To view this discussion on the web visit https://groups.google.com/a/list.nist.gov/d/msgid/pqc-forum/ecaa7ac9-7cb6-4aca-8a46-56f1fa6953d0n%40list.nist.gov.

--

You received this message because you are subscribed to the Google Groups "pqc-forum" group.

To unsubscribe from this group and stop receiving emails from it, send an email to pqc-forum+unsubscribe@list.nist.gov.

To view this discussion on the web visit https://groups.google.com/a/list.nist.gov/d/msgid/pqc-forum/CAFeDd5ZDr3W4bVztdqvAdCCNfJp7Uc%3D0748oFHyc0j5_q3TB6g%40mail.gmail.com.

From: Doge Protocol <dogeprotocol1@gmail.com> via pqc-forum@list.nist.gov
To: pqc-forum <pqc-forum@list.nist.gov>
CC: bbru...@gmail.com <bbrumley@gmail.com>
Subject: Re: [pqc-forum] Implementation Security in NIST Standardization Process
Date: Friday, June 24, 2022 05:28:38 PM ET

You are right in many ways, perhaps, the previous post was very generalized.
To clarify;

If its not possible at all to implement a cryptoscheme without addressing easily exploitable side channel attacks, then it will ofcourse be a reason against standardization.

However if there are some implementations that are vulnerable and other implementations of that cryptoscheme that address them with acceptable performance tradeoffs, then it shouldn't become a reason to disqualify.

If we blindly make a binary choice on "does this cryptoscheme have a side-channel attack yes/no", then likely there will be no cryptoscheme that will qualify for standardization, leave alone pq schemes; considering there will always be side channel attacks that can exploit cryptoschemes in specific hardware with specific configuration settings (especially the lesser used onea).

On Friday, June 24, 2022 at 1:11:12 PM UTC-7 bbru...@gmail.com wrote:

> Anyone suggesting to decouple side-channel aspects from
> standardization should be summarily ignored.
>
> This kind of incompetence is a regression to late 90s, early 2000s
> (now legacy) ECC standardization.
>
> It is tragic that I have to remind security experts of this in public.
> It's like saying "let's design this system now, and worry about
> security later." Design and standardization decisions you make now
> affect implementation aspects later. Again, I feel embarrassed for my
> field as I forcibly type that out on my keyboard.
>
> Hyvää juhannusta!
>
> BBB

PS Don't interpret my message to mean that HertzBleed is somehow
especially applicable to SIKE. It's not.

On Fri, Jun 24, 2022 at 10:04 PM Doge Protocol <dogepr...@gmail.com> wrote:
>
> In above message, making a correction:
>
> Replacing "common side channel attacks" with "easily exploitable side channel attacks that impact a large percentage of devices with commonly used settings".
>
> On Friday, June 24, 2022 at 11:05:46 AM UTC-7 Doge Protocol wrote:
>>
>>
>> In general, it may be preferrable to decouple implementation security from the standardization process, especially with respect to such side channel attacks. However since performance characterestics play a role in standardization, only implementations that are immune/resistant to common side channel attacks should be used to evaluate performance.
>>
>> There is a big catch with this approach though; not enough scrutiny may have been done on all cryptoschemes for side channel attacks like HertzBleed. So it may not be level playing grounds to evaluate, for example, the first pq scheme (SIKE) on an implementation that slows it down becauae of addressing HertzBleed. Other cryptoschemes may have similar or worse performance degradation as well, but just that not enough research may have been done on this front.
>>
>> Besides this, hardware keeps evolving, new hardware may expose a newer category of hithereto unknown side channel attacks. What will happen if say, this happens after standardization, resulting in a large perf degradation?
>> On Friday, June 24, 2022 at 7:28:34 AM UTC-7 Sydney Antonov wrote:
>>>
>>> I think it's important to distinguish between what I'll call digital
>>> and physical side-channels.
>>>
>>> Digital side-channels (e.g. cache-based timing attacks) can be exploited
>>> remotely and implementations can (and, in my opinion, should) be made

> >>> immune to them relatively easily (but immune implementations may be
> >>> less efficient).
> >>>
> >>> Physical side-channels depend on the adversary's physical capabilities
> >>> to collect data and it's difficult, if not impossible, to make
> >>> implementations fully immune to them.
> >>>
> >>> I think that when implementation efficiency is being considered, only
> >>> the efficiency of implementations immune to digital side-channels should
> >>> be considered.
> >>>
> >>> If even the AES implementations in Intel CPUs are vulnerable to Turbo
> >>> Boost side-channels I think it's pretty clear that Turbo Boost must be
> >>> disabled on Intel CPUs to make them immune to digital side-channels. This
> >>> means implementation efficiency on Intel CPUs should be measured with
> >>> Turbo Boost disabled.
> >>>
> >>> Sydney
> >
> > --
> > You received this message because you are subscribed to the Google Groups "pqc-forum" group.
> > To unsubscribe from this group and stop receiving emails from it, send an email to pqc-forum+...@list.nist.gov.
> > To view this discussion on the web visit https://groups.google.com/a/list.nist.gov/d/msgid/pqc-forum/ecaa7ac9-7cb6-4aca-8a46-56f1fa6953d0n%40list.nist.gov.

--
You received this message because you are subscribed to the Google Groups "pqc-forum" group.
To unsubscribe from this group and stop receiving emails from it, send an email to pqc-forum+unsubscribe@list.nist.gov.
To view this discussion on the web visit https://groups.google.com/a/list.nist.gov/d/msgid/pqc-forum/35bb61b7-9a6f-4402-ac8b-470c03ccd0fbn%40list.nist.gov.

From: D. J. Bernstein <djb@cr.yp.to> via pqc-forum@list.nist.gov
To: pqc-forum@list.nist.gov
Subject: Re: [pqc-forum] Implementation Security in NIST Standardization Process
Date: Saturday, June 25, 2022 03:54:16 AM ET
Attachments: smime.p7m

The decision to consider side-channel attacks was already made by the official NISTPQC evaluation criteria, which state "Schemes that can be made resistant to side-channel attack at minimal cost are more desirable than those whose performance is severely hampered by any attempt to resist side-channel attacks."

There's an ambiguity here that should be officially resolved. Suppose scheme X costs 100000 without protection and 101000 with protection, while scheme Y costs 10 without protection and 3000 with protection. Does X qualify as "minimal cost" because of the small difference between 101000 and 100000, so the evaluation criteria prefer X over Y, unless someone decides to punish X by speeding up the 100000? (The bizarre features of a "yes" answer have been pointed out before.) Or does Y qualify as "minimal cost" because 3000 is the minimum of {3000,101000}?

Either way, it's important to keep in mind here that _evaluating_ the cost of building side-channel-resistant implementations is challenging:

* For the relatively simple case of timing attacks, there's a clear strategy to verifiably eliminate all data flow from secrets to the side channel, but this takes engineering effort that's often skipped——and cutting any corner can have disastrous consequences; see, e.g., the https://eprint.iacr.org/2021/1485 HQC+BIKE attacks.

* For the much more complicated case of the attacker being able to place (or access) physical sensors close to secrets, it isn't plausible that the data flow can be eliminated. Trying to mask secrets so that they're expensive to recover from the side-channel data is bleeding-edge research, as illustrated by the masked SABER implementation broken in https://eprint.iacr.org/2021/079.

Decisions based on claims regarding the cost of side-channel resistance have to be accompanied by a risk analysis. How likely are the claims to hold up after further investigation? If the claims turn out to be wrong, what's the impact on the decision?

Many people also seem to implicitly assume that comparing costs of _attacks_ against _unprotected_ implementations is a useful predictor of the costs for _users_ of _protected_ implementations. But the available evidence is that, no, it's not a useful predictor. For example, we've previously seen people comparing Classic McEliece to Frodo, so let's try that comparison on these axes:

(1) What are the relative costs of Classic McEliece and Frodo in various applications? This is already a complicated comparison purely from a network-traffic perspective, giving mixed results depending on how many ciphertexts are transmitted per public key, and then considering CPU time and side-channel protection adds further complications.

(2) What are the relative costs of reported power/EM attacks against non-masked implementations of Classic McEliece and Frodo? As far as I know, the only direct comparison available is in https://eprint.iacr.org/2021/849.pdf, which breaks non-masked Frodo in 86016 traces and doesn't succeed in breaking non-masked Classic McEliece.

Even if the answer to #2 is stable (I'm skeptical, and certainly wouldn't recommend making any decisions on that basis), clearly #2 is not capturing most aspects of #1. #1 is directly addressing NISTPQC evaluation criteria, whereas the connection of #2 to the evaluation criteria is tenuous at best.

——D. J. Bernstein

From: Bo Lin <bolinsco@gmail.com> via pqc-forum@list.nist.gov
To: pqc-forum <pqc-forum@list.nist.gov>
CC: bbru...@gmail.com <bbrumley@gmail.com>
Subject: Re: [pqc-forum] Implementation Security in NIST Standardization Process
Date: Saturday, June 25, 2022 06:29:14 AM ET

Dear BBB,
Thanks for your comment.

Implementation security to a product is well understood now and there is no question about its importance. As I mentioned in the other thread, it will be thoroughly assessed in a regulated market.
If implementation security should be specified by a standard, rather than as a requirement, then what is your opinion to proceed it? Shall a standard candidate submitter provide a list of an implementation implementation guidance? Then if a vendor follow this list of recommendations and the product is broken in the field because the unexpected vulnerability found later, who shall be responsible? Who's going to regulate it?

You mentioned ECC standard. Did you mean ECDSA? The standard is still being suggested in some new spec. see

https://www.emvco.com/emv_insights_post/what-technical-advances-can-we-expect-from-emvco-in-2021/

I believe SM2 in https://www.iso.org/standard/76382.html is another spec.

It is an example that standard should not provide implementation guidance because around 2000, ECDSA's potential vulnerabilities to side channel attacks and fault injection attacks were not as clear as now. At the moment, those potential vulnerabilities are considered in product building by selecting proper parts and implementing proper countermeasures.

What's your opinion?

Thanks,

Bo
On Friday, June 24, 2022 at 9:11:12 PM UTC+1 bbru...@gmail.com wrote:

> Anyone suggesting to decouple side-channel aspects from
> standardization should be summarily ignored.

This kind of incompetence is a regression to late 90s, early 2000s (now legacy) ECC standardization.

It is tragic that I have to remind security experts of this in public. It's like saying "let's design this system now, and worry about security later." Design and standardization decisions you make now affect implementation aspects later. Again, I feel embarrassed for my field as I forcibly type that out on my keyboard.

Hyvää juhannusta!

BBB

PS Don't interpret my message to mean that HertzBleed is somehow especially applicable to SIKE. It's not.

On Fri, Jun 24, 2022 at 10:04 PM Doge Protocol <dogepr...@gmail.com> wrote:
>
> In above message, making a correction:
>
> Replacing "common side channel attacks" with "easily exploitable side channel attacks that impact a large percentage of devices with commonly used settings".
>
> On Friday, June 24, 2022 at 11:05:46 AM UTC-7 Doge Protocol wrote:
>>
>>
>> In general, it may be preferrable to decouple implementation security from the standardization process, especially with respect to such side channel attacks. However since performance characterestics play a role in standardization, only implementations that are immune/resistant to common side channel attacks should be used to evaluate performance.
>>
>> There is a big catch with this approach though; not enough scrutiny may have been done on all cryptoschemes for side channel attacks like HertzBleed. So it may not be level playing grounds to evaluate, for example, the first pq scheme (SIKE) on an implementation that slows it down becauae of addressing HertzBleed. Other cryptoschemes may have similar or worse performance degradation as well, but just that not enough research may have been

done on this front.
>>
>> Besides this, hardware keeps evolving, new hardware may expose a newer category of hithereto unknown side channel attacks. What will happen if say, this happens after standardization, resulting in a large perf degradation?
>> On Friday, June 24, 2022 at 7:28:34 AM UTC-7 Sydney Antonov wrote:
>>>
>>> I think it's important to distinguish between what I'll call digital
>>> and physical side-channels.
>>>
>>> Digital side-channels (e.g. cache-based timing attacks) can be exploited
>>> remotely and implementations can (and, in my opinion, should) be made
>>> immune to them relatively easily (but immune implementations may be
>>> less efficient).
>>>
>>> Physical side-channels depend on the adversary's physical capabilities
>>> to collect data and it's difficult, if not impossible, to make
>>> implementations fully immune to them.
>>>
>>> I think that when implementation efficiency is being considered, only
>>> the efficiency of implementations immune to digital side-channels should
>>> be considered.
>>>
>>> If even the AES implementations in Intel CPUs are vulnerable to Turbo
>>> Boost side-channels I think it's pretty clear that Turbo Boost must be
>>> disabled on Intel CPUs to make them immune to digital side-channels. This
>>> means implementation efficiency on Intel CPUs should be measured with
>>> Turbo Boost disabled.
>>>
>>> Sydney
>
> --
> You received this message because you are subscribed to the Google Groups "pqc-forum" group.
> To unsubscribe from this group and stop receiving emails from it, send an email to pqc-forum+...@list.nist.gov.

> To view this discussion on the web visit https://groups.google.com/a/list.nist.gov/d/msgid/pqc-forum/ecaa7ac9-7cb6-4aca-8a46-56f1fa6953d0n%40list.nist.gov.

--

From: Markku-Juhani O. Saarinen <mjos.crypto@gmail.com> via pqc-forum@list.nist.gov
To: Bo Lin <bolinsco@gmail.com>
CC: pqc-forum <pqc-forum@list.nist.gov>, bbru...@gmail.com <bbrumley@gmail.com>
Subject: Re: [pqc-forum] Implementation Security in NIST Standardization Process
Date: Saturday, June 25, 2022 07:09:59 AM ET

Hi Bo,

I may be repeating myself a bit, but as someone involved with building commercial side-channel protected PQC:

If certified at FIPS 140-3 levels 3 and 4, I'd expect a module to be tested with standard leakage detection methods. The draft SP 800-140F already referred to ISO 17825 and 20085-1/2 for testing of non-invasive (side-channel) attack mitigations, and is coming into ISO 19790:2022 Annex F. Note that especially 17825 is currently under revision.

Under Common Criteria we'd expect there to be suitable protection profiles in near future. Scoring of attack potential can be similar to what is currently done for high-assurance commercial smart cards, secure elements, HSMs etc. This usually implies AVA_VAN vulnerability assessment (The relevant attacks try key recovery, not just leakage detection.)

Note that some countermeasures may be very specific to an implementation architecture and platform or otherwise not be public; the scoring system actually discourages openness somewhat (I don't really agree with the wisdom of this - but this is how it seems to be.) In any case it is difficult even for the algorithm designers to foresee what exactly those countermeasures are.

However, I'd suggest using "masking-friendliness" as an metric; performance and implementation area when that countermeasure is consistently applied.

Cheers,

Markku

On Sat, Jun 25, 2022, 11:29 Bo Lin <bolinsco@gmail.com> wrote:

> Dear BBB,
> Thanks for your comment.

> Implementation security to a product is well understood now and there is no question about its importance. As I mentioned in the other thread, it will be thoroughly assessed in a regulated market.
> If implementation security should be specified by a standard, rather than as a requirement, then what is your opinion to proceed it? Shall a standard candidate submitter provide a list of an implementation implementation guidance? Then if a vendor follow this list of recommendations and the product is broken in the field because the unexpected vulnerability found later, who shall be responsible? Who's going to regulate it?
>
> You mentioned ECC standard. Did you mean ECDSA? The standard is still being suggested in some new spec. see
>
> https://www.emvco.com/emv_insights_post/what-technical-advances-can-we-expect-from-emvco-in-2021/
>
> I believe SM2 in https://www.iso.org/standard/76382.html is another spec.
>
> It is an example that standard should not provide implementation guidance because around 2000, ECDSA's potential vulnerabilities to side channel attacks and fault injection attacks were not as clear as now. At the moment, those potential vulnerabilities are considered in product building by selecting proper parts and implementing proper countermeasures.
>
> What's your opinion?
>
> Thanks,
>
> Bo
> On Friday, June 24, 2022 at 9:11:12 PM UTC+1 bbru...@gmail.com wrote:
>
> > Anyone suggesting to decouple side-channel aspects from
> > standardization should be summarily ignored.
> >
> > This kind of incompetence is a regression to late 90s, early 2000s
> > (now legacy) ECC standardization.
> >
> > It is tragic that I have to remind security experts of this in public.
> > It's like saying "let's design this system now, and worry about
> > security later." Design and standardization decisions you make now
> > affect implementation aspects later. Again, I feel embarrassed for my

field as I forcibly type that out on my keyboard.

Hyvää juhannusta!

BBB

PS Don't interpret my message to mean that HertzBleed is somehow
especially applicable to SIKE. It's not.

On Fri, Jun 24, 2022 at 10:04 PM Doge Protocol <dogepr...@gmail.com> wrote:
>
> In above message, making a correction:
>
> Replacing "common side channel attacks" with "easily exploitable side channel attacks that impact a large percentage of devices with commonly used settings".
>
> On Friday, June 24, 2022 at 11:05:46 AM UTC-7 Doge Protocol wrote:
>>
>>
>> In general, it may be preferrable to decouple implementation security from the standardization process, especially with respect to such side channel attacks. However since performance characterestics play a role in standardization, only implementations that are immune/resistant to common side channel attacks should be used to evaluate performance.
>>
>> There is a big catch with this approach though; not enough scrutiny may have been done on all cryptoschemes for side channel attacks like HertzBleed. So it may not be level playing grounds to evaluate, for example, the first pq scheme (SIKE) on an implementation that slows it down becauae of addressing HertzBleed. Other cryptoschemes may have similar or worse performance degradation as well, but just that not enough research may have been done on this front.
>>
>> Besides this, hardware keeps evolving, new hardware may expose a newer category of hithereto unknown side channel attacks. What will happen if say, this happens after standardization, resulting in a large perf degradation?
>> On Friday, June 24, 2022 at 7:28:34 AM UTC-7 Sydney Antonov wrote:
>>>

> >>> I think it's important to distinguish between what I'll call digital
> >>> and physical side-channels.
> >>>
> >>> Digital side-channels (e.g. cache-based timing attacks) can be exploited
> >>> remotely and implementations can (and, in my opinion, should) be made
> >>> immune to them relatively easily (but immune implementations may be
> >>> less efficient).
> >>>
> >>> Physical side-channels depend on the adversary's physical capabilities
> >>> to collect data and it's difficult, if not impossible, to make
> >>> implementations fully immune to them.
> >>>
> >>> I think that when implementation efficiency is being considered, only
> >>> the efficiency of implementations immune to digital side-channels should
> >>> be considered.
> >>>
> >>> If even the AES implementations in Intel CPUs are vulnerable to Turbo
> >>> Boost side-channels I think it's pretty clear that Turbo Boost must be
> >>> disabled on Intel CPUs to make them immune to digital side-channels. This
> >>> means implementation efficiency on Intel CPUs should be measured with
> >>> Turbo Boost disabled.
> >>>
> >>> Sydney
> >
> > --
> > You received this message because you are subscribed to the Google Groups "pqc-forum" group.
> > To unsubscribe from this group and stop receiving emails from it, send an email to pqc-forum+...@list.nist.gov.
> > To view this discussion on the web visit https://groups.google.com/a/list.nist.gov/d/msgid/pqc-forum/ecaa7ac9-7cb6-4aca-8a46-56f1fa6953d0n%40list.nist.gov.

--
You received this message because you are subscribed to the Google Groups "pqc-forum" group.
To unsubscribe from this group and stop receiving emails from it, send an email to pqc-forum+unsubscribe@list.nist.gov.

> To view this discussion on the web visit https://groups.google.com/a/list.nist.gov/d/msgid/pqc-forum/e40ca4f3-8593-4a72-bf85-20d8af51e6ecn%40list.nist.gov.

--