Dear BIKE-2 designers,

The KAT files for BIKE-2 have wrong or somewhat misleading format.

While in the documentation you state that the public key is r bits long, KAT files has public keys with n bits long where the systematic part is unnecessarily printed out.

The same situation is with the ciphertexts that instead of being r bits long, they are n bits long, with half of the bits being zeros.

I could not find any remark in the documentation or in the README files that clarifies your decision to produce KAT files with unnecessary and redundant outputs.

I see this as a technical programming mistake, and should be fixed.

Regards,
Danilo!

--
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. Visit this group at https://groups.google.com/a/list.nist.gov/group/pqc-forum/.
Dear Danilo,

Thank you for your comment.

BIKE-2 has public keys in systematic form, thus only r bits need to be communicated. BIKE-1 and BIKE-3 do not have public keys in systematic form, thus require n=2r bits.

The reference code implements all BIKE-1, BIKE-2 and BIKE-3 variants, and we wanted to keep the same format for the KAT files, to simplify things. Thus, we made BIKE-2 to be n bits long as well, although the first r bits can obviously be omitted, if/when the protocol is used.

Please note that the additional implementation of BIKE-2 does output only r bits.

Best Regards,
BIKE team

-----Original Message-----
From: Danilo Gligoroski [mailto:danilog@ntnu.no]
Sent: Friday, December 29, 2017 1:40 AM
To: pqc-comments@nist.gov
Cc: pqc-forum@list.nist.gov
Subject: [pqc-forum] OFFICIAL COMMENT: BIKE

Dear BIKE-2 designers,

The KAT files for BIKE-2 have wrong or somewhat misleading format.

While in the documentation you state that the public key is r bits long, KAT files has public keys with n bits long where the systematic part is unnecessarily printed out.

The same situation is with the ciphertexts that instead of being r bits long, they are n bits long, with half of the bits being zeros.

I could not find any remark in the documentation or in the README files that clarifies your decision to produce KAT files with unnecessary and redundant outputs.

I see this as a technical programming mistake, and should be fixed.

Regards,
Danilo!
Hi Danilo,

Is “ciphertexts indistinguishable from random” one of the NIST desiderata? If so, I must have missed it.

Or do you mean that this property suggests some other weakness?

Thanks,
— Mike

> On Jan 4, 2018, at 11:22 PM, Danilo Gligoroski <danilog@ntnu.no> wrote:
> >
> > Topic: Simple ciphertext distinguisher for BIKE-1 and BIKE-2
> > >
> > Dear BIKE-1 and BIKE-2 designers,
> > >
> > If we look at your supplied KAT files for BIKE-2 Level 1 and Level 3 we can notice the following pattern:
> > For BIKE-2, Level 1, the Hamming weights for the first 16 ciphertexts are:
> > (5112, 5062, 5082, 5088, 5152, 5062, 5028, 5046, 5022, 5056, 5076, 5096, 5100, 5114, 5064, 5196).
> > As we can see, all weights are even.
> >
> > For BIKE-2, Level 3, the Hamming weights for the first 16 ciphertexts are:
> > (9975, 9989, 9903, 9915, 9869, 9891, 9927, 10075, 9923, 9895, 10031, 9989, 9843, 10059, 9921).
> > As we can see, all weights are odd.
> >
> > I transformed this observation into the following simple ciphertext distinguisher for BIKE-1 and BIKE-2.
> >
> > For BIKE-1, the public key is (f0, f1). Let the received ciphertext is c = (c0, c1).
> > Compute: c2 = c0 f0^{-1} + c1 f1^{-1}
> >
> > For BIKE-1 Level 1 and Level 5:
> > If HammingWeight(c2) is odd then
> >  - claim that c was not produced by the public key (f0, f1)
> > else
> >  - make a guess with probability 1/2 whether the c was produced by the public key (f0, f1)
> >
> > Success probability of winning the guess game is 0.75
> >
> > For BIKE-1 Level 3
Hi Mike,

When I sent my observation about the ciphertext distinguisher for BIKE-1 and BIKE-2 I was motivated by this part of the NIST call:
"NIST will perform a thorough analysis of the submitted algorithms in a manner that is open and transparent to the public, as well as encourage the cryptographic community to also conduct analyses and evaluation. This combined analysis will inform NIST's decision on the subsequent development of post-quantum standards."

I understand that sentence from the NIST call as "if you notice something, say something" (about some proposed cipher :-)). For the concrete ciphers BIKE-1 and BIKE-2, I don't know will it lead to some other weakness, but our crypto community will know it as one of the properties of BIKE-1 and BIKE-2.

Best regards,
Danilo!

On 05/01/2018 09:15, Mike Hamburg wrote:
> Hi Danilo,
> 
> Is "ciphertexts indistinguishable from random" one of the NIST desiderata? If so, I must have missed it.
> 
> Or do you mean that this property suggests some other weakness?
> 
> Thanks,
> — Mike
>
>> On Jan 4, 2018, at 11:22 PM, Danilo Gligoroski <danielog@ntnu.no> wrote:
>>
>> Topic: Simple ciphertext distinguisher for BIKE-1 and BIKE-2
>>
>>
>> Dear BIKE-1 and BIKE-2 designers,
>>
>> If we look at your supplied KAT files for BIKE-2 Level 1 and Level 3 we can notice the following pattern:
>> For BIKE-2, Level 1, the Hamming weights for the first 16 ciphertexts are:
>> {5112, 5062, 5082, 5088, 5152, 5062, 5028, 5046, 5022, 5056, 5076, 5096, 5100, 5114, 5064, 5196}.
>> As we can see, all weights are even.
>>
>> For BIKE-2, Level 3, the Hamming weights for the first 16 ciphertexts are:
>> {9975, 9989, 9903, 9915, 9869, 9891, 9927, 10075, 9923, 9895, 10031, 9989, 9843, 10059, 9921}. 1