From:	Sumanta <sumanta.sarkar@gmail.com>
Sent:	Wednesday, November 27, 2019 3:58 AM
To:	lwc-forum
Cc:	bishu.math.ynwa@gmail.com; Mridul Nandi; Raghvendra Rohit; #MUSTAFA MAHMOUD MOHAMMED KAIRALLAH#
Subject:	[lwc-forum] ROUND 2 OFFICIAL COMMENT: ORANGE

Dear Orange Team,

We have analyzed Algorithm 3 of the modified Orange (we call it Orange-2). We can trivially show the existence of forgery as follows.

Suppose |AD| = 2n and |M| = 0.
Then in proc_hash:
D_0 = AD
X_0 = K||N
S = [P(X_0)]_n
Y0 = c_0*P(X_0)
X_1 = Y_0 + pad(D_0) = Y_0 + AD   .... (1)
return (X_1, S)

So output of "enc" function is (\lambda, proc_tg(X_1))

As per the description of proc_tg, it is invertible, hence we can obtain X_1 from proc_tg(X_1).
Then from (1), Y_0 = X_1+AD and subsequently
P(X_0) = c_0^{-1}*Y_0   .... (2)

Now consider AD1 such that |AD1| < n .
Then reconstruct Y_0 = c_1*P(X_0)
next X_1 = Y_0 + pad(AD1)
S = [P(X_0)]_n

return (X_1, S)
Then (\lambda, proc_tg(X_1)) becomes a valid output of "enc" function.

The crux of the attack is that the Tag generated by the permutation P is of the same size as that of the input state, and hence leaking the full state. Further if we look carefully, once we get P(X_0) from (2), then we can invert it to get X_0 = K || N, that is it leads to key recovery.

We checked the proof of Orange-2, but there is no mention on the bound of the tag size \tau. In our opinion b-\tau >= 112 is crucial for security. Precisely speaking, there is a key recovery attack that works with O(1/2^(b-\tau)), and this factor is missing in the security proof as well.

We would like to note that in Section 2.1 of the official Orange-1 specification/C implementation, it is mentioned that the tag is limited to 128 bits. Applying this will prevent the attack, however, this is not the case for Orange-2 as can be seen in Algorithm 3 and in the NIST workshop paper "Security Proof of Orange-Zest". Hence, we conclude that the specification of Orange-2 leads to easy key recovery and forgery attacks and both the specification and security proof have flaws that require fixing.

We would also like to point out to Theorem 1 of the workshop paper, which specifies b = r + c, where c = 128. Orange-2 claims to have full rate, so r = 256; implying b = 384. However, if the security bound 4\sigma_v q_p/2^b is the dominant bound at T =

2^112 and D = 2^45, then this implies b <<< 384, which is a contradiction. We believe the dominant term is 4\sigma_e\sigma_v/2^c.

Thanks and best regards

Sumanta, Mustafa and Raghav

--

To unsubscribe from this group, send email to lwc-forum+unsubscribe@list.nist.gov
Visit this group at https://groups.google.com/a/list.nist.gov/d/forum/lwc-forum