Very promising algorithm and protocol. Unfortunately, the most-performant implementation fails to compile with the modern assembler. I think it needs to be fixed, so people can evaluate the true best performance in their environments on their best hardware:

```
$ make
/usr/bin/gcc -no-pie -O3 -mavx2 -l/opt/local/include -o PQCgenKAT_kem cpapke.c fips202.c kem.c ntt.c poly.c
PQCgenKAT_kem.c precomp.c reduce.c rng.c verify.c const.c fips202x4.c ntt_double.s keccak4x/KeccakP-1600-times4-SIMD256.c -L/opt/local/lib -lcrypto
clang: warning: argument unused during compilation: '-nopie' [-Wunused-command-line-argument]
ntt_double.s:90:1: error: 32-bit absolute addressing is not supported in 64-bit mode
   vmovdqu q_vector,%ymm0
^   ntt_double.s:95:1: error: 32-bit absolute addressing is not supported in 64-bit mode
   vmovdqu qinv_vector,%ymm1
   . . .
ntt_double.s:4533:1: error: 32-bit absolute addressing is not supported in 64-bit mode
   vmovdqu q_vector,%ymm0
^   ntt_double.s:4538:1: error: 32-bit absolute addressing is not supported in 64-bit mode
   vmovdqu qinv_vector,%ymm1
^   make: *** [PQCgenKAT_kem] Error 1
```