一种新的OT基于黄政柏的快速ot和privdpi的可重用思想：

只进行一次setup工作，MB端保存Ii的模糊规则

**第一次通信**：与原版相同，只是部分符号重新表述，方便表达

Client端：

由原始会话k生成密钥数组：

K={(,),(,),...,(,)},其中m是规则或令牌的长度，每对(,){0,1}，L一般是128位

生成随机数r：（原版的r每次会话都需要重新生成，我提的草案只进行一次setup，所以r不再重新生成，但是r需要保留，多占用了存储空间）

*K随机化：*

**

**

*其中1im。两端得到(,),MB每次和两端执行OT协议得到,而不需要更新K。MB得到一组(,,...,)，通过异或操作得到加密规则：*

Ik(R）=K1,R1⊕K2,R2⊕…⊕Km,Rm

Ik(R）=k1,R1⊕k2,R2⊕…⊕km,Rm⊕r

MB端保存Ik(R）作为模糊规则

第一次通信的对于token部分稍有区别（为了和之后的通信使用相同的公式，对第一次通信也做了修改。也可以不修改，只需要判断是否是第一次通信即可）

E（token）=k1,t1⊕k2,t2⊕…⊕km,tm⊕k(k为TLS的会话密钥也异或进来，不会影响安全性，因为前面的ki,ti是根据k生成的128个128位随机串。MB端得到的是混杂有随机数r的，不会泄露密钥k)

α=H（E（token）⊕W）（W仍使用128位随机数）

β=r⊕W⊕k

把α和β发送给MB（安全性说明：这里是MB唯一得到没有被Hash函数处理的k，但是依然有随机数W进行保护，MB也不知道r）

匹配：

α\*=H（Ik(R）⊕β）= H（k1,R1⊕k2,R2⊕…⊕km,Rm⊕W⊕k）

α=H（E（token）⊕W）=H（k1,t1⊕k2,t2⊕…⊕km,tm⊕k⊕W）

至此，第一次通信的设计完成，与原版没什么大的区别

===============================================================================**第i次通信**

不再进行setup部分（节省通信开销和计算开销）

MB再本地保存Ik(R）

Client端保存密钥随机数组K，随机数r，和新的会话密钥k，

E（token）= k1,t1⊕k2,t2⊕…⊕km,tm⊕k，

（之前出现过的token计算步骤也可以减少128次异或，只异或⊕k，）（安全性：

Token都有随机数W异或，并且使用H函数，k，不会泄露）

α=H（E（token）⊕W）（W仍使用128位随机数）

β=r⊕W⊕k，

MB计算α\*=H（Ik(R）⊕β）= H（k1,R1⊕k2,R2⊕…⊕km,Rm⊕W⊕k，）

α=H（E（token）⊕W）=H（k1,t1⊕k2,t2⊕…⊕km,tm⊕k，⊕W）

比较α\*和α即可

