解决了恶意程序的检测机器学习问题

构架一个trie的数据结构加快搜素速度

机器学习模型可以加入检测

分析结果只有客户端能看见---比blindbox进步之处，保证了分析结果的安全性

所有在中间间的流量都保持加密状态（与设计的协议相比，设计的协议也是token保持加密状态，但是token的加密并不是aes加密，而是异或加密）

完整性方面：各个协议都有考虑到

-------系统框架

ssl流量不在通过middlebox转发，而是直接发送给另一个client

对攻击的定义不在是只有关键字的匹配，而是增加了程序符合已知的训练模型，机器学习的匹配

连接建立提供7个参数，用于后续的加解密和检测。握手过程middlebox并不参加

sap连接也采用令牌化的token，源连接不做任何处理。与其他协议的不同在于，为了适应机器学习，token需要进行特殊的预处理。（其他协议仅仅进行分割）

MB进行两次匹配：关键字的匹配和机器学习模型的匹配。机器学习的分析结果mb也不应该看到，将其发送给接收者。mb可以断开关键字匹配的规则。

接受者可以判断是否经过MB的审查

-------协议设计

bilndbox使用的setup工作，这里使用公钥和私钥完成（效率问题需要解决）

setup步骤不需要进行信息交换。

使用离散对数问题构造加密算法

使用这个算法MB不需要解密有效负载（设计协议也不需要解密，只进行加密完成匹配）

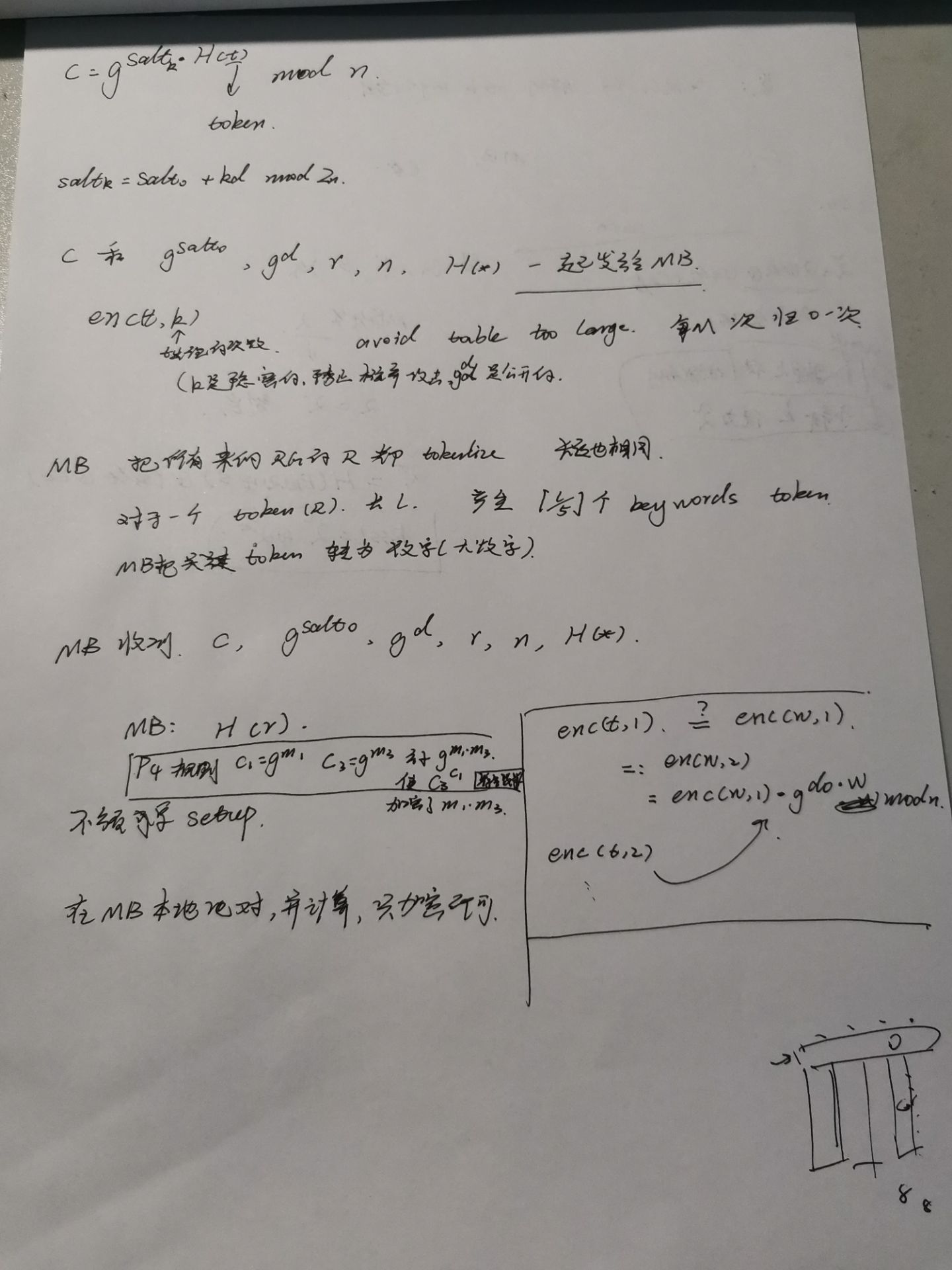
-----关键字匹配部分

匹配不算成立，所有都匹配，且偏移量也算才成立

令牌大小（5字节，别的8字节）-----滑动窗口长速度慢

偏移量通过匹配的token之间相隔几个来判断偏移量是否正确

匹配方法-----图片



--------解决mb的遗留问题

如何有效地将多个匹配的关键字标记组合到一个关键字中，因为关键字也是由MB标记的？

如何将多个匹配的关键字映射到特定规则， 因为 一个 规则 可能 包含 多个 关键字？

mb匹配到一个关键字后可以减少搜索下一个token的范围

使用逐级的hash表解决这个问题

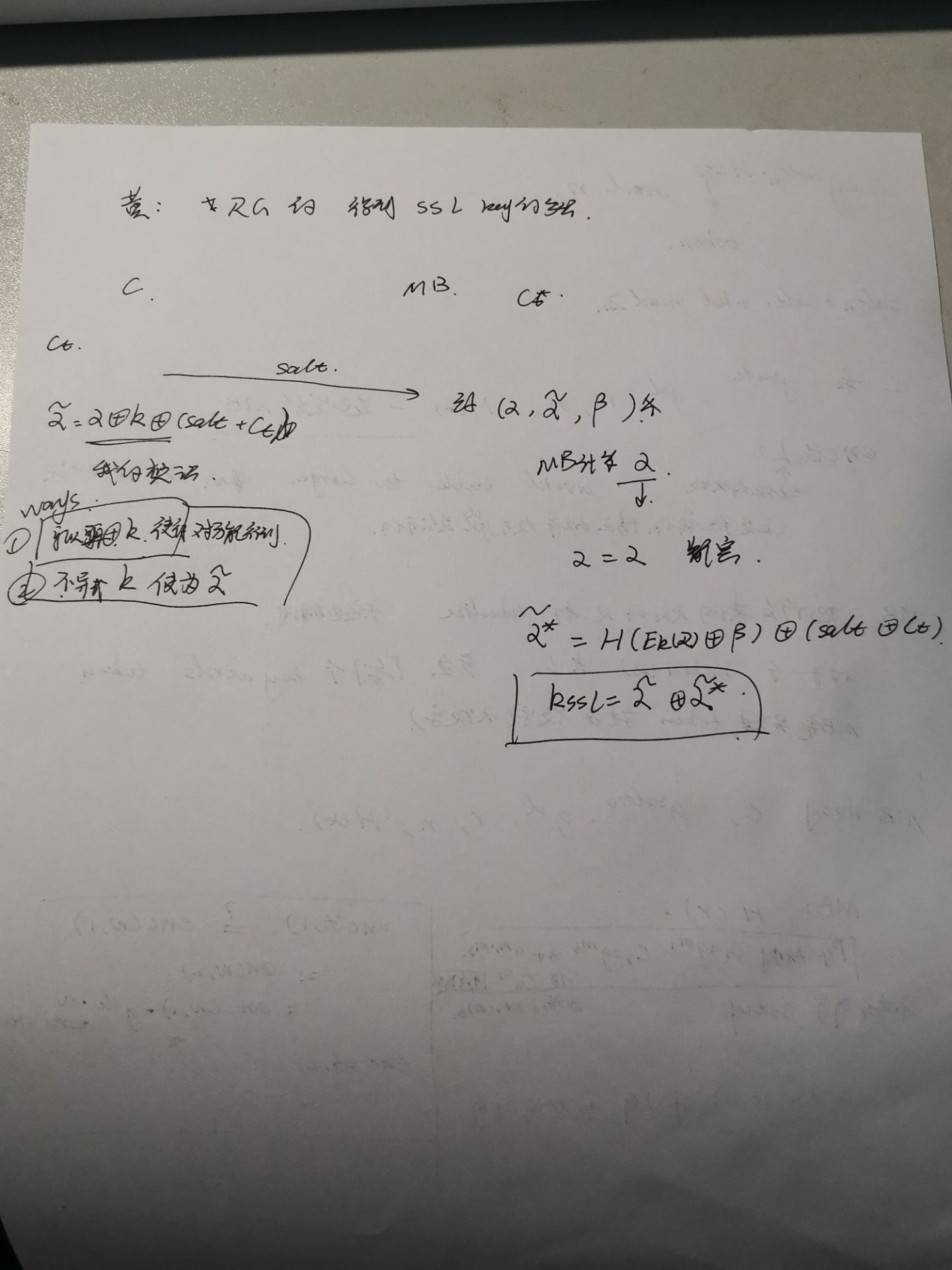
-----------------------------正则表达式匹配

黄的正则表达式匹配：

mb可以解密流量进行正则表达式的判断

通过异或操作得到加密的sslkey，当匹配上时，直接异或就得到了ssl密钥

（图片）



感想：

虽然没有进行setup，但是MB得到rule后的计算并不少，也是只进行一次计算。比起blindbox来说好了不少，通信流量减少了，计算开销减少了。比起privdpi来说通信流量减少了，但是MB端的计算量是否有变化并没有比较实验。

对于正则表达式来说，都需要进行解密流量才能进行匹配，是否可以在加密状态下进行正则表达式匹配呢？难度太大了，并不知道加密状态下的每个字符是构成，正则规则都没办法写，如果想要清楚每个字符的加密后是什么样的，是不可能的事情，加密无论是ssl还是tls都使用的aes加密，块加密并不能得到单个字符的一一对应。同一个token中的相同字母的加密结果都不同，只能解密后在判断。

解密后进行正则表达式匹配。