**创新创业项目说明**

小组成员：胡锦垚，董力源，徐徽成，冯瑞珩

**说明：**1、由于我们是在第三方软件写完代码后由最终的项目主要完成人提交上去的，所以上传时只是一个人上传了，具体贡献度排序即为各项目完成人的排列顺序；

2、由于小组成员都为第一次使用Github提交代码，所以对相关流程不太清楚，便商议先完成项目任务，完成后统一提交，所以所有项目代码都为同一天内提交，所带来的诸多不便还请老师谅解。

**成员账号名称：**

董力源：antibioticv

胡锦垚：Joinwant

徐徽成：silgold

冯瑞珩：Chenke1905

**项目名称：Birthday Attacks**

简介：利用生日攻击原理，在有限的时间复杂度下尝试找出sm3杂凑算法的一对碰撞。

完成人：胡锦垚 徐徽成 董力源 冯瑞珩

代码说明：在sm3加密算法的基础上，设置一个生日攻击函数，仅需调用函数便可以寻找碰撞。

运行指导：无需输入，只需运行即可，若结果中出现success即为此次运行找到了碰撞，若仅有over，则没有发现碰撞。运行多次，发现成功概率接近1/2。

实验截图:



**项目名称：Length extension attacks**

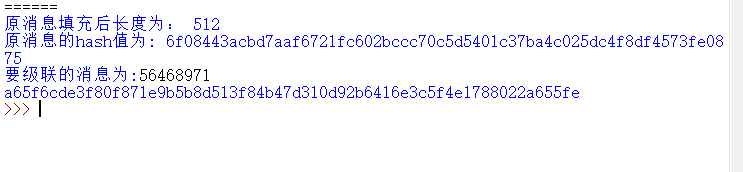
简介：利用长度扩展攻击原理，在已知一个消息的长度和对应的杂凑值后，让原消息再加上自定义的消息，得出新消息的杂凑值。

完成人：胡锦垚 董力源 冯瑞珩 徐徽成

代码说明：在sm3加密算法的基础上，增添长度扩展攻击部分，完成所述要求。

运行指导：运行后开始可以看到原消息填充后长度（实际上原消息长度已知，这里已经是根据原消息算出它应有的填充后长度）和它的hash值。只需输入要级联上的消息，等待输出即可。

实验截图:



**项目名称：Merkle Tree**

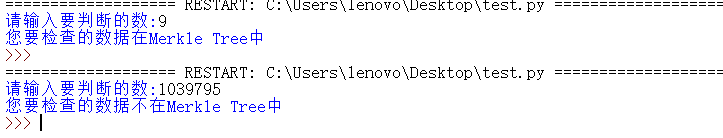
简介：先构建一个Merkle Tree，并使得可以检测数据是否在树中，即完成proof。

完成人：胡锦垚 冯瑞珩 徐徽成 董力源

代码说明：在sm3加密算法的基础上，从低向上一步步完成消息的级联和hash最终完成整棵树。在proof过程中，不断选择枝丫进行级联和hash，到达最顶端时看hash值是否与根相等，相等就存在，不相等即不存在。（目前，判断时仅支持输入整数）

运行指导：因为构建整棵树的过程中，树枝太多（2\*\*20），每步向上都要调用sm3算法，所以在构造整棵树的过程中较慢，等待较长时间后，可以输入要判断的数据，等待判断结果输出即可。

实验截图:

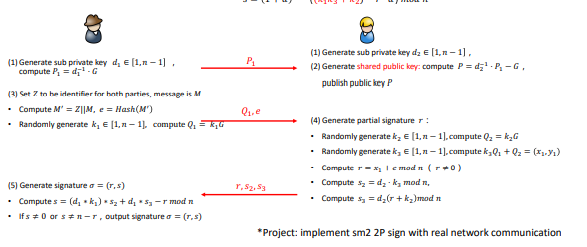


**项目：Implement sm2 2P sign with real network communication**

简介：编程模拟网络通信，基于sm2来实现两方签名

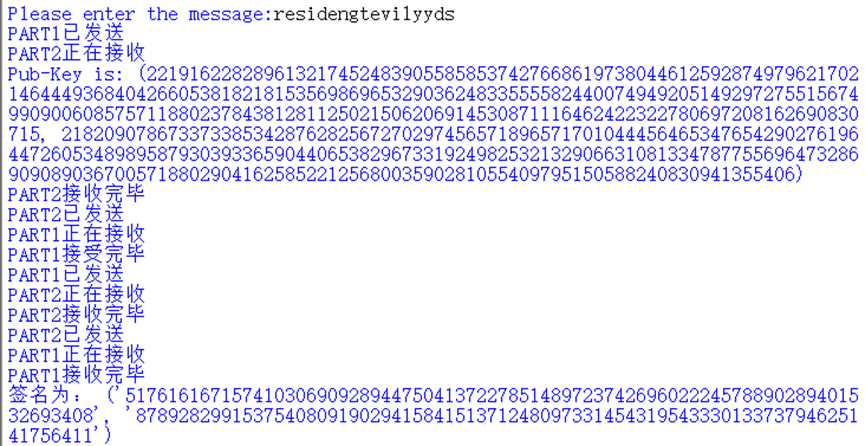
主要完成人：董力源，胡锦垚，徐徽成，冯瑞珩

代码说明：编程实现下图要求，其余见代码注释



运行指导：打开python 3.0及以上版本，直接点击run即可，运行文件：sm2 2p sign.py。

实验截图：

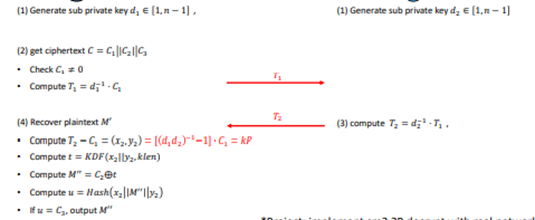


**项目：Implement sm2 2P decrypt with real network communication**

简介：编程模拟网路通信，基于sm2来实现两方解密

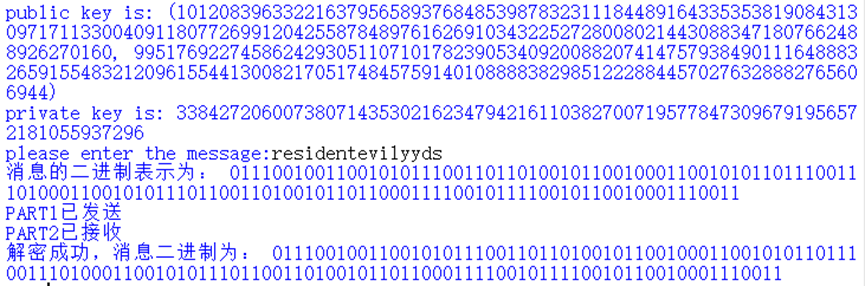
主要完成人：董力源，冯瑞珩，胡锦垚，徐徽成

代码说明：编程实现下图要求，其余见代码注释；



运行指导：打开python 3.0及以上版本，直接点击run即可，运行文件：sm2 2p decrypt.py。

实验截图：



**项目：Implement a PGP scheme with SM2**

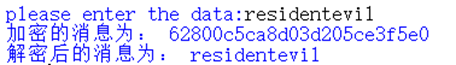
简介：生成密钥，对消息进行对称加密，利用sm2算法将密钥加密，解密时先利用sm2解密密钥，随后对称解密消息。

主要完成人：董力源，徐徽成，冯瑞珩，胡锦垚

代码说明：引用gmssl实现sm2加解密，引用Cryptdome.Cipher实现对称密码的实现（本次采用AES进行加解密），密钥生成函数参考见： [gmssl实现sm2](https://blog.csdn.net/u014651560/article/details/113744296)。其余见代码注释；

运行指导：打开python 3.0及以上版本，直接点击run即可，运行文件：sm2 PGP scheme.py。

实验截图：



**项目名称：impl sm2 with RFC6979**

简介：在sm2的基础上用RFC6979来防止重复使用同一私钥

完成人：徐徽成 胡锦垚 董力源 冯瑞珩

代码说明：在sm2加密算法的基础上，用k = SHA256(d + HASH(m))公式来使私钥生成复杂化，从而避免重复使用同一私钥而导致出现的安全性隐患。

运行指导：打开python 3.0及以上版本，直接点击run即可，运行文件：sm2 PGP scheme.py。

实验截图：



**项目名称：report on the application of this deduce technique in Ethereum with ECDSA** 简介：说明在特殊情况下仅凭签名推导出ECDSA的公钥

完成人：徐徽成 冯瑞珩 胡锦垚 董力源

任务说明：在猜测Z与公钥部分无关时，可以用e=H（Z||M）推出公钥，而不需要知道公私钥，产生了一定的安全隐患。因而需要使用与公钥有一定联系的Z来防止这样情况的出现。

运行指导：直接查看文档。

**项目名称：research report on MPT**

简介：研究并了解MPT这一数据结构

完成人：徐徽成 董力源 冯瑞珩 胡锦垚

任务说明：在默克尔树和前缀树两种树结构的基础上进行优化而开发出的集有两者优点的数据结构，在以太坊（Ethereum）有重要应用。

运行指导：直接查看文档。

**项目：send a tx on Bitcoin testnet, and parse the tx data down to every bit, better**

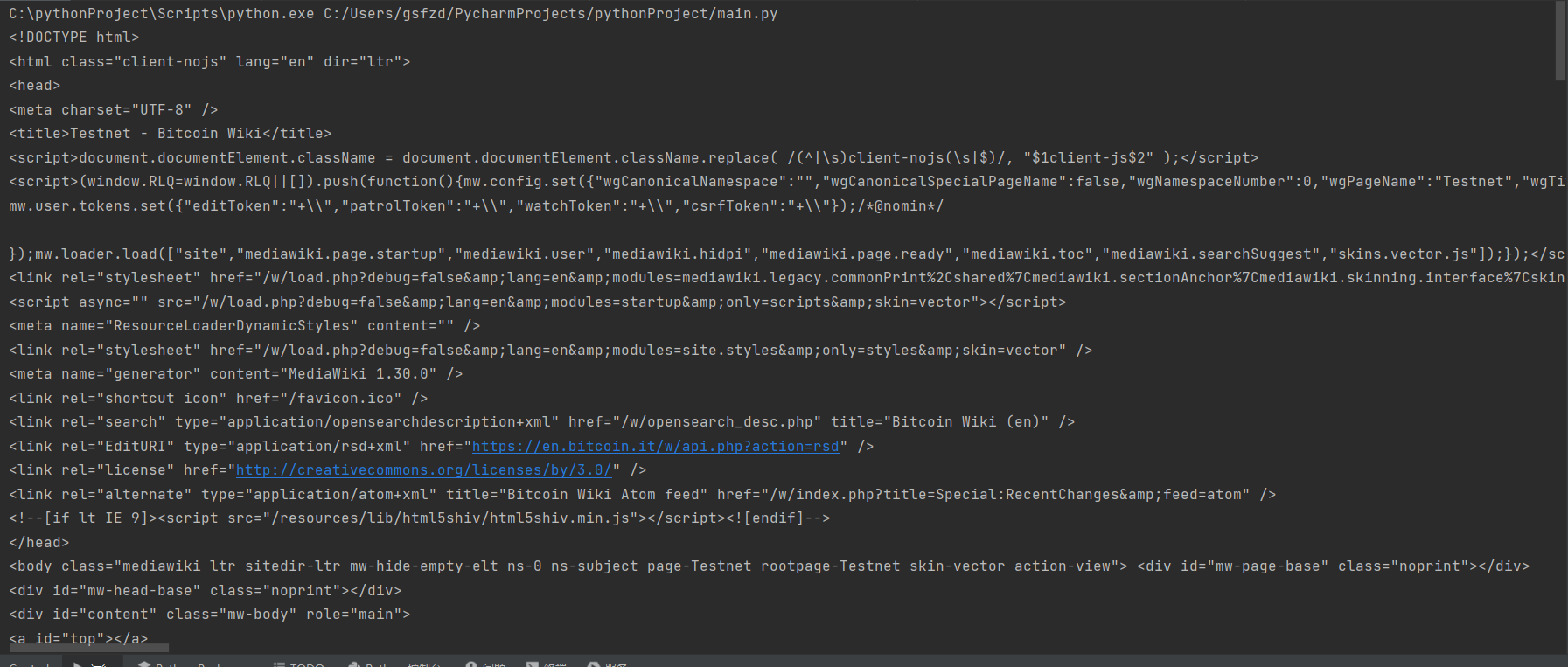
**write script yourself**

简介：向Bitcoin textnet网站发送信息，输出返回的响应信息

完成人：冯瑞珩，董力源，徐徽成，胡锦垚

代码说明：利用requests.post，python网络编程向指定网页发送消息

实验截图：



**项目：forge a signature to pretend that you are Satoshi**

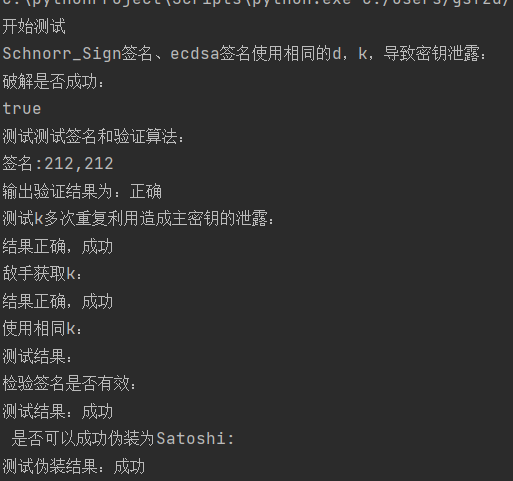
简介：伪造合法签名使得你可以被认为是Satoshi

完成人：冯瑞珩，董力源，徐徽成，胡锦垚

代码说明：私钥：随机取一个在 ( 1 ， n − 1 ) 区间上的整数作为私钥 公钥：Q = da ∗ G ，其中Q是公钥，可知公钥是通过私钥生成的，而G是椭圆曲线上的基点

生成一个临时密钥k 计算 P = k ∗ G其中P是椭圆曲线上的一个点取P点的x坐标， r ≡ x (mod n) 使用SHA1函数计算信息的哈希值，使用H(m)表示。

实验截图：



**项目：impl sm4 with AESNI**

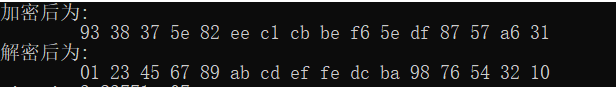
简介：利用AESNI方法和SIMD指令实现SM4的优化

完成人：董力源，胡锦垚，徐徽成，冯瑞珩

代码说明：文件分为一个头文件及两个源文件，具体说明见代码注释

运行指导：需导入头文件和两个源文件，同时还需导入工具包packges.config，随后直接开始调试即可完成

实验截图：



**已完成项目为：**

1、implement the naïve birthday attack of reduced SM3

2、implement length extension attack for SM3, SHA256, etc

3、Impl Merkle Tree following RFC6962

4、report on the application of this deduce technique in Ethereum with ECDSA

5、impl sm2 with RFC6979

6、Implement a PGP scheme with SM2

7、implement sm2 2P sign with real network communication

8、implement sm2 2P decrypt with real network communication

9、send a tx on Bitcoin testnet, and parse the tx data down to every bit, better write script yourself

10、forge a signature to pretend that you are Satoshi

11、research report on MPT

**未完成项目为：**

1、implement the Rho method of reduced SM3

2、do your best to optimize SM3 implementation (software)

3、Try to Implement this scheme(below)

4、verify the above pitfalls with proof-of-concept cod

5、Implement the above ECMH scheme

6、PoC impl of the scheme, or do implement analysis by Google

7、Find a key with hash value “sdu\_cst\_20220610” under a message composed of your name followed by your student ID. For example, “San Zhan 202000460001”.

8、Find a 64-byte message under some k fulfilling that their hash value is symmetrical

9、project idea