BHP 私钥、公钥、地址介绍

BHP 源码：https://github.com/BhpAlpha/

私钥

BHP 的公私钥验证方法使用的是 ECC 椭圆曲线算法。

这类非对称加密算法的基本机制如下，私钥你保留着，公钥是公开的。你用私钥对一串数据进行签名。别人可以用 数据、签名、公钥 三者，断定这三者是不是匹配，签名是否有效。

在 BHP 区块链上最主要的权限认证方式就是签名，所以私钥很重要，要保护好.

源码：https://github.com/BhpAlpha/bhp/blob/master/bhp/Wallets/Wallet.cs

```
public WalletAccount CreateAccount()
        {
            byte[] privateKey = new byte[32];
            using (RandomNumberGenerator rng = RandomNumberGenerator.Create())
            {
                rng.GetBytes(privateKey);
            }
            WalletAccount account = CreateAccount(privateKey);
            Array.Clear(privateKey, 0, privateKey.Length);
            return account;
        }
```

公钥

公钥就是私钥的一部分，可以由私钥算出，但是反过来，公钥无法算出私钥。这个计算是单向的。

源码：https://github.com/BhpAlpha/bhp/blob/master/bhp/Wallets/KeyPair.cs

```
public KeyPair(byte[] privateKey)
        {
            if (privateKey.Length != 32 && privateKey.Length != 96 && privateKey.Length != 104)
                throw new ArgumentException();
            this.PrivateKey = new byte[32];
            Buffer.BlockCopy(privateKey, privateKey.Length - 32, PrivateKey, 0, 32);
```

```
                if (privateKey.Length == 32)
                {
                    this.PublicKey = Cryptography.ECC.ECCurve.Secp256r1.G * privateKey;
                }
                else
                {
                    this.PublicKey = Cryptography.ECC.ECPoint.FromBytes(privateKey,
Cryptography.ECC.ECCurve.Secp256r1);
                }
            }
```

地址脚本

地址脚本，看起来像是对公钥前面后面各加了一个字节

```
pubkey=03bdb1311ca8fe82e2ed65483ae18a37d4219beef14fd2abd86158746a0a28effb
addrscript=2103bdb1311ca8fe82e2ed65483ae18a37d4219beef14fd2abd86158746a0a28effbac
```

实际上他是一个智能合约，将他反编译的话、

就是：

PushBytes[pubkey]

CheckSig

这样两条指令。

当你访问你的账户的时候，比如用你的账户给别人转账，就会调用这个合约来验证。

这个合约的意义是用你的公钥和交易数据 和交易签名进行验证。

只有你签名的合约才能动你的账户

地址 ScriptHash

地址 ScriptHash 就是地址脚本取了个 Hash

```
byte[] script = GetScriptFromPublicKey(publicKey);
var scripthash = sha256.ComputeHash(script);
scripthash = ripemd160.ComputeHash(scripthash);
return scripthash;
```

一次 sha256，一次 ripemd160

地址

地址和 WIF 很相似，不过他是 ScriptHash 加了盐，加了验证功能，然后 base58 编码