FISCO BCOS的外部账户生成

原创 白兴强 FISCO BCOS开源社区 2019-08-06

白兴强

FISCO BCOS核心开发者

优秀的联盟链就是要快

AUTHOR | 作者

1

账户是什么

FISCO BCOS使用账户来标识和区分每一个独立的用户。在采用公私钥体系的区块链系统里，每一个账户对应着一对公钥和私钥。其中，由公钥经哈希等安全的单向性算法计算后，得到的地址字符串被用作该账户的账户名，即账户地址。而仅有用户知晓的私钥则对应着传统认证模型中的密码。这类有私钥的账户也常被称为外部账户或账户。

FISCO BCOS中部署到链上的智能合约在底层存储中也对应一个账户，我们称这类账户为合约账户。与外部账户的区别在于，合约账户的地址是部署时确定，根据部署者的账户地址及其账户中的信息计算得出，并且合约账户没有私钥。

本文将主要介绍外部账户的生成，不讨论合约账户相关内容，关于生成的外部账户的使用方式，请参考FISCO BCOS各SDK的说明文档。

账户的使用场景

在FISCO BCOS中，账户有以下使用场景：

- SDK需要持有外部账户私钥，使用外部账户私钥对交易签名。区块链系统中，每一次对合约写接口的调用都是一笔交易，而每笔交易需要用账户的私钥签名。

- 权限控制需要外部账户的地址。FISCO BCOS权限控制模型，根据交易发送者的外部账户地址，判断是否有写入数据的权限。

- 合约账户地址唯一的标识区块链上的合约。每个合约部署后，底层节点会为其生成合约地址，调用合约接口时，需要提供合约地址。

外部账户的生成

出于方便，下文中提及外部账户，均简称为账户。

FISCO BCOS提供了get_account.sh脚本和Web3SDK接口用以创建账户，同时，console和Web3SDK也支持加载创建的账户私钥，用于交易签名。

用户可将账户私钥，存储为PEM或PKCS12格式的文件。其中，PEM格式使用明文存储私钥，而PKCS12格式使用用户提供的口令加密存储私钥，详情可以参考这里：
https://zh.wikipedia.org/wiki/PKCS_12。

使用get_account.sh脚本生成账户（操作）

- **获取脚本**

```
curl -LO https://media.githubusercontent.com/media/FISCO-BCOS/LargeFil
```

执行上面的指令，看到如下输出，则下载到了正确的脚本，否则请重试。

```
Usage: ./get_account.sh
    default       generate account and store private key in PEM format
    -p            generate account and store private key in PKCS12 for
    -k [FILE]     calculate address of PEM format [FILE]
    -P [FILE]     calculate address of PKCS12 format [FILE]
    -h Help
```

- **生成PEM格式存储的账户私钥**

```
bash get_account.sh
```

执行上面的命令，可以得到类似下面的输出，包括账户地址和以账户地址为文件名的私钥PEM文件。

```
[INFO] Account Address   : 0xee5fffba2da55a763198e361c7dd627795906ead
[INFO] Private Key (pem) : accounts/0xee5fffba2da55a763198e361c7dd6277
```

- **生成PKCS12格式存储的账户私钥**

```
bash get_account.sh -p
```

执行上面的命令，可以得到类似下面的输出，按照提示输入密码，生成对应的p12文件。

```
Enter Export Password:
Verifying - Enter Export Password:
[INFO] Account Address : 0x02f1b23310ac8e28cb6084763d16b25a2cc7f5e1
[INFO] Private Key (p12) : accounts/0x02f1b23310ac8e28cb6084763d16b25a
```

使用Java-SDK接口生成账户

有时我们需要在代码中生成新的账户，这个时候就需要借助Java-SDK(项目名为web3SDK)提供的接口。

如下所示，Java-SDK提供生成账户、计算账户地址和获取公钥等功能，与get_account.sh脚本相比更多了对国密账户生成的支持。

```
import org.fisco.bcos.web3j.crypto.EncryptType
import org.fisco.bcos.web3j.crypto.Credentials
import org.fisco.bcos.web3j.crypto.gm.GenCredential

//创建普通账户
EncryptType.encryptType = 0;
//创建国密账户，向国密区块链节点发送交易需要使用国密账户
// EncryptType.encryptType = 1;
Credentials credentials = GenCredential.create();
//账户地址
String address = credentials.getAddress();
//账户私钥
String privateKey = credentials.getEcKeyPair().getPrivateKey().toStri
//账户公钥
String publicKey = credentials.getEcKeyPair().getPublicKey().toString
```

上述接口可以直接在Java业务代码中使用，同时Java-SDK也提供了加载PEM格式或PKCS12格式存储的私钥的功能，详情请参考这里：
https://fisco-bcos-documentation.readthedocs.io/zh_CN/latest/docs/sdk/sdk.html#id5

账户地址计算方法

FISCO BCOS的账户地址由ECDSA公钥计算得来，对ECDSA公钥的16进制表示计算keccak-256sum哈希，取计算结果的后20字节的16进制表示作为账户地址，每个字节需要两个16进制数表示，所以账户地址长度为40。FISCO BCOS的账户地址与以太坊兼容。

下面简要演示账户地址计算步骤：

- 使用OpenSSL生成椭圆曲线私钥，椭圆曲线的参数使用secp256k1。执行下面的命令，生成PEM格式的私钥并保存在ecprivkey.pem文件中。

```
openssl ecparam -name secp256k1 -genkey -noout -out ecprivkey.pem
```

- 根据私钥计算公钥，然后使用公钥计算对应的账户地址。需要获取keccak-256sum工具，可以从这里下载：
  https://github.com/vkobel/ethereum-generate-wallet/tree/master/lib

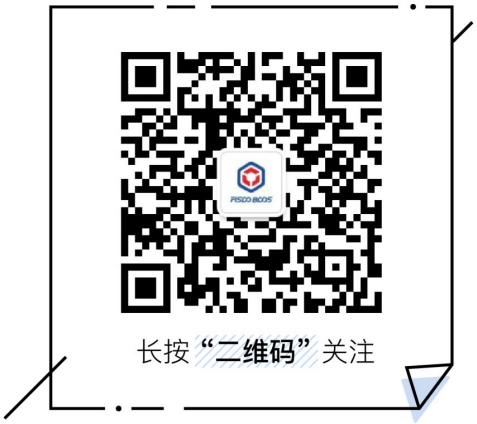
```
openssl ec -in ecprivkey.pem -text -noout 2>/dev/null| sed -n '7,11p'
```

得到类似下面的输出，就是计算得出ecprivkey.pem对应的账户地址。

```
dcc703c0e500b653ca82273b7bfad8045d85a470
```

总结

本文简要介绍了FISCO BCOS外部账户的定义、生成以及账户地址的计算方法。未来，我们也会开放更多好用的配套组件，来帮助开发者更方便安全地管理账户。


长按“二维码”关注