

笔记: gocio.sudo 合约

gocio.sudo部署

gocio.sudo 合约的部署需要创建 gocio.sudo 账户。

gocio.sudo 需要足够的RAM来托管合同以及足够分CPU和网络带宽来部署合同。

创建gocio.sudo账户

1. 部署准备

创建账户前, gocio.sudo 首先需要提交创建账户的事务; 所有的区块生产者的事务存储在JSON文件中, 以便 gocio.msigs 合约直接使用cleos命令调用。

cleos system newaccount: 创建新账户命令 (仅接受单个公钥作为新账户的所有者和活动权限); 通过active许可可满足新账户的所有者和主动权限gocio。

- 生成一个事务以捕获创建新用户所涉及的必要操作

```
$ cleos system newaccount -s -j -d --transfer --stake-net "1.000 GOC" --
stake-cpu "1.000 GOC" --buy-ram-kbytes 50 gocio gocio.sudo
GOC5n442Qz4yVc4LbdPCDnxNSseAiUCrNjRxAfPhUvM8tWS5svid6 >
generated_account_creation_trx.json

{
  "expiration": "2018-11-12T06:22:21",
  "ref_block_num": 1491,
  "ref_block_prefix": 1786775602,
  "max_net_usage_words": 0,
  "max_cpu_usage_ms": 0,
  "delay_sec": 0,
  "context_free_actions": [],
  "actions": [{
    "account": "gocio",
    "name": "newaccount",
    "authorization": [{
      "actor": "gocio",
      "permission": "active"
    }
  ],
  "data":
    "0000000000ea106500004d1a03ea1065010000000100027514f5ab53a2fee4810aea47a22b
    8b2f509ec6a9aa4996285a03c513a4e0947101000000010000000100027514f5ab53a2fee48
    10aea47a22b8b2f509ec6a9aa4996285a03c513a4e0947101000000"
  }, {
    "account": "gocio",
    "name": "buyrambytes",
```

```

    "authorization": [{
      "actor": "gocio",
      "permission": "active"
    }
  ],
  "data": "0000000000ea106500004d1a03ea106500c80000"
}, {
  "account": "gocio",
  "name": "delegatebw",
  "authorization": [{
    "actor": "gocio",
    "permission": "active"
  }
  ],
  "data":
"0000000000ea106500004d1a03ea10651027000000000000004474f43000000001027000000
00000004474f4300000000001"
}
],
"transaction_extensions": [],
"signatures": [],
"context_free_data": []
}

```

```

ttt@tangtingtings-MacBook-Pro:~/GOCint/build/programs/cleos$ cat newaccount_payload.json
{
  "creator": "gocio",
  "name": "gocio.sudo",
  "owner": {
    "threshold": 1,
    "keys": [],
    "accounts": [{
      "permission": {"actor": "gocio", "permission": "active"},
      "weight": 1
    }],
    "waits": []
  },
  "active": {
    "threshold": 1,
    "keys": [],
    "accounts": [{
      "permission": {"actor": "gocio", "permission": "active"},
      "weight": 1
    }],
    "waits": []
  }
}

```

- 生成一个包含创建账户 `gocio::newaccount` 的事务，保存于 `generated_newaccount_trx.json`，将在后面被用到：

```

$ cleos --wallet-url http://127.0.0.1:6666 --url http://127.0.0.1:8000 push
action -s -j -d gocio newaccount newaccount_payload.json -p gocio >
generated_newaccount_trx.json

```

```

ttt@tangtingtings-MacBook-Pro:~/GOCint/build/programs/cleos$ cat generated_newaccount_trx.json
{
  "expiration": "2018-11-12T08:39:31",
  "ref_block_num": 17935,
  "ref_block_prefix": 1347860107,
  "max_net_usage_words": 0,
  "max_cpu_usage_ms": 0,
  "delay_sec": 0,
  "context_free_actions": [],
  "actions": [{
    "account": "gocio",
    "name": "newaccount",
    "authorization": [{
      "actor": "gocio",
      "permission": "active"
    }]
  },
  {
    "data": "0000000000ea106500004d1a03ea1065010000000001000000000ea1065000000a8ed3232010000010000000001000000000ea106500000000a8ed3232010000"
  }
],
  "transaction_extensions": [],
  "signatures": [],
  "context_free_data": []
}

```

- 生成一个包含 `gocio::setpriv` 的事务，赋予 `gocio.sudo` 账户特权：

```

$ cleos --wallet-url http://127.0.0.1:6666 --url http://127.0.0.1:8000 push
action -s -j -d gocio setpriv '{"account": "gocio.sudo", "is_priv": 1}' -p
gocio > generated_setpriv_trx.json

```

generated_setpriv_trx.json:用于使 `gocio.sudo` 账户具有特权

```

ttt@tangtingtings-MacBook-Pro:~/GOCint/build/programs/cleos$ cat generated_setpriv_trx.json
{
  "expiration": "2018-11-12T08:45:00",
  "ref_block_num": 18595,
  "ref_block_prefix": 135009729,
  "max_net_usage_words": 0,
  "max_cpu_usage_ms": 0,
  "delay_sec": 0,
  "context_free_actions": [],
  "actions": [{
    "account": "gocio",
    "name": "setpriv",
    "authorization": [{
      "actor": "gocio",
      "permission": "active"
    }]
  },
  {
    "data": "00004d1a03ea106501"
  }
],
  "transaction_extensions": [],
  "signatures": [],
  "context_free_data": []
}

```

- 建立generated_newaccount_trx.json文件，通过复制create_sudo_account_trx.json，并修改前3个字段来实现：

```

ttt@tangtingtings-MacBook-Pro:~/GOCint/build/programs/cleos$ cat create_sudo_account_trx.json
{
  "expiration": "2018-11-12T08:39:31",
  "ref_block_num": 0,
  "ref_block_prefix": 0,
  "max_net_usage_words": 0,
  "max_cpu_usage_ms": 0,
  "delay_sec": 0,
  "context_free_actions": [],
  "actions": [{
    "account": "gocio",
    "name": "newaccount",
    "authorization": [{
      "actor": "gocio",
      "permission": "active"
    }]
  },
  {
    "data": "0000000000ea106500004d1a03ea10650100000000010000000000ea106500000000a8ed323201000001000000000010000000000ea106500000000a8ed3232010000"
  }
],
  "transaction_extensions": [],
  "signatures": [],
  "context_free_data": []
}

```

字段 `ref_block_num` 和 `ref_block_prefix` 的值设为0, 在 `gocio.misg::exec` 的时候会设置相应的值。

对于 `expiration` 的值, 在 `gocio.misg::exec` 的时候会设置相应的值, 但是在交易的提交、批准和执行的过程中, 如果当前时间超过 `expiration` 设定的时间, 则无法执行, 所以此处设置为任意久一个时间值。

- 将 `generated_account_creation_trx.json` 中的 `actions` 的除第一个之外的所有actions添加到 `create_sudo_account_trx.json` 的 `actions` 中, 并将 `generated_setpriv_trx.json` 中的 `actions` 中的内容添加到 `create_sudo_account_trx.json` 的 `actions` 中, 结果如下:

```

$ cat create_sudo_account_trx.json
{
  "expiration": "2018-11-12T08:39:31",
  "ref_block_num": 0,
  "ref_block_prefix": 0,
  "max_net_usage_words": 0,
  "max_cpu_usage_ms": 0,
  "delay_sec": 0,
  "context_free_actions": [],
  "actions": [{
    "account": "gocio",
    "name": "newaccount",
    "authorization": [{
      "actor": "gocio",
      "permission": "active"
    }]
  },
  {
    "data": "0000000000ea106500004d1a03ea10650100000000010000000000ea106500000000a8ed323201000001000000000010000000000ea106500000000a8ed3232010000"},{
    "account": "gocio",
    "name": "buyrambytes",
    "authorization": [{

```

```

        "actor": "gocio",
        "permission": "active"
    }
],
"data": "0000000000ea106500004d1a03ea106500c80000"
},{
    "account": "gocio",
    "name": "delegatebw",
    "authorization": [{
        "actor": "gocio",
        "permission": "active"
    }]
},
"data":
"0000000000ea106500004d1a03ea1065102700000000000004474f43000000001027000000
00000004474f430000000001"
},{
    "account": "gocio",
    "name": "setpriv",
    "authorization": [{
        "actor": "gocio",
        "permission": "active"
    }]
},
"data": "00004d1a03ea106501"
}
],
"transaction_extensions": [],
"signatures": [],
"context_free_data": []
}

```

active的区块生产者的json文件，在之后用 `gocio.msigs` 合约提交事务时需要用到。

2. 提交创建gocio.sudo账户

通过多签合约，选取一个lead block producer来提交create_sudo_account_trx.json中的事务：

```

$ cleos multisig propose_trx createsudo producer_permissions.json
create_sudo_account_trx.json blkproducera
executed transaction:
bf6aaa06b40e2a35491525cb11431efd2b5ac94e4a7a9c693c5bf0cfed942393  744 bytes
772 us
#   gocio.msigs <= gocio.msigs::propose
{"proposer":"blkproducera","proposal_name":"createsudo","requested":
[{"actor":"blkproducera","permis...
warning: transaction executed locally, but may not be confirmed by the
network yet

```

3. 审核通过创建gocio.sudo账户

对于已经提交了的事务，每一个活跃的区块生产者应该第一时间审查，确保不会批准他们不同意的事务。

通过多签合约，使用以下命令，使提交的事务被审核：

```
$ cleos multisig review blkproducera createsudo >
create_sudo_account_trx_to_review.json
```

每一个区块生产者进行简单的事务审核。

```
$ cleos multisig approve blkproducera createsudo '{"actor": "blkproducerb",
"permission": "active"}' -p blkproducerb
```

4. 执行创建gocio.sudo账户

若上述的审核中，通过率达到要求后（例如，21个区块生产者中至少通过15个），任何人都可以使用 `gocio.msig::exec` 执行批准创建 `gocio.sudo` 账户的操作：

```
$ cleos multisig exec blkproducera createsudo blkproducera
executed transaction:
7ecc183b99915cc411f96dde7c35c3fe0df6e732507f272af3a039b706482e5a 160 bytes
850 us
# gocio.msig <= gocio.msig::exec
{"proposer":"blkproducera","proposal_name":"createsudo","executer":"blkprod
ucera"}
warning: transaction executed locally, but may not be confirmed by the
network yet
```

之后，可以查看 `gocio.sudo` 账户是否已经创建：

```
$ cleos get account gocio.sudo
privileged: true
permissions:
  owner      1:      1 gocio@active,
  active     1:      1 gocio@active,
memory:
  quota:     49.74 KiB   used:      3.33 KiB

net bandwidth:
  staked:    1.0000 SYS      (total stake delegated from
account to self)
  delegated: 0.0000 SYS      (total staked delegated to
account from others)
  used:      0 bytes
```

```
available:      2.304 MiB
limit:         2.304 MiB

cpu bandwidth:
  staked:       1.0000 SYS      (total stake delegated from
account to self)
  delegated:    0.0000 SYS      (total staked delegated to
account from others)
  used:         0 us
  available:    460.8 ms
  limit:        460.8 ms

producers:     <not voted>
```