# 安装和部署（secretnote方式）

0、安装：

pip install secretnote -i <https://pypi.tuna.tsinghua.edu.cn/simple>

…/scql$ git clone https://github.com/secretflow/scql.git . (0.8.1b1)

1、在scql/examples/p2p-tutorial执行bash setup.sh

生成alice和bob的公钥、私钥

2、然后执行docker compose -p tk-sn up -d

-p是指定项目名称用的，-d是后台启动

查看容器集群的状态：

docker compose -p tk-sn ps

如果mysql的STATUS为healthy就好了

3、检查端口号：

cat .env

看到如下输出：

ALICE\_PORT=8081

BOB\_PORT=8082

SCQL\_IMAGE=secretflow/scql:latest

4、启动alice

执行secretnote --mode=scql --party=alice --host=http://127.0.0.1:8081

终端看到了这个：<http://localhost:8888/secretnote/scql>

浏览器输入：<http://xx.yy.zz.ww:8888/secretnote/scql/>，看到用户为alice

（

区分一下secretflow：[http:// xx.yy.zz.ww:8888/secretnote/secretflow](http://172.25.172.56:8888/secretnote/secretflow)

这里也有scql，但是没有用户

）

5、启用bob

secretnote --mode=scql --party=bob --host=http://127.0.0.1:8082

输出：<http://127.0.0.1:8889/secretnote/scql>

注意：4、5中Secretnote的启动设定中的host，必须和3给的端口一致，否则项目无法创建

# 使用SCQL(基于brokerctl工具)

需要完成3.1中的1-3

1. 构建brokerctl:

…/scql$ go build -o brokerctl cmd/brokerctl/main.go

1. 创建项目：

./brokerctl create project --project-id "demo" --host <http://localhost:8081>

1. 查看项目：Members只有alice

./brokerctl get project --host <http://localhost:8081>

1. 添加项目参与方
2. alice邀请bob参与项目

./brokerctl invite bob --project-id "demo" --host http://localhost:8081

1. Bob查看invite：

./brokerctl get invitation --host <http://localhost:8082>

1. Bob加入：

./brokerctl process invitation 1 --response "accept" --project-id "demo" --host <http://localhost:8082>

1. 检查参与方：Members出现alice和bob

./brokerctl get project --host <http://localhost:8081>

1. 创建数据表
2. Alice数据初始化

./brokerctl create table ta --project-id "demo" --columns "ID string, credit\_rank int, income int, age int" --ref-table alice.user\_credit --db-type mysql --host <http://localhost:8081>

1. 查看alice数据

./brokerctl get table ta --host http://localhost:8081 --project-id "demo"

1. 创建bob数据

./brokerctl create table tb --project-id "demo" --columns "ID string, order\_amount double, is\_active int" --ref-table bob.user\_stats --db-type mysql --host http://localhost:8082

1. 检查bob数据

./brokerctl get table tb --host http://localhost:8082 --project-id "demo"

1. CCL授权
2. Alice操作自己的数据表

./brokerctl grant alice PLAINTEXT --project-id "demo" --table-name ta --column-name ID --host http://localhost:8081

./brokerctl grant alice PLAINTEXT --project-id "demo" --table-name ta --column-name credit\_rank --host http://localhost:8081

./brokerctl grant alice PLAINTEXT --project-id "demo" --table-name ta --column-name income --host http://localhost:8081

./brokerctl grant alice PLAINTEXT --project-id "demo" --table-name ta --column-name age --host http://localhost:8081

1. Alice设置bob访问自己表的权限

./brokerctl grant bob PLAINTEXT\_AFTER\_JOIN --project-id "demo" --table-name ta --column-name ID --host http://localhost:8081

./brokerctl grant bob PLAINTEXT\_AFTER\_GROUP\_BY --project-id "demo" --table-name ta --column-name credit\_rank --host http://localhost:8081

./brokerctl grant bob PLAINTEXT\_AFTER\_AGGREGATE --project-id "demo" --table-name ta --column-name income --host http://localhost:8081

./brokerctl grant bob PLAINTEXT\_AFTER\_COMPARE --project-id "demo" --table-name ta --column-name age --host http://localhost:8081

1. Bob操作自己的表

./brokerctl grant bob PLAINTEXT --project-id "demo" --table-name tb --column-name ID --host http://localhost:8082

./brokerctl grant bob PLAINTEXT --project-id "demo" --table-name tb --column-name order\_amount --host http://localhost:8082

./brokerctl grant bob PLAINTEXT --project-id "demo" --table-name tb --column-name is\_active --host http://localhost:8082

1. Bob设置alice访问权限

./brokerctl grant alice PLAINTEXT\_AFTER\_JOIN --project-id "demo" --table-name tb --column-name ID --host http://localhost:8082

./brokerctl grant alice PLAINTEXT\_AFTER\_COMPARE --project-id "demo" --table-name tb --column-name is\_active --host http://localhost:8082

./brokerctl grant alice PLAINTEXT\_AFTER\_AGGREGATE --project-id "demo" --table-name tb --column-name order\_amount --host http://localhost:8082

1. 查看alice能访问的东西（自己表直接访问，另一方的数据操作后可以访问）

./brokerctl get ccl --project-id "demo" --parties alice --host <http://localhost:8081>

1. 查看bob可以访问的东西

./brokerctl get ccl --project-id "demo" --parties bob --host http://localhost:8081

1. 联合SQL查询：

./brokerctl run "SELECT ta.credit\_rank, COUNT(\*) as cnt, AVG(ta.income) as avg\_income, AVG(tb.order\_amount) as avg\_amount FROM ta INNER JOIN tb ON ta.ID = tb.ID WHERE ta.age >= 20 AND ta.age <= 30 AND tb.is\_active=1 GROUP BY ta.credit\_rank;" --project-id "demo" --host http://localhost:8081 --timeout 3

结果如下：

2 rows in set: (0.200944026s)

+-------------+-----+-------------------+-------------------+

| credit\_rank | cnt | avg\_income | avg\_amount |

+-------------+-----+-------------------+-------------------+

| 5 | 6 | 18069.72427368164 | 7743.392951965332 |

| 6 | 4 | 336018.1408996582 | 5499.425048828125 |

+-------------+-----+-------------------+-------------------+

# 使用SCQL(基于secretnote)

需要完成3.1中的1-5

1. 创建项目：alice网页上点击add new project按钮，新建项目test
2. 邀请bob：点击alice右上角的+发送请求

在bob的主页右上角的铃铛会刷新出提示，点击accept

1. 添加alice数据：（需要在alice的界面添加）

表名称为ta，关联表alice.user\_credit，数据列ID string, credit\_rank int, income int, age int

1. 添加bob数据（需要在bob的界面添加）

表名tb，关联表bob.user\_stats，字段ID string, order\_amount double, is\_active int

1. Alice配置ccl, 给自己的全是plaintext，给bob的如下

ID after join

credit\_rank after group by

income after aggregate

age after compare

1. Bob配置ccl, 给alice的如下：

ID

order\_amount

is\_active

1. Alice端查询，新建一个notebook，执行下面三块代码：

select \* from ta

-- 成功输出

SELECT \* from tb

-- not visiable to alice

SELECT ta.credit\_rank, COUNT(\*) as cnt, AVG(ta.income) as avg\_income, AVG(tb.order\_amount) as avg\_amount FROM ta INNER JOIN tb ON ta.ID = tb.ID WHERE ta.age >= 20 AND ta.age <= 30 AND tb.is\_active=1 GROUP BY ta.credit\_rank;

1. 实验结果如下，和brokerctl方式的结果一样

