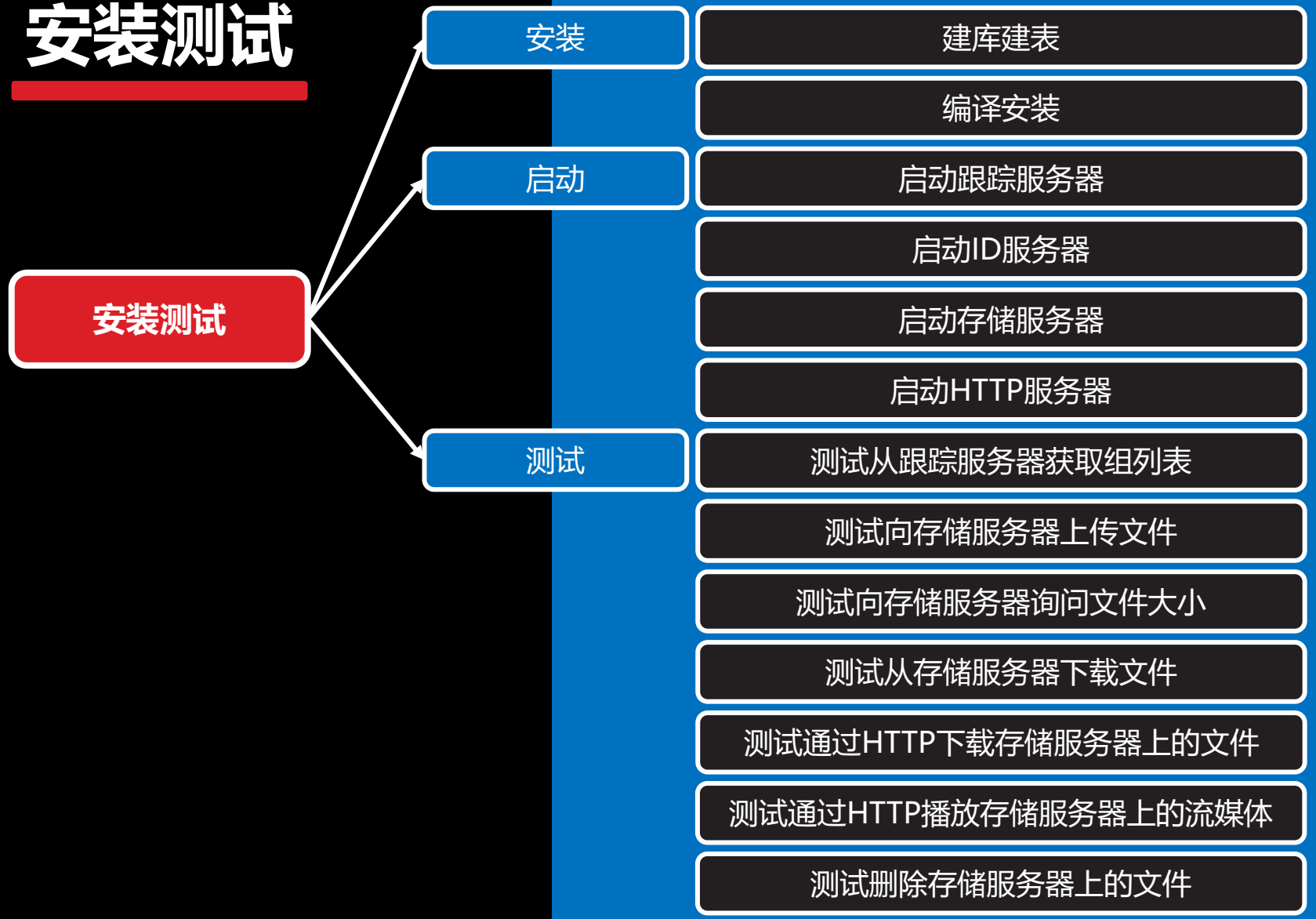
分布式流媒体

安装测试

Unit07

安装测试


安装测试
安装
建库建表
编译安装
启动
启动跟踪服务器
启动ID服务器
启动存储服务器
启动HTTP服务器
测试
测试从跟踪服务器获取组列表
测试向存储服务器上传文件
测试向存储服务器询问文件大小
测试从存储服务器下载文件
测试通过HTTP下载存储服务器上的文件
测试通过HTTP播放存储服务器上的流媒体
测试删除存储服务器上的文件

安装

建库建表

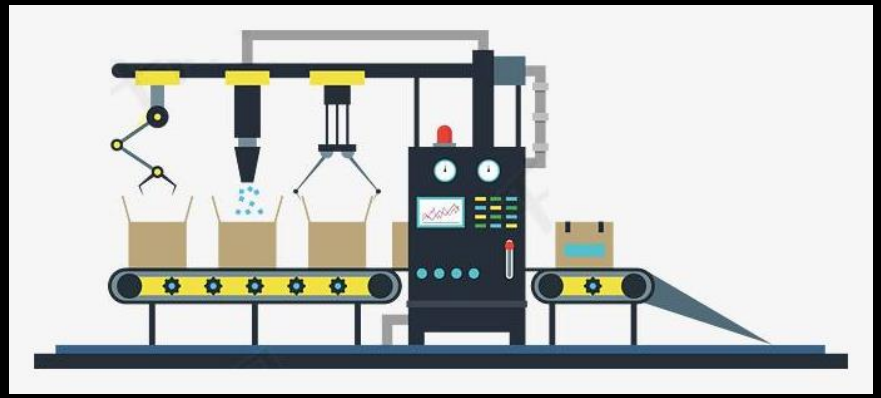
- 分别为跟踪服务器、ID服务器和存储服务器创建数据库tnv_trackerdb、tnv_idsdb和tnv_storagedb，并执行sql目录下的建表脚本，初始化表结构
  - $ cd ~/Projects/TNV/sql
    $ mysql -uroot -p123456
  - mysql> create database tnv_trackerdb;
    mysql> create database tnv_idsdb;
    mysql> create database tnv_storagedb;
    mysql> source tracker.sql
    mysql> source id.sql
    mysql> source storage.sql
    mysql> exit

知识讲解

编译安装

- 直接执行src目录下的安装脚本install，完成整个项目的编译、链接和安装
  - $ cd ~/Projects/TNV/src
    $ ./install

知识讲解

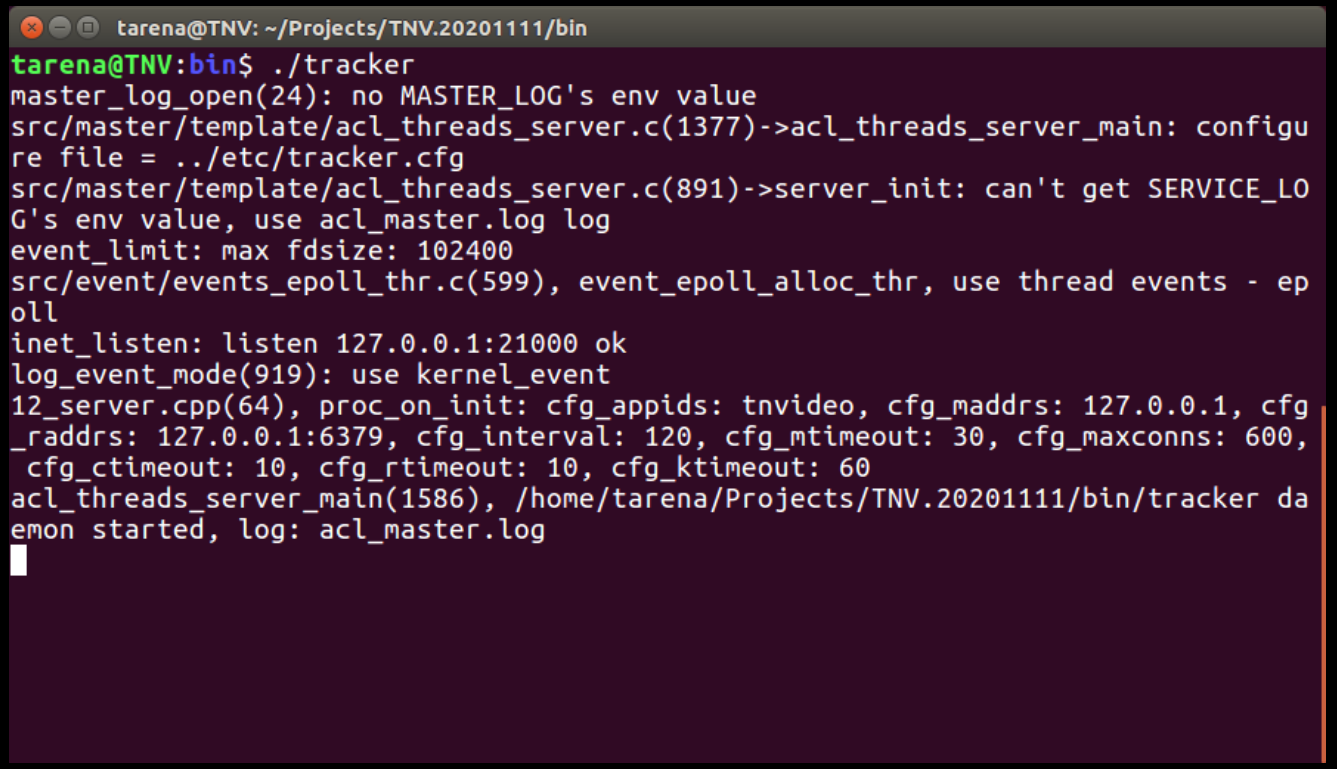


启动

启动跟踪服务器

- 直接执行bin目录下的tracker可执行程序
  - $ cd ~/Projects/TNV/bin
    $ ./tracker

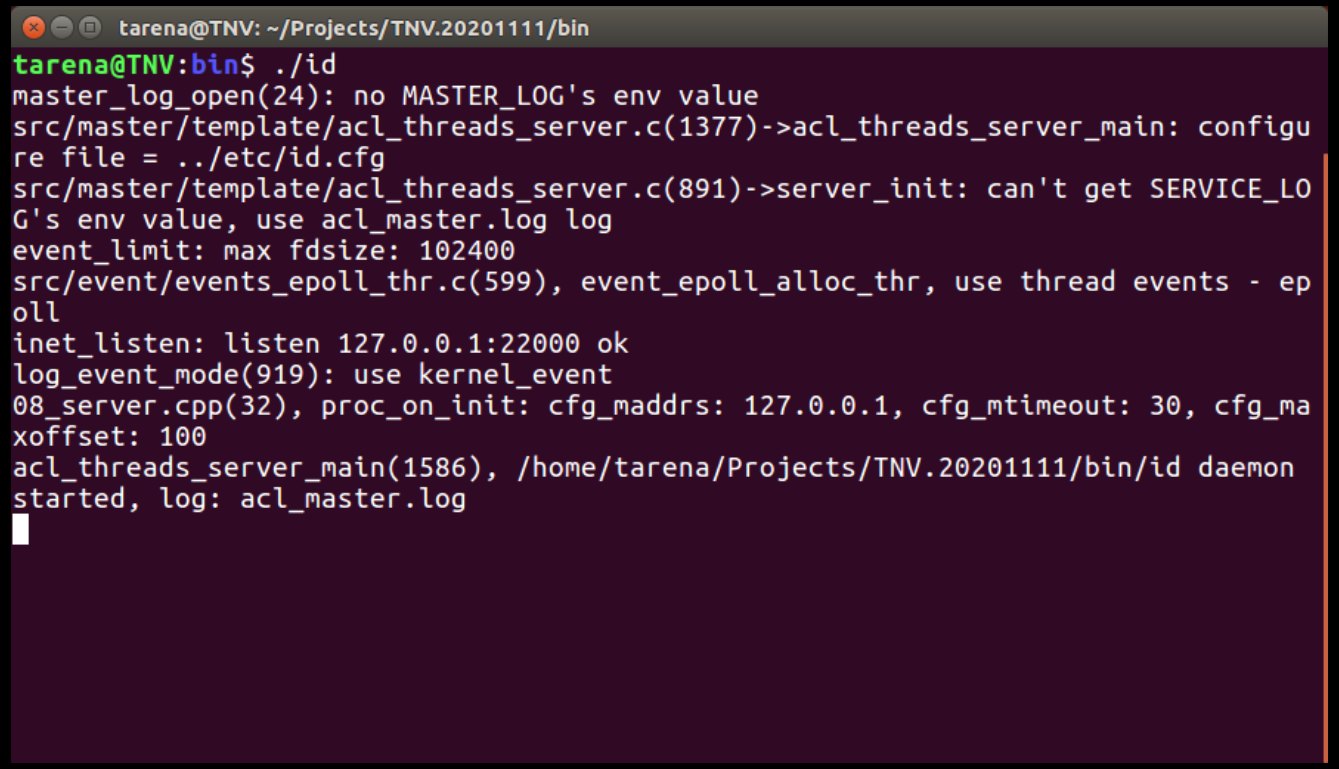
知识讲解



启动ID服务器

- 直接执行bin目录下的id可执行程序
  - $ cd ~/Projects/TNV/bin
    $ ./id

知识讲解



启动存储服务器

- 直接执行bin目录下的storage可执行程序
  - $ cd ~/Projects/TNV/bin
    $ ./storage

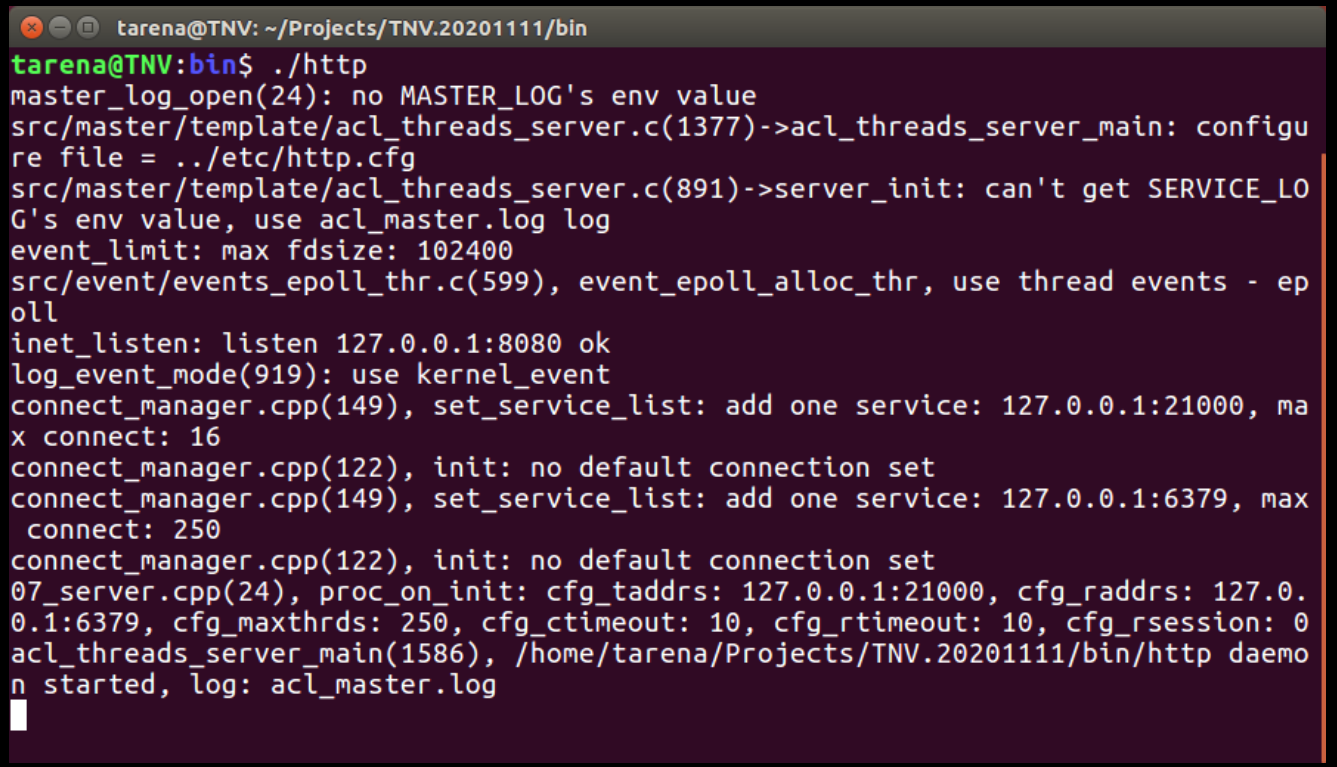
知识讲解

```
tarena@TNV: ~/Projects/TNV.20201111/bin
tarena@TNV:bin$ ./storage
master_log_open(24): no MASTER_LOG's env value
src/master/template/acl_threads_server.c(1377)->acl_threads_server_main: configu
re file = ../etc/storage.cfg
src/master/template/acl_threads_server.c(891)->server_init: can't get SERVICE_LO
G's env value, use acl_master.log log
event_limit: max fdsize: 102400
src/event/events_epoll_thr.c(599), event_epoll_alloc_thr, use thread events - ep
oll
inet_listen: listen 127.0.0.1:23000 ok
log_event_mode(919): use kernel_event
16_server.cpp(97), proc_on_init: cfg_gpname: group001, cfg_spaths: ../data, cfg_
taddrs: 127.0.0.1:21000, cfg_iaddrs: 127.0.0.1:22000, cfg_maddrs: 127.0.0.1, cfg
_raddrs: 127.0.0.1:6379, cfg_bindport: 23000, cfg_interval: 10, cfg_mtimeout: 30
, cfg_maxconns: 600, cfg_ctimeout: 10, cfg_rtimeout: 10, cfg_ktimeout: 60
acl_threads_server_main(1586), /home/tarena/Projects/TNV.20201111/bin/storage da
emon started, log: acl_master.log
14_tracker.cpp(101), join: bodylen: 0, command: 100, status: 0
14_tracker.cpp(178), beat: bodylen: 0, command: 100, status: 0
14_tracker.cpp(178), beat: bodylen: 0, command: 100, status: 0
14_tracker.cpp(178), beat: bodylen: 0, command: 100, status: 0
14_tracker.cpp(178), beat: bodylen: 0, command: 100, status: 0
```

启动HTTP服务器

- 直接执行bin目录下的http可执行程序
  - $ cd ~/Projects/TNV/bin
    $ ./http

知识讲解

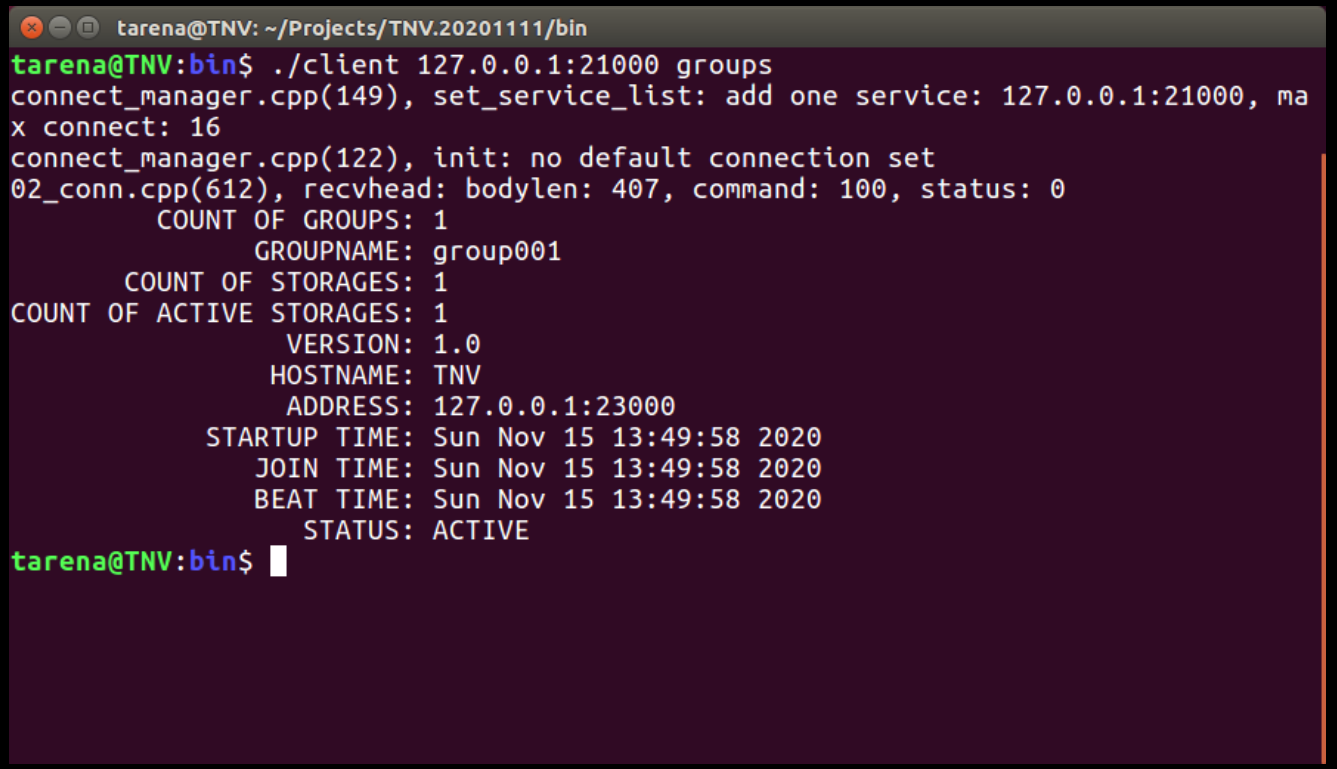


测试

测试从跟踪服务器获取组列表

- 执行如下命令行
  - $ cd ~/Projects/TNV/bin
    $ ./client 127.0.0.1:21000 groups

知识讲解



测试向存储服务器上传文件

- 执行如下命令行
  - $ cd ~/Projects/TNV/bin
    $ ./client 127.0.0.1:21000 upload tnvideo tnv001 ../upload/001.mp4

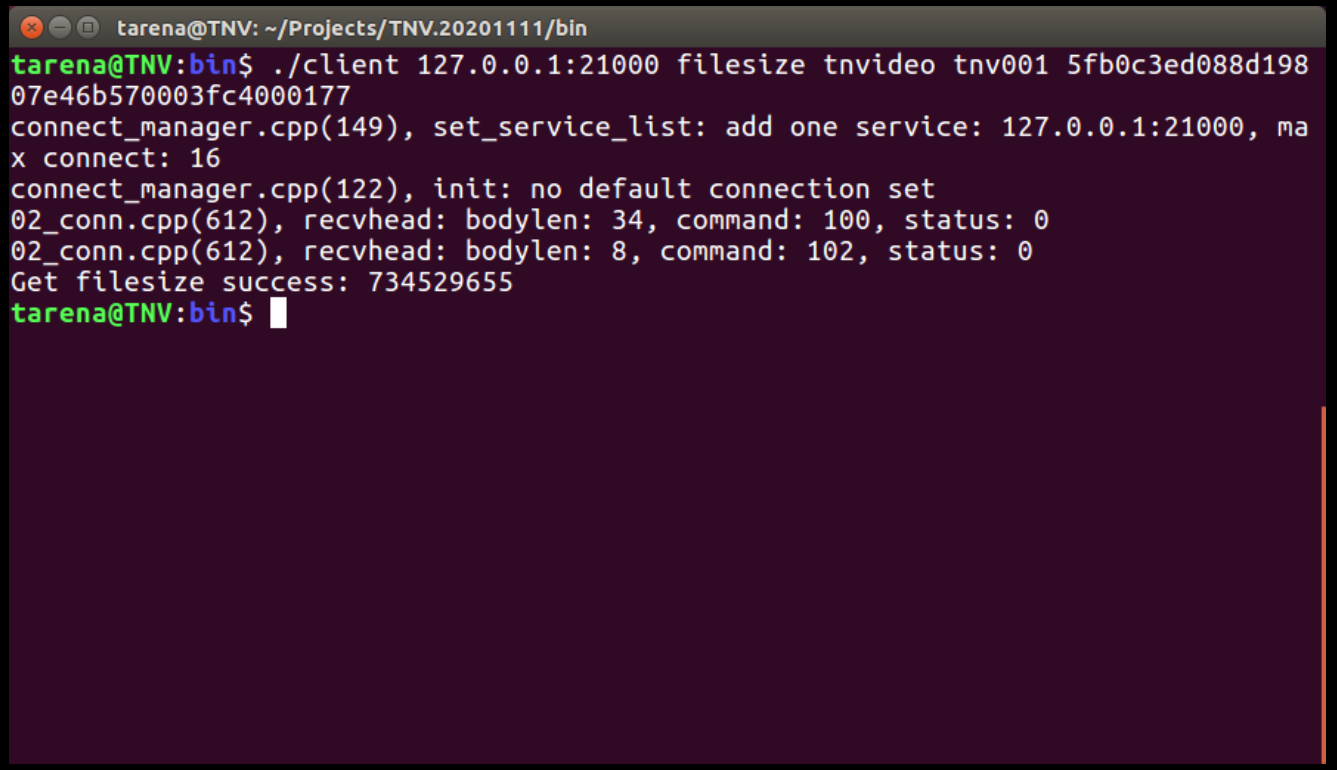
知识讲解

测试向存储服务器询问文件大小

- 执行如下命令行
  - $ cd ~/Projects/TNV/bin

    $ ./client 127.0.0.1:21000 filesize tnvideo tnv001 5fb0c3ed088d19807e46b570003fc4000177

知识讲解

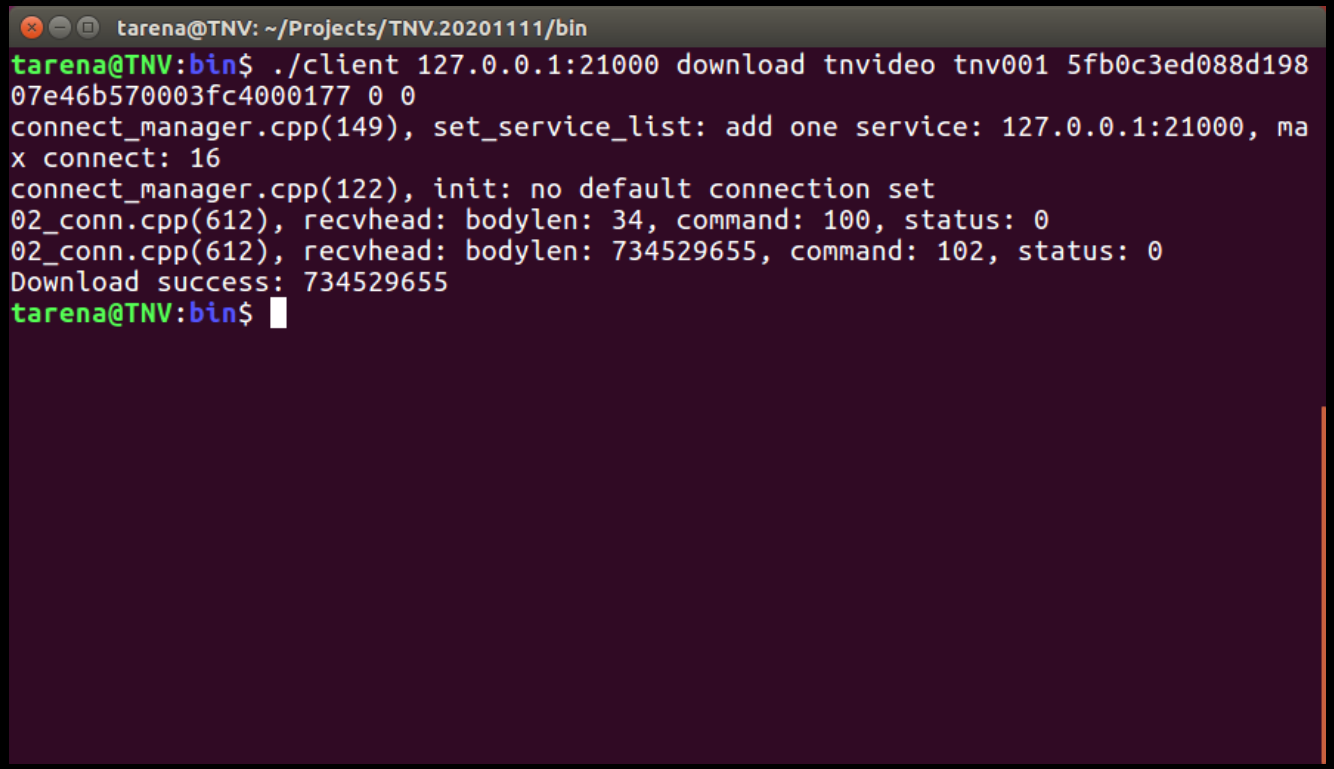


测试从存储服务器下载文件

- 执行如下命令行
  - $ cd ~/Projects/TNV/bin

    $ ./client 127.0.0.1:21000 download tnvideo tnv001 5fb0c3ed088d19807e46b570003fc4000177 0 0

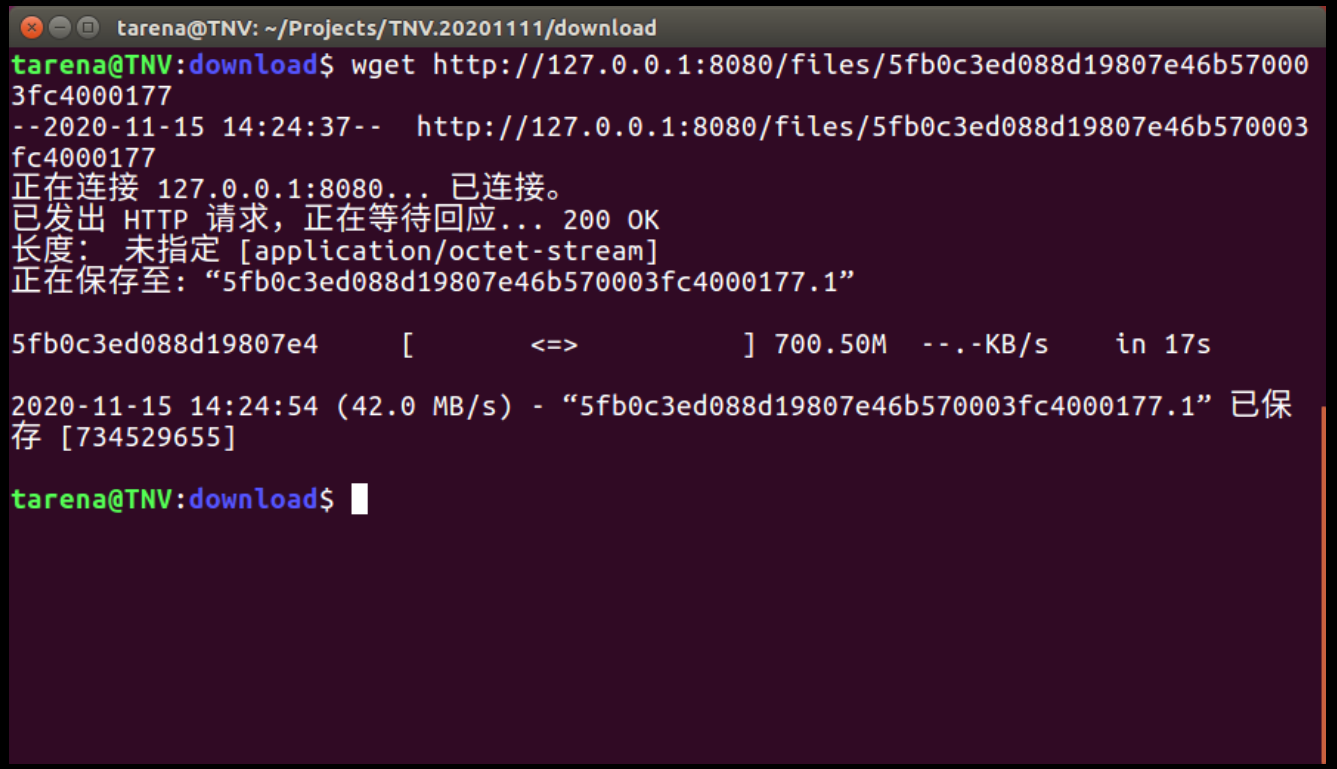
知识讲解



测试通过HTTP下载存储服务器上的文件

- 执行如下命令行
  - $ cd ~/Projects/download
    $ wget http://127.0.0.1:8080/files/5fb0c3ed088d19807e46b570003fc4000177

知识讲解



测试通过HTTP播放存储服务器上的流媒体

- 在QtPlayer中打开如下URL
  - http://127.0.0.1:8080/files/5fb0c3ed088d19807e46b570003fc4000177

知识讲解

测试删除存储服务器上的文件

- 执行如下命令行
  - $ cd ~/Projects/TNV/bin

    $ ./client 127.0.0.1:21000 delete tnvideo tnv001 5fb0c3ed088d19807e46b570003fc4000177

知识讲解

总结和答疑