

ISCSI and Libvirt

Clémentine Hayat <clem@lse.epita.fr>



GSOC Libvirt





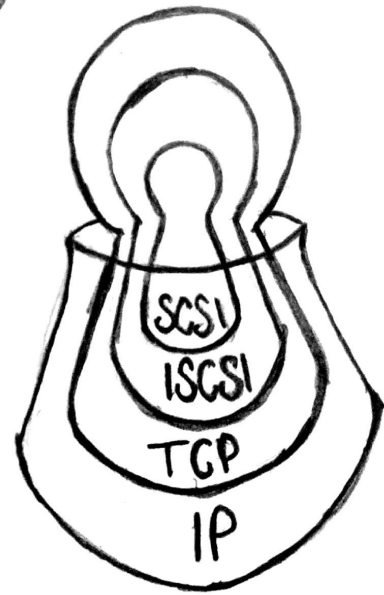
What is iSCSI?



SCSI standard

- Command
- Bus
- Protocol...





```

struct scsi_task *
iscsi_read12_task(struct iscsi_context *iscsi, int lun, uint32_t lba,
                 uint32_t datalen, int blocksize,
                 int rdprotect, int dpo, int fua, int fua_nv, int group_number,
                 iscsi_command_cb cb, void *private_data)
{
    struct scsi_task *task;
    if (datalen % blocksize != 0) {
        iscsi_set_error(iscsi, "Datalen:%d is not a multiple of "
                       "the blocksize:%d.", datalen, blocksize);
        return NULL;
    }
    task = scsi_cdb_read12(lba, datalen, blocksize, rdprotect,
                          dpo, fua, fua_nv, group_number);
    if (task == NULL) {
        iscsi_set_error(iscsi, "Out-of-memory: Failed to create "
                       "read12 cdb.");
        return NULL;
    }
    if (iscsi_scsi_command_async(iscsi, lun, task, cb,
                                NULL, private_data) != 0) {
        scsi_free_scsi_task(task);
        return NULL;
    }
    return task;
}

```





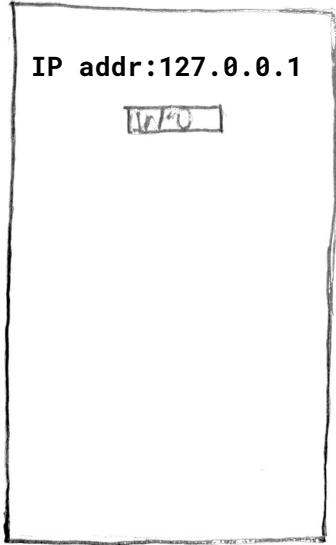
How does iSCSI works?



Discovery Session



Hey! Who is in there?



My IQN is:

`iqn.2003-01.org.linux-iscsi.clem.x8664:sn.f39f01319546`



```

void
discovery(struct iscsi_context *iscsi_context, const char *portal)
{
    if (iscsi_connect_sync(iscsi_context, portal) != 0) {
        printf("iscsi_connect failed. %s\n", iscsi_get_error(iscsi_context));
        exit(1);
    }
    printf("connected, send login command\n");
    iscsi_set_session_type(iscsi_context, ISCSI_SESSION_DISCOVERY);
    if (iscsi_login_sync(iscsi_context) != 0) {
        printf("iscsi_login failed : %s\n", iscsi_get_error(iscsi_context));
        exit(1);
    }
    printf("Logged in to target, send discovery command\n");
    struct iscsi_discovery_address *addr = iscsi_discovery_sync(iscsi_context);
    if (!addr) {
        printf("failed to send discovery command : %s\n", iscsi_get_error(iscsi_context));
        exit(1);
    }
    printf("discovery complete, send logout command\n");
    if (iscsi_logout_sync(iscsi_context) != 0) {
        printf("iscsi_logout failed : %s\n", iscsi_get_error(iscsi_context));
        exit(1);
    }
    printf("disconnect socket\n");
    if (iscsi_disconnect(iscsi_context) != 0) {
        printf("Failed to disconnect old socket\n");
        exit(1);
    }
}

```



Normal Session

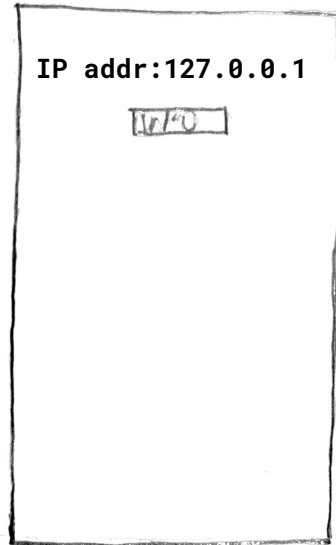


Hey! My iqn is initiator_iqn.

My user id is user.

My pass is pass.

Can I go see target_iqn?



Yes you're allowed to!



```

void
normallogin(struct iscsi_context *iscsi_context, char *user, char *passwd, struct client_state *client_state)
{
    printf("Reconnect with normal login to [%s]\n", client_state->target_address);
    printf("Use targetname [%s] when connecting\n", client_state->target_name);
    if (iscsi_set_targetname(iscsi_context, client_state->target_name)) {
        printf("Failed to set target name\n");
        exit(1);
    }
    if (iscsi_set_session_type(iscsi_context, ISCSI_SESSION_NORMAL) != 0) {
        printf("Failed to set session type to normal\n");
        exit(1);
    }
    if (iscsi_connect_sync(iscsi_context, client_state->target_address) != 0) {
        printf("iscsi_connect failed : %s\n", iscsi_get_error(iscsi_context));
        exit(1);
    }
    printf("connected, send login command\n");
    iscsi_set_target_username_pwd(iscsi_context, user, passwd);
    if (iscsi_login_sync(iscsi_context) != 0) {
        printf("iscsi_login failed\n");
        exit(1);
    }
}
}

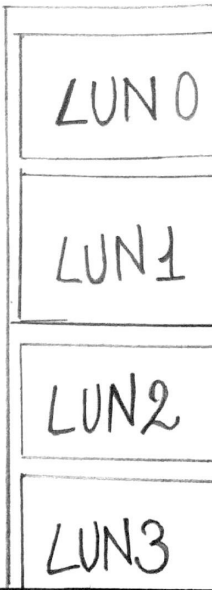
```



Hello!
So what are your storages?



Hey! I have :
Lun0, Lun1... LunN



```

void
reportluns(struct iscsi_context *iscsi_context, struct client_state *client_state)
{
    struct scsi_reportluns_list *list;
    int full_report_size;

    printf("Logged in normal session, send reportluns\n");
    struct scsi_task *scsi_task = iscsi_reportluns_sync(iscsi_context, 0, 16);
    /* ... */
    if (scsi_task->status != SCSI_STATUS_GOOD) { /* ... */}
    full_report_size = scsi_datain_getfullsize(scsi_task);
    if (full_report_size > scsi_task->datain.size) {
        printf("We did not get all the data we need in reportluns, ask again\n");
        scsi_free_scsi_task(scsi_task);
        scsi_task = iscsi_reportluns_sync(iscsi_context, 0, full_report_size);
        if (!scsi_task) {
            printf("failed to send reportluns command\n");
            scsi_free_scsi_task(scsi_task);
            exit(10);
        }
    }
    list = scsi_datain_unmarshall(scsi_task);
    if (!list) { /* ... */}
    printluns(list, client_state);
    scsi_free_scsi_task(scsi_task);
}

```



*I'm gonna command on
lun x then.*



It worked!



LUN0

LUN1

LUN2

LUN3




```

int
testUnitReady(struct iscsi_context *iscsi, int lun)
{
    do {
        if (!(task = iscsi_testunitready_sync(iscsi, lun))) {
            /*...*/
        }

        if (task->status != SCSI_STATUS_CHECK_CONDITION ||
            task->sense.key != SCSI_SENSE_UNIT_ATTENTION ||
            task->sense.ascq != SCSI_SENSE_ASCQ_BUS_RESET)
            break;
        scsi_free_scsi_task(task);
    } while (1);

    if (task->status != SCSI_STATUS_GOOD) {
        virReportError(VIR_ERR_INTERNAL_ERROR,
                      _("Failed testunitready: %s"),
                      iscsi_get_error(iscsi));
        goto cleanup;
    }
    ret = 0;
    /*...*/
}

```



```

if (!(task = iscsi_inquiry_sync(ic, lun, 0, 0, 64)) ||
    task->status != SCSI_STATUS_GOOD) { }
if (!(inq = scsi_datain_unmarshall(task))) { }

printf("inquiry returned type=%d vendor=%s product=%s\n",
    inq->device_type, inq->vendor_identification, inq->product_identification);
scsi_free_scsi_task(task);

if (type == SCSI_INQUIRY_PERIPHERAL_DEVICE_TYPE_DIRECT_ACCESS) {
    struct scsi_readcapacity10 *rc10 = NULL;

    if (!(task = iscsi_readcapacity10_sync(ic, lun, 0, 0)) ||
        task->status != SCSI_STATUS_GOOD) { }

    if (!(rc10 = scsi_datain_unmarshall(task))) { }

    printf("readcapacity10 returned lba=%lu block_size=%lu\n",
        (unsigned long) rc10->lba,
        (unsigned long) rc10->block_size);

    size = rc10->block_size;
    size *= rc10->lba;
}

```



```
Initiator => target (ip) -  
-lun0  
-iqn1 -lun1  
-lun0  
-iqn2 -lun1  
-lun2
```





How does libvirt work?



- Storage pool
- Domain
- XML





**What already has
libvirt?**



- Storage Pool using iscsiadm
 - And libiscsi
 - And iscsiadm





What is my Gsoc?



Storage Pool using libiscsi





U.S.E.R

KARNIVAL





**What have I
done?**



-storage pool -domain

-lun0

-iqn1

-lun1

Initiator => target (ip) -

-lun0

-iqn2

-lun1

-lun2



Add libiscsi to libvirt buildsystem



Storage Pool commands



XML d'exemple

```
<pool type='iscsi-direct'>  
  <name>remote-storage</name>  
  <source>  
    <host name='0.0.0.0'/>  
    <device path='iqn.2003-01.org.linux-iscsi.clem.x8664:sn.f39f01319546'/>  
    <initiator>  
      <iqn name='iqn.2005-03.org.open-iscsi:clem'/>  
    </initiator>  
  </source>  
</pool>
```





What's left to do?



Conclusion



Links

- <https://github.com/HCClem/libvirt>
- <https://github.com/HCClem/iSCSI-test>
- <https://github.com/sahlberg/libiscsi>



Questions?

