ТЗ Сервер для платежных терминалов (Linux)

1.1. Сервер открывает сокет, принимает по нему данные с платежных терминалов для бесконтактных банковских карт (см. Приложение 1). Затем отправляет их на хост банка (см. Приложение 2), принимает с хоста банка результат транзакции (успешно/не успешно, ID транзакции и т.д). Записывает данные транзакции в MySql (см. Приложение 3). Возвращает ответ терминалу.

Весь трафик защищается SSL
Весь трафик логируется в таблице PacketLog
Логи ошибок записывается в таблицу Log
С каждым клиентом (терминалом) нужно работать в отдельном потоке, обработка запросов должна быть полностью параллельна. Средняя нагрузка: 2 запроса в секунду. Пиковая: 30 запросов в секунду.

Принимать соединения клиентов по портам: 85 - для SSL, 82 - без SSL (для отладки)

Для тестирования кода будут предоставлены терминалы и доступ к банковскому хосту.

1.2. В случае успеха транзакции - сервер отправляет данные чека на сервер облачной кассы (nanokassa.ru), принимает ответ, записывает его в MySql.
Описание API:
https://nanokassa.ru/integration/documentation/
Пример формирования пакета:
https://nanokassa.ru/wa-data/public/site/files/send-simple-req-to-nanokassa.zip

Затем, сервер обращается к облачной кассе для получения идентификационных данных только что отправленного чека: Номер ФН (Фискальный Номер),Номер ФД (Фискальный документ), Номер ФПД (Фискальный Признак Документа, также известный как ФП). Затем, сервер возвращает эти данные на платежный терминал (чтобы он мог сформировать и выдать клиенту QR-код).
API для получения данных отправленного чека:
http://fp.nanokassa.com/getfp?nuid=712514037529915720722 43&qnuid=1572075985444617454936 9&auth=base
nuid и qnuid сервер облачной кассы вернет после формирования чека

1.3. Сервер будет работать одновременно на 2-х независимых и равноценных физических серверах. Терминалы могут подключаться к любому из них. Базы MySql должны синхронизироваться в реальном времени. При выходе из строя одного сервера - второй должен сохранять работоспособность. При восстановлении - база MySql должна автоматически синхронизироваться со вторым (рабочим) сервером.

Приложение 1. Протокол передачи данных Платежный терминал - Сервер

1. Протокол бинарный. Формат пакета:

Заголовок 1 (4 байта)		Заголовок 2 (7 байт)		Тело пакета	
Длина пакета (Заголовок 2 + Тело пакета) (2 байта)	CRC16 Заголовок 2 + Тело пакета (2 байта)	Мак-адрес клиента (6 байт)	Порядковый номер пакета (1 байт)	Тип пакета (1 байт)	Данные (до 1400 байт)

Функция вычисления CRC16 приведена ниже
Порядковый номер пакета начинается с 0. Затем 1, 2, 3 … 255, 0, 1….

2. После соединения сокета, клиент отправляет пакет Auth (0x01), в разделе Данные передаются номер версии прошивки (2 байта). Сервер проверяет в таблице Terminals, есть ли такой Мак, и, при наличии, отвечает таким же типом пакета Auth (0x01). Раздел Данные заполнить серверным временем (4 байта) в UnixTime. Если Мак в БД отсутствует - закрыть соединение, сделать запись в таблицу Log (в MySql).
Пример пакета Auth:
Запрос:
0A00 8781 C8FD190568EA 00 01 7310

Заголовок 1 (4 байта)		Заголовок 2 (7 байт)		Тело пакета (3 байта)	
0x0A 0x00 (dec - 10)	0x87 0x81	0xC8 0xFD 0x19 0x05 0x68 0xEA	0x00	0x01	0x73 0x10 (версия: 4211)

Ответ:
0C00 8B55 C8FD190568EA 00 01 0B99DE5D

Заголовок 1 (4 байта)		Заголовок 2 (7 байт)		Тело пакета (5 байт)	
0x0C 0x00 (dec - 12)	0x8B 0x55	0xC8 0xFD 0x19 0x05 0x68 0xEA	0x00	0x01	0x0B 0x99 0xDE 0x5D (27.11.19 18:40:59)

При успехе - добавлять запись в таблицу Sessions, запоминать SessionID.

После успешной аутентификации, клиент может отправить 3 типа пакета: Ping (0x02), Log (0x05), Purchase (0x13)

3. Пакет Ping
Тело пакета:

Тип паке	ClientTime - Время на устройстве в	PingMs - Значение последнего пинга

та 0x02	тиках (4 байта)	в миллисекундах (2 байта)
0x02	0x3F 0x00 0x00 0x00	0x64 0x00

В ответ сразу же вернуть клиенту этот же пакет

Тип пакета 0x02	ClientTime только что принятого от клиента пакета Ping (4 байта)
0x02	0x3F 0x00 0x00 0x00

Клиент, вычитая ClientTime из текущего времени, в момент приема пакета с сервера вычислит пинг.

4. Пример пакета Log:
   5A0065919884E34F9AAB0A05030301004D7561727442726964676 53A20312C2063616D3A20312C2070696E67536D61727470686F6E653A20302C206F6E654368616E6E656C5363616E3A20302C204841503A20302C2064656275673A2030

   Пояснение:
   5A00 CRC16 (Заголовок 1) 9884E34F9AAB 01 (Заголовок 2)
   05 (Тип пакета) 03 (LogPart) 03 (LogType) 0100 (LogCode) 4D (Длина текста - 77 (dec))
   7561727442726964676 53A20312C2063616D3A20312C2070696E67536D61727470686F6E653A20302C206F6E654368616E6E656C5363616E3A20302C204841503A20302C2064656275673A2030 (Текст лога: “uartBridge: 1, cam: 1, pingSmartphone: 0, oneChannelScan: 0, HAP: 0, debug: 0”)

   Записать в таблицу Log. Клиенту не отвечать.

5. Пакет Purchase
   Тело пакета:

Поле	**Тип данных (размер в байтах)**	**Пример**
Тип пакета		0x0D (13 - dec)
Сумма платежа	int (4)	0xFF 0x03 (1023 - dec, 10 рублей 23 копейки)

Размер поля “Реквизиты карты”	byte (1)	0x25 (37 - dec)
Реквизиты карты	char (см. выше)	34323937343930303034353730373332443231313032303133393431303030303030363536 (hex)
Размер поля “Криптограмма транзакции”	byte (1)	0x5F (95 - dec)
Криптограмма транзакции	char (см. выше)	5F2A0206435F3401018202200095050000000009A031910299C01009F02060000000142989F101706011103A000000F0300001600000000000003280B2159F1A0206439F2608D8EBDC1765169C9F9F2701809F360200399F3704E0110103 (hex)

При получении - отправить запрос на транзакцию в банковский хост (см. Приложение 2)
После получения ответа от банковского хоста - перенаправить ответ клиенту:

Тело пакета:

Поле	**Тип данных (размер в байтах)**	**Пример**
Тип пакета		0x0D (13 - dec)
Платеж прошел	bool (1)	0x01 (успех)
Размер поля “Расшифровка ошибки”	byte (1)	0x1D (29 - dec)
Расшифровка ошибки	char (см. выше)	“Недостаточно средств на счете”

Затем, после получения ответа от nanokassa, отправить на терминал команду показа QR-кода
Тело пакета:

Поле	Тип данных (размер в байтах)	Пример
Тип пакета		0x0E (14 - dec)
Размер поля “Текст QR”	byte (1)	0x48 (72 - dec)
Текст QR	char (см. выше)	“t=20180518T220500&s=975.88&fn=8710000101125654&i=99456&fp=1250448795&n=1” (t = время транзакции, s = сумма, fn = номер фискального накопителя, i = номер фискального документа, fp = фискальный признак, n всегда = 1)

Типы пакетов

```
enum PacketTypes {
    PType_Auth = 1,
    PType_Ping = 2,
    PType_SensorData = 3,
    PType_File = 4,
    PType_Log = 5,
    PType_NodePacket = 6,
    PType_BridgePacket = 7,
    PType_ReAuth = 8,
    PType_Cam = 9,
    PType_NodeFile = 10,
    PType_VendFile = 11,
    PType_RegisterBridge = 12,
    PType_Purchase = 13,
    PType_ShowQR = 14,
    PType_Count
};
```

Функции для вычисления CRC16

```
#define _u8  unsigned char
#define _u16 unsigned short

_u16 crcTable [256];

//---------------------------------------------------------------------------------------------------------------
void MakeCRC16Table()
{
  _u16 r;
  int s, s1;

  for(s = 0; s < 256; s++)
  {
    r = ((_u16)s)<<8;

    for (s1 = 0; s1 < 8; s1++)
    {
    if (r&(1<<15))
        r = (r<<1)^0x8005;
    else
        r = r<<1;
    }
     crcTable[s] = r;
   }
}

//---------------------------------------------------------------------------------------------------------------
_u16 GetCRC16(const _u8 *buf, _u16 len)
{
    _u16 crc;
    crc = 0xFFFF;
    while (len--)
    {
        crc = crcTable [((crc>>8)^*buf++)&0xFF] ^ (crc<<8);
    }
    crc ^= 0xFFFF;

    return crc;
}
```

Приложение 2. Протокол передачи данных Сервер - Хост банка

1. Из таблицы Terminals выбрать нужную запись по полю TerminalMac. Загрузить сертификаты SSL.
2. Выбрать из таблицы BpcHosts все записи, попытаться соединиться с первым хостом, используя SSL сертификаты п.1. При неудаче - перебрать следующие хосты.
3. Отправить пакет 0200 - запрос платежной транзакции:
   Пример:
   303232383032303032300580 20C0820030303030303030303030303134323938313032393131313131313030303030303131393130323931313131313130373032303030323337343239373439303030343537303733324432313130323031333934313030303030303635366270633303032374C754C75204D455243482020202020363433009 55F2A0206435F3401018202200950500000000009A031910299C01009F02060000000142989F101706011103A000000F03000160000000000000003280B2159F1A0206439F2608D8EBDC1765169C9F9F2701809F360200399F3704E0110103

   Расшифровка:
   Поля, отмеченные
   желтым - константы, их значение всегда одинаково
   синим - берутся из пакета терминала
   красным - из таблицы Terminals

Поле	Тип данных (размер в байтах)	Пример (в ASCII, если не указано иное)
Размер пакета	char [4]	0228 = 30323238 (hex)
MTI	char[4]	0200
BITMAP	byte[8]	3230058020C08200 (hex)
Processing code	char[6]	000000
Сумма платежа	char[12]	000000014298 (142 рубля 98 копеек)
Время передачи	char[10]	1029111111 (текущее время в формате MMDDhhmmss)
SYSTEM TRACE NO.	char[6]	000001

Время транзакции карты	char[12]	191029111111 (текущее время в формате YYMMDDhhmmss)
POS ENTRY MODE	char[3]	070
FUNCTION CODE	char[3]	200
POS CONDITION CODE	char[2]	02
Размер поля “Реквизиты карты”	char[2]	37
Реквизиты карты	char (var)	4297490004570732D21102013941000000656
TERMINAL ID	char[8]	bpc00027
MERCHANT ID	char[15]	LuLu MERCH
CURRENCY CODE	char[3]	643
Размер поля “EMV Data”	byte (2)	00 95 (hex) = 95. Формат: BCD
EMV Data	byte(var)	5F2A0206435F3401018202200095050000000009A031910299C01009F02060000000142989F101706011103A000000F0300001600000000000003280B2159F1A0206439F2608D8EBDC1765169C9F9F2701809F360200399F3704E0110103

4. Принять ответный пакет с хоста банка:

Пример:

303130323032313072300000A8080003136343239373439303030343537303733323030303030303030303030303030313432393831313239303634303333330303030303131393130323931313131313130303031313333330303733335303035627063303030323736343 3

Поля, отмеченные

желтым - игнорировать

синим - нужные данные

Поле	Тип данных (размер в байтах)	Пример (в ASCII, если не указано иное)
Размер пакета	char [4]	0102 = 30313032 (hex)
MTI	char[4]	0210. Если не 0210 - ошибка
BITMAP	byte[8]	723000000A808000 (hex)
Длина поля PAN	char[2]	16
PAN (номер банковской карты)	char[var]	4297490004570732
Processing code	char[6]	000000
Сумма платежа	char[12]	000000014298 (142 рубля 98 копеек)
Время передачи	char[10]	1129064033 (в формате MMDDhhmmss)
SYSTEM TRACE NO.	char[6]	000001
Время транзакции карты	char[12]	191029111111 (в формате YYMMDDhhmmss)
RRN	char[12]	000113300735
Номер подтверждения*	char[6]	000000
Код ответа (результат)	char[3]	000
TERMINAL ID	char[8]	bpc00027
CURRENCY CODE	char[3]	643

*В случае неуспешной транзакции это поле будет отсутствовать. Узнать это можно проверив бит №38 поля BITMAP. Если = 1 - присутствует, если = 0 - отсутствует. Использовать этот бит для определения успешна транзакция или нет

Записать в таблицу Transactions: Сумму платежа, RRN, Номер подтверждения, Код ответа, TerminalID, Мак терминала, Текущее время, Результат транзакции (1 - успешно, 0 - не успешно)

Если транзакция не успешна - выбрать из таблицы ResponseDescriptions текст ответа клиенту (по Коду ответа).

Рабочий код, который делает все вышеперечисленное на тестовых данных (без записи в MySql):
https://github.com/sphinxgames/ClientServer/blob/master/SSLTest.c

Приложение 3. Структура БД MySql

1. Таблица Terminals

Поле	Тип	Описание
ID	INTEGER	
TerminalMac	INTEGER	Мак-адрес терминала, в форме числа 8 байт
SSLCert	VARCHAR	Сертификат SSL для подключения к банковскому хосту
SSLKey	VARCHAR	Закрытый ключ SSL для подключения к банковскому хосту
TerminalID	VARCHAR	для формирования пакета хосту банка
MerchantID	VARCHAR	для формирования пакета хосту банка
Version	INTEGER	Версия прошивки. Обновлять при Auth
Ping	INTEGER	Обновлять при Ping
LastOnlineTime	DATETIME	Обновлять при любом пакете
OwnerID	INTEGER	Владелец терминала (Юр. лицо)

2. Таблица Owners

Поле	Тип	Описание
ID	INTEGER	
Name	VARCHAR	Название клиента

3. Таблица Log

Поле	Тип	Описание
SessionID	INTEGER	ID клиентской сессии, во время которой произошла запись в лог
Time	DATETIME	
LogPart	INTEGER	Раздел логов, источник записи (Клиент, Сервер)
LogType	INTEGER	Уведомление/ Ошибка / Критическая ошибка
LogCode	INTEGER	Код ошибки
Text	VARCHAR	Содержание записи

4. Таблица Sessions

Поле	Тип	Описание
ID	INTEGER	ID клиентской сессии
IP	INTEGER	IP-адрес клиента
TerminalMac	INTEGER	Мак-адрес подключившегося терминала
TimeStart	DATETIME	Время открытия сессии
TimeEnd	DATETIME	Время закрытия сессии

5. Таблица PacketLog

Поле	Тип	Описание
ID	INTEGER	
SessionID	INTEGER	ID клиентской сессии
Type	INTEGER	Тип пакета
Time	DATETIME	Время приема/отправки пакета
Direction	INTEGER	0 - от клиента серверу. 1 - от сервера клиенту
Data	Blob	Содержимое пакета

6. Таблица BpcHosts

Поле	Тип	Описание
ID	INTEGER	
IP	INTEGER	
Port	INTEGER	

7. Таблица ResponseDescriptions

Поле	Тип	Описание
ResponseCode	INTEGER	
Description	VARCHAR	Описание ошибки

8. Таблица Transactions

Поле	Тип	Описание

ID	INTEGER	
Summ	FLOAT	Сумма платежа
RRN	VARCHAR	из ответа хоста банка
ApprovalNumber	VARCHAR	из ответа хоста банка
ResponseCode	INTEGER	код ответа хоста банка
TerminalID	VARCHAR	из ответа хоста банка
TerminalMac	INTEGER	Terminals.TerminalMac
Time	DATETIME	Время добавления записи (NOW())
Result	INTEGER	1- успех. 0 - отказ
Kassa_fn	VARCHAR	Номер фискального накопителя (от nanokassa)
Kassa_i	VARCHAR	Номер фискального документа (от nanokassa)
Kassa_fd	VARCHAR	Фискальный признак (от nanokassa)