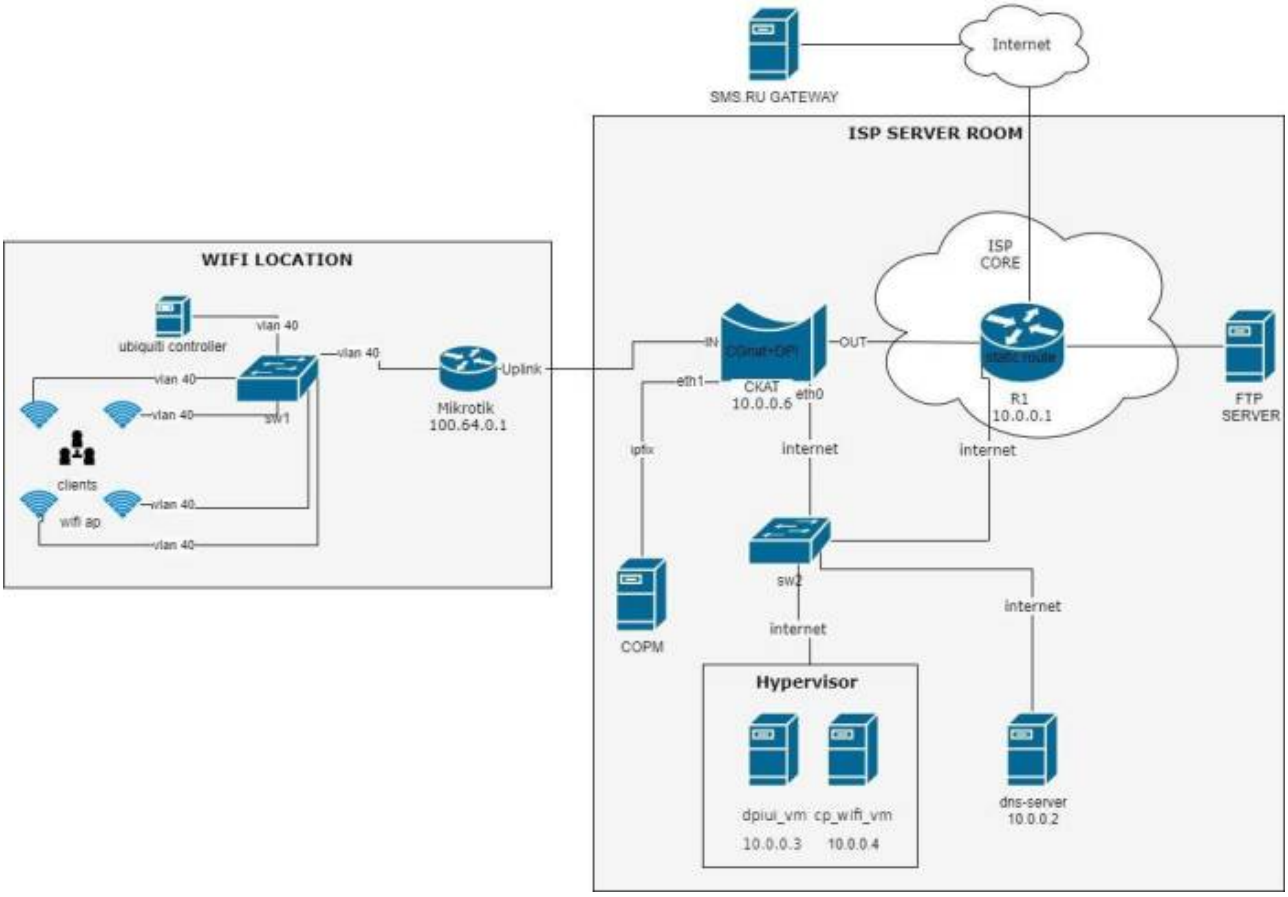
Содержание

Настройка GUI, CKAT и WiFi HotSpot при включенном менеджменте сессий

Топология сети

1. Подключить оборудование согласно топологии сети.



Последовательность действий при авторизации

1. Абонент подключается к сети WiFi
2. Появляется welcome page с информацией, что абонент должен открыть браузер и идентифицировать себя
3. Абонент отрывает браузер, при переходе на любой URL происходит переадресация абонента на страницу идентификации
4. Абонент вводит телефонный номер, запрашивает код доступа
5. Код доступа отправляется на телефонный номер через SMS
6. Абонент вводит полученный код доступа
7. Происходит запись сессионных куки на абонентское устройство с сохранением заданного периода, а также переход на запрошенный пользователем URL

Установка виртуальных машин (ВМ)

1. Создать две виртуальные машины со следующими минимальными характеристиками:
   - ВМ `dpiui_vm` – 1 cpu, 2Gb Ram, 50G hard disk, Guest OS Centos 7, nic 1
   - ВМ `cp_wifi_vm` – 1 cpu, 1Gb Ram, 30G hard disk, Guest OS Centos 7, nic
2. Устанавить последнюю версию CentOS 7 (build-2009 на момент написания) на обе виртуальные машины. По ходу установки выбрать минимальную установку (minimal install).
   После установки ОС открыть консоль и установить пакеты на обе ВМ: сначала

   ```
   yum install epel-release
   ```

   и после:

   ```
   yum install nano tcpdump openssh-server openssh-clients
   ```

3. Выключить selinux на обоих ВМ:
   - Отредактировать файл `/etc/sysconfig/selinux`
   - Задать значение параметра `SELINUX=disabled` и перезагрузить ВМ

Установка и настройка dpiui_vm

1. Установить DPIUI на `dpiui_vm` по инструкции
2. Настроить сеть на обоих ВМ и СКАТ:

   ```
   BOOTPROTO=static
   ONBOOT=yes
   IPADDR=10.0.0.x
   NETMASK=255.255.255.0
   GATEWAY=10.0.0.1
   DNS1=10.0.0.2
   ```

   IPADDR — указать для каждого хоста согласно схеме (или использовать свою адресацию).

3. Войти в GUI и добавить обе ВМ и СКАТ в разделе «ОБОРУДОВАНИЕ», следуя инструкции:

Установка и настройка cp_wifi_vm

1. Установить пакет `wifi_hotspot` на ВМ `cp_wifi_vm` по инструкции:
2. Отредактировать конфиг файл для Hotspot:

   ```
   nano /var/www/html/wifi_hotspot/backend/.env
   ```

   Изменить/добавить только эти строчки:

   1. **`AAA_HOTSPOT_IP` – 10.0.0.4**
      Адрес NAS сервера, IPv4/IPv6, если неизвестно – 0.0.0.0

2. **AAA_HOTSPOT_PORT – 0**
   Порт NAS сервера, число, если неизвестно - 0
3. **AAA_HOTSPOT_ID – 2**
   ИД точки подключения к сети передачи данных, целое число в диапазоне от 0 до 1000, необходимо заполнять для абонентов публичных Wifi-точек, соответствует идентификатору точки подключения в поле 1 из выгрузки точек подключения
4. **AAA_EXPORT_ENABLED=1**
   Включить экспорт AAA
5. **AUTH_CODE_LENGTH=4**
   Поменять кол-во символов в коде для SMS авторизации

Если прописали параметр AUTH_CODE_LENGTH, то надо в файле /var/www/html/wifi_hotspot/frontend/env.js установить значение:

```
AppEnv.AuthCodePlaceHolder = "0000";
```

В конце выполнить команду:

```
php /var/www/html/wifi_hotspot/backend/artisan queue:restart
```

Установка и настройка dhcp-isc на cp_wifi_vm

1. Установить пакет dhcp-isc — yum install dhcp expect
2. Далее настраиваем скрипты статического arp и конфиг файл dhcpd.conf:
   - Сначала конфиг файл dhcpd — nano /etc/dhcp/dhcpd.conf
     Поставить свои значения option domain-name и option ntp-servers!

```
ddns-update-style none;
authoritative;
db-time-format local;


log-facility local7;


subnet 100.64.0.0 netmask 255.255.252.0 {
  range 100.64.0.3 100.64.3.254;
  default-lease-time 600;
  max-lease-time 600;
  option subnet-mask 255.255.252.0;
  option broadcast-address 100.64.3.255;
  option routers 100.64.0.1;
  option ntp-servers <ntp-server>;
  option domain-name-servers 10.0.0.2;
  option domain-name "name.local";


  on commit {
      set ClientIP = binary-to-ascii(10, 8, ".", leased-address);
```

```
      set ClientMac = concat (
      suffix (concat ("0", binary-to-ascii (16, 8, "",
substring(hardware,1,1))),2), ":",
      suffix (concat ("0", binary-to-ascii (16, 8, "",
substring(hardware,2,1))),2), ":",
      suffix (concat ("0", binary-to-ascii (16, 8, "",
substring(hardware,3,1))),2), ":",
      suffix (concat ("0", binary-to-ascii (16, 8, "",
substring(hardware,4,1))),2), ":",
      suffix (concat ("0", binary-to-ascii (16, 8, "",
substring(hardware,5,1))),2), ":",
      suffix (concat ("0", binary-to-ascii (16, 8, "",
substring(hardware,6,1))),2));
      log(concat("Request: IP: ", ClientIP, " Mac: ", ClientMac));

 execute("/usr/local/etc/dhcpd/clients_add_drop.sh", "add",
ClientIP, ClientMac);}
  on release {
      set ClientIP = binary-to-ascii(10, 8, ".", leased-address);
      set ClientMac = concat (
      suffix (concat ("0", binary-to-ascii (16, 8, "",
substring(hardware,1,1))),2), ":",
      suffix (concat ("0", binary-to-ascii (16, 8, "",
substring(hardware,2,1))),2), ":",
      suffix (concat ("0", binary-to-ascii (16, 8, "",
substring(hardware,3,1))),2), ":",
      suffix (concat ("0", binary-to-ascii (16, 8, "",
substring(hardware,4,1))),2), ":",
      suffix (concat ("0", binary-to-ascii (16, 8, "",
substring(hardware,5,1))),2), ":",
      suffix (concat ("0", binary-to-ascii (16, 8, "",
substring(hardware,6,1))),2));


      log(concat("Release: IP: ", ClientIP, " Mac: ", ClientMac));
      execute("/usr/local/etc/dhcpd/clients_add_drop.sh",
"drop_rls", ClientIP, ClientMac);}
  on expiry {
      set ClientIP = binary-to-ascii(10, 8, ".", leased-address);
      log(concat("Timeout: IP: ", ClientIP));
      execute("/usr/local/etc/dhcpd/clients_add_drop.sh",
"drop_exp", ClientIP);}
}
subnet 10.0.0.0 netmask 255.255.255.0 {
}
```

Создаём директории и изменяем её права:

```
mkdir /usr/local/etc/dhcpd/ && chown dhcpd:dhcpd
/usr/local/etc/dhcpd/
```

```
touch /usr/local/etc/dhcpd/clients_add_drop_mysql.sh && touch
/usr/local/etc/dhcpd/clients_add_drop.sh
&& chown dpiacc:dpiacc /usr/local/etc/dhcpd/*

chmod 755 /usr/local/etc/dhcpd/


chmod 755 /usr/local/etc/dhcpd/*
```

Теперь скопируем следующий скрипт в /usr/local/etc/dhcpd/clients_add_drop.sh:

```
#!/usr/bin/expect -f


set METHOD [lindex $argv 0]
set IP_ADDR [lindex $argv 1]
set MAC_ADDR [lindex $argv 2]
set MAC_ADDR [string toupper $MAC_ADDR]
# клиентский интерфейс на микротике:
set INT_ClIENT "vWifi"
set status 0


#Записываем dhcp-lease (start and end) в базе hotspot
spawn /usr/local/etc/dhcpd/./clients_add_drop_mysql.sh "$METHOD"
"$IP_ADDR" "$MAC_ADDR"


expect "end_mysql";


#Подключаемся к роутеру
spawn ssh -i /usr/local/etc/dhcpd/.ssh/id_rsa admin+t@100.64.0.1 -
oStrictHostKeyChecking=no -oUserKnownHostsFile=/dev/null
expect {
    "password:" {send "\n";}
    "timeout" {set status 1;}
    ">" {}
}
if { $METHOD == "add" && $status == 0} {
send "ip arp add address=$IP_ADDR mac-address=$MAC_ADDR
interface=$INT_ClIENT\r";
expect ">";


send "ip firewall address-list remove \[find address=$IP_ADDR
list=DROP_CLIENTS\]\r";
expect ">";
send "log info \"ADD: $IP_ADDR -- $MAC_ADDR\"\r";
expect ">"
```

```
send "quit\r";
expect eof
} elseif { $METHOD == "drop_rls" && $status == 0} {
send "ip arp remove \[find mac-address=$MAC_ADDR\]\r";
expect ">";
send "ip firewall address-list add address=$IP_ADDR
list=DROP_CLIENTS\r";
expect ">";
send "log info \"DROP_RLS: $IP_ADDR -- $MAC_ADDR\"\r";
expect ">"
send "quit\r";
expect eof
} elseif { $METHOD == "drop_exp" && $status == 0} {
send "ip arp remove \[find address=$IP_ADDR\]\r";
expect ">";
send "ip firewall address-list add address=$IP_ADDR
list=DROP_CLIENTS\r";
expect ">";
send "log info \"DROP_EXP: $IP_ADDR\"\r";
expect ">"
send "quit\r";
expect eof
} elseif {$status == 0} {
send "quit\r";


expect eof
exit 1;
}


set status 0


#Подключаемся к скату и прописываем статическую запись абона.
spawn ssh -i /usr/local/etc/dhcpd/.ssh/id_rsa dpisu@10.0.0.6 -
oStrictHostKeyChecking=no -oUserKnownHostsFile=/dev/null


expect {
    "password" {send "\r"}
    "timeout" {set status 1; exit 4}
    "\$" {}
}
if {$status == 0} {
send "/var/dpiui2/add_captive_portal_auth_ivstar.sh $IP_ADDR\r"
expect "\$"
send "exit\r";
expect eof
}<code>И скопируем в /usr/local/etc/dhcpd/clients_add_drop_mysql.sh
скрипт для добавления в базу hotspot данных о dhcp-lease:<code
```

```
bash>#!/bin/bash
METHOD=$1
IP_ADDR=$2
MAC_ADDR=$3


MYSQL_CONNECT_LEASEDB="mysql -u root -pvasexperts -Dwifi_hotspot -
h 127.0.0.1"


if [ "$METHOD" = "add" ]; then
    echo "insert into hotspot_aaa(TYPE,MAC,IP)
values("1",\""$MAC_ADDR"\",\""$IP_ADDR"\");" |
$MYSQL_CONNECT_LEASEDB
elif
   [ "$METHOD" = "drop_rls" ]; then
    echo "insert into hotspot_aaa(TYPE,MAC,IP)
values("2",\""$MAC_ADDR"\",\""$IP_ADDR"\");" |
$MYSQL_CONNECT_LEASEDB


elif
   [ "$METHOD" = "drop_exp" ]; then
    echo "insert into hotspot_aaa(TYPE,MAC,IP)
values("2",\"""\",\""$IP_ADDR"\");" | $MYSQL_CONNECT_LEASEDB
fi

echo "end mysql"
```

Включаем сервер dhcpd и добавим правило в firewall:

```
systemctl enable dhcpd
systemctl start dhcpd
firewall-cmd --permanent –add-service=dhcp
firewall-cmd --reload
```

3. Создадим скрипт для переноса файла сессий на ftp:

```
mkdir  /srv/aaa/
mkdir  /srv/aaa/processed/
mkdir  /srv/aaa/script/
touch /srv/aaa/script/script.sh
```

Скопируем содержимое в /srv/aaa/script/script.sh:

```
#!/bin/bash


FTP_ADDR="<ip ftp>"
FTP_USER="<user ftp>"
FTP_PASS="<password ftp>"
```

```
#директория с aaa hotspot
DIR="/var/www/html/wifi_hotspot/backend/storage/aaa_events"


ls $DIR | while read f; do
    curl --user $FTP_USER:$FTP_PASS --upload-file $DIR/$f
ftp://$FTP_ADDR/ISP/aaa/  > /dev/null 2>&1
    mv $DIR/$f /srv/aaa/processed
```

и добавим на выполнение в cron:

```
crontab -e
*/5 * * * * /srv/aaa/script/script.sh
```

4. Создадим открытый и закрытый ключ:

```
mkdir usr/local/etc/dhcpd/.ssh && cd usr/local/etc/dhcpd/.ssh
ssh-keygen -t rsa
```

Секретную фразу оставляем пустой
Внимание! Переносим id.pub на скат (10.0.0.6) и микротик (100.64.0.1)!

- скат (10.0.0.6): перенести файл по ssh на скат и добавить в authorized_keys

```
cat id.pub >>  ~/.ssh/authorized_keys
```

- микротик (100.64.0.1): перенести файл по ssh или через web интерфейс и сделать import

```
user ssh-keys import public-key-file=id.pub user=admin
```

Настройка СКАТ

1. Настроим на скате db для юзеров:

```
nano /etc/dpi/fastdpi.conf
udr=1
```

2. Настроим фильтрацию по федеральному списку:

```
black_list_sm=0
federal_black_list=1
#редирект на страничку
black_list_redirect=http://block.lan/
```

3. Сделаем класс по умолчанию: class_order=0
4. Включим выгрузку ipfix:
    - Настроить интерфейс eth1: nano /etc/sysconfig/network-scripts/ifcfg-eth1

```
BOOTPROTO=none
ONBOOT=yes
IPADDR=<ip address>
PREFIX=24
```

```
netflow=8
netflow_dev=eth1
netflow_timeout=20
netflow_full_collector_type=2
netflow_full_collector=127.0.0.1:1500
netflow_passive_timeout=10
netflow_active_timeout=20
netflow_rate_limit=30
ipfix_dev=eth1

ipfix_tcp_collectors=<ip:port ipfix collectors>
ipfix_meta_tcp_collectors=<ip:port ipfix collectors>
ipfix_observation=127
ipfix_dns_tcp_collectors=<ip:port ipfix collectors>
ipfix_nat_udp_collectors=<ip:port ipfix collectors>
```

5. Сделаем трафик в class 7 минимальным:

```
tbf_class7=rate 1kbit
tbf_inbound_class7=rate 1kbit
```

6. Включим редирект на captive portal: cp_server=10.0.0.4 (ip cp)
7. Выключим nat для приватных адресов: nat_exclude_private=1
8. Остальные настройки СКАТ:

```
ctrl_port=29000
ctrl_dev=lo
scale_factor=1
num_threads=2
class_order=0
mem_tracking_flow=1500000
mem_tracking_ip=3000000
http_parse_reply=1
rlimit_fsize=32000000000
```

9. Заменить содержимое скрипта /var/dpiui2/add_captive_portal_auth_ivstar.sh на следующие:

```
#!/bin/sh
fdpi_ctrl load --service 5 --profile.name='hotspot_white_list_profile'
--ip $1
fdpi_ctrl load --service 11 --profile.name='NAT_PUBLIC_WIFI' --ip $1
fdpi_ctrl load --policing --profile.name='wifi_hotspot_auth_policing' -
-ip $1
```

10. Добавить открытый ключ для доступа с hotspot на скат в файл /home/dpisu/.ssh/authorized_keys:

```
#!/bin/sh
fdpi_ctrl load --service 5 --profile.name='hotspot_white_list_profile'
--ip $1
fdpi_ctrl load --service 11 --profile.name='NAT_PUBLIC_WIFI' --ip $1
fdpi_ctrl load --policing --profile.name='wifi_hotspot_auth_policing' -
-ip $1
```

Сохраняем все изменения в файле /etc/dpi/fastdpi.conf и делаем reboot.

11. Настроим интерфейс eth0 для доступа к hotspot и dpiui

```
nano /etc/sysconfig/network-scripts/ifcfg-eth0
```

```
BOOTPROTO=none
ONBOOT=yes
IPADDR=10.0.0.6
PREFIX=24
DNS1=10.0.0.2
```

Настройка DPI и Hotspot через DPIUI

Настройка приоритизации по протоколам

1. Переходим во вкладку Управление DPI → ПРИОРИТИЗАЦИЯ ПО ПРОТОКОЛАМ (DSCP) → Редактор

- cs0 – что пропускаем
- cs1 – что зажимаем тарифом
- cs7 – что зажимаем глобально

```
Bittorrent cs7
default cs1
dns cs0
http cs0
https cs0
```

2. CG-NAT в СКАТе:
Переходим во вкладку Управление услугами → Услуги → CGNAT
Создаем профиль:
Описание: NAT_WIFI\\Тип: CGNAT
Nat IP пул: <public ip>
Число tcp сессий: 1000 (на абонента)
Число udp сессий: 1000 (на абонента)

Настройка Hotspot:

1. Переходим во вкладку Управление услугами → Hotspot
   Web сервер: WiFi-Hotspot (BM (cp_wifi_vm) заведенная ранее в dpiui)

Captive portal URL: https://10.0.0.4 (url cp)
Время жизни сессии: 36000
URL для редиректа: https://google.ru (страничка редиректа после успешной авторизации)

2. Включаем WiFi и SMS авторизацию
   SMS авторизацию через сервис sms.ru:
   Метод: Post
   Url: https://sms.ru/sms/send
3. Тело (From):

```
api_id  =  <id из личного кабинета sms.ru>
to  =  [PHONE]
msg  =  Ваш код для WIFI: [CODE]
```

Настройка тарифов Hotspot (в редакторе):

1. Тариф для авторизации:

```
htb_inbound_root=rate 5mbit ceil 5mbit burst 2500kbit cburst 2500kbit
htb_inbound_class0=rate 8bit ceil  5mbit burst 8bit cburst 2500kbit
htb_inbound_class1=rate 8bit ceil  8bit burst 8bit cburst 8bit
htb_inbound_class2=rate 8bit ceil  8bit burst 8bit cburst 8bit
htb_inbound_class3=rate 8bit ceil  8bit burst 8bit cburst 8bit
htb_inbound_class4=rate 8bit ceil  8bit burst 8bit cburst 8bit
htb_inbound_class5=rate 8bit ceil  8bit burst 8bit cburst 8bit
htb_inbound_class6=rate 8bit ceil  8bit burst 8bit cburst 8bit
htb_inbound_class7=rate 8bit ceil  8bit burst 8bit cburst 8bit
htb_root=rate 100kbit ceil 100kbit burst 50kbit cburst 50kbit
htb_class0=rate 8bit ceil  100kbit burst 8bit cburst 50kbit
htb_class1=rate 8bit ceil  8bit burst 8bit cburst 8bit
htb_class2=rate 8bit ceil  8bit burst 8bit cburst 8bit
htb_class3=rate 8bit ceil  8bit burst 8bit cburst 8bit
htb_class4=rate 8bit ceil  8bit burst 8bit cburst 8bit
htb_class5=rate 8bit ceil  8bit burst 8bit cburst 8bit
htb_class6=rate 8bit ceil  8bit burst 8bit cburst 8bit
htb_class7=rate 8bit ceil  8bit burst 8bit cburst 8bit
```

2. Тариф для бесплатного WiFi:

```
htb_inbound_root=rate 10mbit ceil 10mbit burst 5mbit cburst 5mbit
htb_inbound_class0=rate 8bit ceil  10mbit burst 8bit cburst 5mbit
htb_inbound_class1=rate 8bit ceil  10mbit burst 8bit cburst 5mbit
htb_inbound_class2=rate 8bit ceil  10mbit burst 8bit cburst 5mbit
htb_inbound_class3=rate 8bit ceil  10mbit burst 8bit cburst 5mbit
htb_inbound_class4=rate 8bit ceil  10mbit burst 8bit cburst 5mbit
htb_inbound_class5=rate 8bit ceil  10mbit burst 8bit cburst 5mbit
htb_inbound_class6=rate 8bit ceil  10mbit burst 8bit cburst 5mbit
htb_inbound_class7=rate 8bit ceil  8bit burst 8bit cburst 8bit
htb_root=rate 10mbit ceil 10mbit burst 5mbit cburst 5mbit
htb_class0=rate 8bit ceil  10mbit burst 8bit cburst 5mbit
htb_class1=rate 8bit ceil  10mbit burst 8bit cburst 5mbit
htb_class2=rate 8bit ceil  10mbit burst 8bit cburst 5mbit
```

```
htb_class3=rate 8bit ceil  10mbit burst 8bit cburst 5mbit
htb_class4=rate 8bit ceil  10mbit burst 8bit cburst 5mbit
htb_class5=rate 8bit ceil  10mbit burst 8bit cburst 5mbit
htb_class6=rate 8bit ceil  10mbit burst 8bit cburst 5mbit
htb_class7=rate 8bit ceil  8bit burst 8bit cburst 8bit
```

3. Услуги:
Переходим к управлению услугами и включаем CGNAT и выбираем профиль NAT_WIFI

4. Белый список:
Переходим во вкладку Управление услугами → услуги → черные и белые списки.
Выбираем нужный профиль и создаем список: `ip 10.0.0.4 (ip cp)`
Если для cp есть запись в dns, то добавляем так: `cn example.com`
Сохраняем настройки через интерфейс.

Настройка Микротика 100.64.0.1

1. Настройка клиентского интерфейса микротика
   Обновить до Router OS 6.48.x

```
/interface vlan
add arp=reply-only arp-timeout=10m interface=sfp1 name=vWifi vlan-id=40


/ip settings
set icmp-rate-limit=5 rp-filter=strict


/ip address
add address=100.64.0.1/22 interface=vWifi network=100.64.0.0


/ip dhcp-relay
add dhcp-server=10.0.0.4 disabled=no interface=vWifi local-
address=100.64.0.1 name=relay1


/ip dns
set servers=10.0.0.2


/ip route
add distance=1 dst-address=10.0.0.4/32 gateway=<указать шлюз> pref-
src=100.64.0.1


/system clock
set time-zone-name=Europe/Moscow
```

```
/system ntp client
set enabled=yes primary-ntp=<указать ntp сервер>


/tool bandwidth-server
set authenticate=no enabled=no
```

2. Настроить ip связь между dhcp/hotspot и микротиком

Настройка unifi network

1. Настроить точки ubiquiti:
    - Установить unifi network на сервер
    - Настроить dhcp для выдачи настроек точкам
    - Если точки и контроллер в разных подсетях, то в dhcp указываем option 43 и присваиваем ей значение ip контроллера (в формате hex). Используя инструкцию: https://help.ui.com/hc/en-us/articles/204909754-UniFi-Device-Adoption-Methods-for-Remote-UniFi-Controllers
      Внимание! Нужно переключиться на старый интерфейс, для этого надо отжать рычажок в System Settings → New USER Interface
2. Настроить Сеть и прочее:
    - Перейти в настройки и далее в Network
      Создать новою сеть и указать vlan 40 и название WiFi-Client, шлюз указать как 100.64.0.1/22, остальное не имеет значение
    - Перейти в настройки и далее в Guest Control
      В Pre-Authorization Access указать ip hotspot (10.0.0.4)
    - Перейти в настройки и далее в Wireless Networks
        - Создаём wifi сеть
        - Сразу открываем ADVANCED OPTIONS
        - Вписываем любое имя/SSID
        - Ставим галочку напротив Enabled
        - Ставим галочку напротив Open
        - Ставим галочку напротив Guest Policy
        - В Network выбираем WiFi-Client
        - Ставим галочку напротив Block LAN to WLAN Multicast and Broadcast Data
        - Ставим галочку напротив Allow BSS Transition with WNM
        - Ставим галочку напротив Block Tunneled Link Direct Setup (TDLS) connections
        - Ставим галочку напротив Isolates stations on layer 2 (ethernet) level
    - Нажимаем Save