

Содержание

Короткие команды alias	3
findip <IP> — поиск по IP-адресу	3
findmac <MAC> — поиск по MAC-адресу	3
findall — вывод всех абонентов	4
findsub <IP> — просмотр правил ограничения скорости	4
findfilter <IP> — просмотр правил услуги 16	6
dropip <IP> — удаление статуса авторизации, L2 свойств; сброс авторизации для DHCP сессии	6
leasesdhcpstatus — просмотр статистики утилизации пулов в KEA и СКАТ (в СКАТ с разделением по VRF)	6
subping <IPinet> <IPsubs> — Ping Internet зоны от “имени” абонента	8
findcgnatstatus <IP> — просмотр статуса CGNAT	8
healthcheck <IP> — просмотр статуса системы	9
bngrestart — перезапуск fastDPI, fastPCRF для сброса всех сессий и применения холодных параметров	9
bngstop — остановка fastDPI, fastPCRF	9
bngstart — запуск fastDPI, fastPCRF	9
bngstatus — статус fastDPI, fastPCRF	9
bngfullrestart — перезапуск fastDPI, fastPCRF для сброса всех сессий и применения холодных параметров с очисткой UDR	10

Короткие команды alias

Команды alias позволяют пользователю с помощью одного слова или даже символа запускать любую команду или группу команд, в том числе с опциями, параметрами и файлами.

alias — просмотр доступных коротких команд.

findip <IP> — поиск по IP-адресу

Соответствует командам:

```
if [[ $# -ne 1 ]]; then
    echo "Use: findip 'IP_ADDRESS'"
else
    PrintDate
    echo "Authorizaion status is: "
    fdpi_cli subs auth show ${1}
    echo "L2 properties are: "
    fdpi_cli subs prop show ${1}
    echo "DHCP session info: "
    fdpi_cli dhcp show ip=${1}
    echo "Accounting session is: "
    fdpi_cli pcrf acct show ip=${1}
fi
```

findmac <MAC> — поиск по MAC-адресу

Сначала находится активный (non expired) IP-адрес для этого MAC адреса, затем аналогичные команды для findip.

Соответствует командам:

```
if [[ $# -ne 1 ]]; then
    echo "Use: findmac 'mac address'"
else
    PrintDate
    declare MAC="${1}"
    declare ROWS=$(fdpi_cli subs prop show mac="${MAC}" --json --strict)
    if [[ $(echo "${ROWS}" | jq -r '[]|length') != 0 ]]; then
        declare SUB_IP=$(echo "${ROWS}" | jq -r
'.[]|records[].address.n_ip')
        echo "${SUB_IP}"
        echo "Autharizaion status is: "
        fdpi_cli subs auth show "${SUB_IP}"
        echo "L2 properties are: "
        fdpi_cli subs prop show "${SUB_IP}"
    fi
fi
```

```

    echo "DHCP session info: "
    fdpi_cli dhcp show ip="${SUB_IP}"
    echo "Accounting session is: "
    fdpi_cli pcrf acct show ip="${SUB_IP}"
    return 0
else
    echo "${MAC} is not exist"
fi
unset SUB_IP
unset ROWS
unset MAC
fi

```

findall — вывод всех абонентов

Соответствует командам:

```

PrintDate
declare SUBS_ACTIVE=$(fdpi_cli subs prop show active --json --strict)
if [[ $( echo "${SUBS_ACTIVE}" | jq -ajr [].records) != 'null' ]]; then
    echo -e "$(echo "${SUBS_ACTIVE}" | jq -ajr
'.[].records[]|\n\(.address.n_ip)|\(.props.vrf)|\(.props.mac)|\(.props.l2su
bs_id)|\(.props.auth_status.ts_timestamp
.absolute_time)\n")" | grep -v -e '^"' |column -t -N
IP,VRF,MAC,L2_SUBS_ID,SESSION_STARTED -s '|'
    else
        echo "No one connected subscriber."
    fi

```

findsub <IP> — просмотр правил ограничения скорости

Соответствует командам:

```

PrintDate
echo "IP Address: ${1}"
echo "Subscriber's policing rules: "
echo -e "$(fdpi_ctrl list --policing --ip "${1}" --outformat=json2 | jq
-ajr
'.lpolicings[].policing|\n'outbound'|\(.outbound.classes[0].class)|\(.outbo
und.classes[0].ceil_value) \(.ou
tbound.classes[0].ceil_unit)|\n'outbound'|\(.outbound.classes[1].class)|\(.o
utbound.classes[1].ceil_value)
\(.outbound.classes[1].ceil_unit)|\n'outbound'|\(.outbound.classes[2].class)
|\(.ou
tbound.classes[2].ceil_value)
\(.outbound.classes[2].ceil_unit)|\n'outbound'|\(.outbound.classes[3].class)
|\(.outbound.classes[3].ceil_value)

```

```

\(.outbound.classes[3].ceil_unit)|\n'outbound' |
\(.outbound.classes[4].class)|\(.outbound.classes[4].ceil_value)
\(.outbound.classes[4].ceil_unit)|\n'outbound'|\(.outbound.classes[5].class)
|\(.outbound.classes[5].ceil_value) \(.outbound.
classes[5].ceil_unit)|\n'outbound'|\(.outbound.classes[6].class)|\(.outbound
.classes[6].ceil_value)
\(.outbound.classes[6].ceil_unit)|\n'outbound'|\(.outbound.classes[7].class)
|\(.outbound.
classes[7].ceil_value)
\(.outbound.classes[7].ceil_unit)|\n'inbound'|\(.inbound.classes[0].class)|\
(.inbound.classes[0].ceil_value)
\(.inbound.classes[0].ceil_unit)|\n'inbound'|\(.inbound.c
lasses[1].class)|\(.inbound.classes[1].ceil_value)
\(.inbound.classes[1].ceil_unit)|\n'inbound'|\(.inbound.classes[2].class)|\
(.inbound.classes[2].ceil_value) \(.inbound.classes[2].ceil_uni
t)|\n'inbound'|\(.inbound.classes[3].class)|\(.inbound.classes[3].ceil_value
)
\(.inbound.classes[3].ceil_unit)|\n'inbound'|\(.inbound.classes[4].class)|\
(.inbound.classes[4].ceil_value) \(.
inbound.classes[4].ceil_unit)|\n'inbound'|\(.inbound.classes[5].class)|\(.in
bound.classes[5].ceil_value)
\(.inbound.classes[5].ceil_unit)|\n'inbound'|\(.inbound.classes[6].class)|\
.inbound
.classes[6].ceil_value)
\(.inbound.classes[6].ceil_unit)|\n'inbound'|\(.inbound.classes[7].class)|\
(.inbound.classes[7].ceil_value) \(.inbound.classes[7].ceil_unit)\n''")|grep
-v -e '^"'|co
lumn -t -N DIRECTION,CLASS,RATE_LIMIT -s '|'

echo 'IMPORTANT! "8 bps" is equivalent to "0 bps"'

```

Пример Policing для конкретного абонента:

```

Current time: 2024-08-21T13:54:15.917Z
IP Address: 89.179.251.130
Subscriber's policing rules:
DIRECTION CLASS RATE_LIMIT
outbound 0 110 mbps
outbound 1 110 mbps
outbound 2 8 bps
outbound 3 8 bps
outbound 4 110 mbps
outbound 5 110 mbps
outbound 6 110 mbps
outbound 7 110 mbps
inbound 0 110 mbps
inbound 1 110 mbps
inbound 2 8 bps
inbound 3 8 bps
inbound 4 110 mbps
inbound 5 110 mbps

```

```
inbound 6 110 mbps
inbound 7 110 mbps
IMPORTANT! "8 bps" is equivalent to "0 bps"
```

На SKAT BRAS от VAS Experts запрет на тот или иной абонентский трафик достигается отнесением его к определенному сервису (для IP, TCP, UDP используются разные), маппинга этого сервиса в определенный QoS Class, и выделением этому классу 8bit полосы пропускания. Подробнее о реализованных таким образом абонентских фильтрах — в HLD.

findfilter <IP> — просмотр правил услуги 16

Разрешенные TCP ресурсы и адрес для редиректа.

Соответствует командам:

```
PrintDate
fdpi_ctrl list --service 16 --ip ${1}
```

dropip <IP> — удаление статуса авторизации, L2 свойств; сброс авторизации для DHCP сессии

Следующий DHCP запрос от абонента отправится на авторизацию); остановка Accounting сессии.

Соответствует набору команд:

```
PrintDate
echo "DHCP session info: "
fdpi_cli dhcp reauth ip=${1}
echo "Dropping autharizaion status..."
fdpi_cli subs auth clear ${1}
echo "Dropping L2 properties..."
fdpi_cli subs prop del ${1}
echo "Accounting session is: "
fdpi_cli pcrf acct stop ip=${1}
```

leasesdhcpstatus — просмотр статистики утилизации пулов в KEA и SKAT (в SKAT с разделением по VRF)

Соответствует командам:

```
KEA_CTRL_SOCKET='/var/kea4-ctrl/kea4-ctrl-socket'
CMD_STATUS='{"command":"stat-lease4-get"}'
CMD_CONFIG='{"command":"config-get"}'
```

```

sendtodhcp(){
    echo ${1} | socat - UNIX-CONNECT:${KEA_CTRL_SOCKET}
}

jsongetvalue(){
    echo ${1} | jq -r .${2}
}

jsongetcount(){
    echo ${1} | jq -r .${2} | jq 'length'
}

getstat(){
    echo $(sendtodhcp ${CMD_STATUS})
}

POOL_STATUS=$(jsongetvalue "$(sendtodhcp ${CMD_STATUS})"
arguments.'''result-set''')
KEA_CONFIG=$(jsongetvalue "$(sendtodhcp ${CMD_CONFIG})" arguments.Dhcp4)
SUBS_NUMBER=$(jsongetvalue "${POOL_STATUS}" rows | jq 'length')
for (( SUBS_PROP_INDEX=0; SUBS_PROP_INDEX < ${SUBS_NUMBER};
SUBS_PROP_INDEX++ )); do
    for (( j=0; j <= ${SUBS_NUMBER}; j++ )); do
        SUBNETS_NUMBER=$(jsongetcount "${KEA_CONFIG}" '''shared-
networks'''[${SUBS_PROP_INDEX}].subnet4)
        if [[ ${SUBNETS_NUMBER} > 1 ]]; then
            for (( SUBNET_INDEX=0; SUBNET_INDEX < SUBS_NUMBER;
SUBNET_INDEX++ )); do
                if [[ $(jsongetvalue "${KEA_CONFIG}" '''shared-
networks'''[${SUBS_PROP_INDEX}].subnet4[${SUBNET_INDEX}].id) != 'null' ]];
then
                    if [[ $(jsongetvalue "${KEA_CONFIG}" '''shared-
networks'''[${SUBS_PROP_INDEX}].subnet4[${SUBNET_INDEX}].id) ==
$(jsongetvalue "${POOL_STATUS}" rows[${j}][0]) ]]; th
en
                        echo 'Pool name: '$(jsongetvalue
"${KEA_CONFIG}" '''shared-networks'''[${SUBS_PROP_INDEX}].name)' Total
addresses: '$(jsongetvalue "${POOL_STATUS}" rows[${j}]
[1]) 'Percent usage:'$(echo "($(jsongetvalue "${POOL_STATUS}"
rows[${j}][3])/$(jsongetvalue "${POOL_STATUS}" rows[${j}][1))*100" | bc -l)
                    fi
                fi
            done
        else
            if [[ $(jsongetvalue "${KEA_CONFIG}" '''shared-
networks'''[${SUBS_PROP_INDEX}].subnet4[0].id) != 'null' ]]; then
                if [[ $(jsongetvalue "${KEA_CONFIG}" '''shared-
networks'''[${SUBS_PROP_INDEX}].subnet4[0].id) == $(jsongetvalue
"${POOL_STATUS}" rows[${j}][0]) ]]; then
                    echo 'Pool name: '$(jsongetvalue "${KEA_CONFIG}"

```

```

'''shared-networks'''[${SUBS_PROP_INDEX}].name)' Total addresses: '
$(jsongetvalue "${POOL_STATUS}" rows[${j}][1]
) 'Percent usage:'$(echo "$(jsongetvalue "${POOL_STATUS}"
rows[${j}][3])/$(jsongetvalue "${POOL_STATUS}" rows[${j}][1))*100" | bc -
l)
fi
fi
done
done

fdpi_cli dhcp show stat vrf

```

subping <IPinet> <IPsubs> – Ping Internet зоны от “имени” абонента

Соответствует командам:

```

if [[ "$1" == "--help" ]] || [[ "$1" == "-h" ]] || [[ $# -eq 0 ]] || [[
$# -gt 2 ]] ; then
    echo "Usage:"
    echo "subping <IP_address_in_the_internet_side>
<IP_address_of_the_subscriber>"
else
    fdpi_cli ping inet ${1} from ${2} n=5
fi

```

findcgnatstatus <IP> – просмотр статуса CGNAT

Какой публичный адрес выделен, статистика использования TCP и UDP сессий.

Соответствует команде:

```

sudo fdpi_ctrl list status --service 11 --ip <ip>

```

Проверьте установленный лимит на количество абонентских сессий в созданных NAT профилях:

```

sudo fdpi_ctrl list all profile --service 11

```

Проверьте, не заполнен ли белый IP-адрес, который использует абонент, на 100%

```

sudo fdpi_ctrl list status --service 11 --profile.name CGNAT-INET | grep
puwhb=100.00

```

где CGNAT-INET - имя 11 сервиса NAT

healthcheck <IP> — просмотр статуса системы

Соответствует команде:

```
echo ----PROCESS STATE----&&systemctl status fastdpi&&systemctl status fastpcrf&&echo ----LINK STATE----&&fdpi_cli dev link state show|grep -A 1 Device&& echo ----MGMT ROUTER STATE----&&birdc show status|tail -n +2&& birdc show protocols|tail -n +2&& echo ----MAIN ROUTER STATE----&&birdc show status|tail -n +2&& birdc show protocols|tail -n +2;echo ----AUTHORIZED SUBSCRIBERS----&&fdpi_cli subs auth show|tail -n 6&& echo ----DHCP SUBSCRIBERS----&&fdpi_cli dhcp show all|tail -n 1
```

bngrestart — перезапуск fastDPI, fastPCRF для сброса всех сессий и применения холодных параметров

Соответствует команде:

```
systemctl stop fastdpi && systemctl restart fastpcrf && systemctl start fastdpi
```

bngstop — остановка fastDPI, fastPCRF

Соответствует команде:

```
systemctl stop fastdpi && systemctl stop fastpcrf
```

bngstart — запуск fastDPI, fastPCRF

Соответствует команде:

```
systemctl start fastdpi && systemctl start fastpcrf
```

bngstatus — статус fastDPI, fastPCRF

Соответствует команде:

```
systemctl status fastdpi && systemctl status fastpcrf
```

bngfullrestart — перезапуск fastDPI, fastPCRF для сброса всех сессий и применения холодных параметров с очисткой UDR

```
declare -a TABLES_LIST=("policing" "profile_names" "profiles"
"vchannel_policing" "vchannel_services")
declare -a DB_BAK_DIR="/var/db/dpi/recovered"
declare -a UDR_BAK="/tmp/data.mdb"
declare -a UDR="/var/db/dpi/data.mdb"
if [ ! -d "${DB_BAK_DIR}" ]; then
    mkdir -p "${DB_BAK_DIR}"
else
    for TABLE in "${TABLES_LIST[@]"; do
        if [ -f "${DB_BAK_DIR}/dump.${TABLE}.load ]; then
            rm -f "${DB_BAK_DIR}/dump.${TABLE}.load"
        fi
    done
fi

/bin/cp -f "${UDR}" "${UDR_BAK}"

for TABLE in "${TABLES_LIST[@]"; do
    mdb_dump -f "${DB_BAK_DIR}/dump.${TABLE}.load -s "${TABLE}"
/var/db/dpi
declare -a RETVAL=$?
if [ "${RETVAL}" -ne 0 ]; then
    echo "ERROR: Can not make dump of table ${TBALE} from UDR to
${DB_BAK_DIR}/dump.${TABLE}.load"
    return 1
fi
done

systemctl stop fastdpi
systemctl stop fastpcrf

/bin/rm -f /var/db/dpi/lock.mdb
/bin/rm -f "${UDR}"

for TABLE in "${TABLES_LIST[@]"; do
    mdb_load -f "${DB_BAK_DIR}/dump.${TABLE}.load /var/db/dpi
declare -a RETVAL=$?
if [ "${RETVAL}" -ne 0 ]; then
    echo "ERROR: Can not load table ${DB_BAK_DIR}/dump.${TABLE}.load
to UDR"

    /bin/rm -f /var/db/dpi/lock.mdb
    /bin/cp -f "${UDR_BAK}" "${UDR}"
    systemctl start fastpcrf
    systemctl start fastdpi
    return 1
fi
done
```

```
fi
```

```
done
```

```
systemctl start fastpcrf
```

```
systemctl start fastdpi
```