

Интеграция с Carbon Billing 4

Для версий старше 452_047.
настройки в биллинге

1. Создать нас в биллинге
2. Выбрать тип NAS "Другой"
3. Указать пароль для COA
4. Указать IP адрес XGE
5. Указать "Указать Имя скрипта событий" - xge_router.sh
6. Предварительно записать /var/lib/event/event_inc.sh
7. Скопировать скрипты событий

```
cp -f /var/lib/event/example/xge_router.sh /var/lib/event/xge_router.sh
cp -f /var/lib/event/event_inc.sh.tmpl /var/lib/event/event_inc.sh
```

При отсутствии скриптов в каталоге /var/lib/event/examples, нужно создать новые одноименные файлы и скопировать в них соответствующий текст скрипта, который опубликован в примечании данной статьи.

8. Выполнить в командной строке ASR два запроса

```
nas_id=1
#id
sqlexec "insert into nas_radius_params (NAS_ID,ATTRIBUTE,THEVALUE,OP) values ($nas_id,
'Filter-Id', 'session Framed-IP-Address rate set in \ $rate_in \ $ceil_in out \ $rate_out
\ $ceil_out', '+=')"
sqlexec "insert into nas_radius_params (NAS_ID,ATTRIBUTE,THEVALUE,OP) values ($nas_id,
'Filter-Id', 'session Framed-IP-Address nat \ $nat', '+=')"
```

Проверьте, что у вас в тарифах указаны rate и ceil. Если они не указаны, то ограничение скорости происходить не будет

Настройки в XGE1
через консоль

1. ввести команду menu
 2. для настроек VPN: Carbon XGE Router 5 -> Настройки VPN сервера... включить требуемые типы авторизации и указать dns сервера, которые будут отдаваться абонентам
-
1. Настройки Radius
 2. указать Radius-secret, Пароль COA-клиента, Порт COA-сервера, Общий пароль для IPoE авторизации - те же параметры, как указаны в биллинге.
 3. Сервер авторизации - IP-адрес биллинга с 1812 портом (прим.10.90.245.55:1812)
 4. Сервер аккаунтинга - IP-адрес биллинга с 1813 портом (прим.10.90.245.55:1813)
 5. IP адрес COA-клиента - IP-адрес биллинга
 6. IP адрес этого NAS-сервера - IP-адрес XGE, с которого будет происходить взаимодействие с биллингом (тот, что указан в биллинге, как адрес NASa)
-
1. Настройки сенсора netflow
 2. Сервер сбора статистики - IP-адрес биллинга с 9996 портом (прим.10.90.245.55:9996)

После указанных настроек XGE должен заработать.

Примечание

event_inc.sh

```
LOG_LEVEL=ALL
```

```

SENDER=$1; shift
EVENT=$1; shift
DATA=$@

for VAR in $DATA; do
    [[ "$VAR" = "*" ]] && eval ${VAR%*=}=\'${VAR#*=}\'
done
is_daemon=1

LOG INFO "$SENDER $EVENT $DATA"

if [ "$nas_ip" == "0.0.0.0" ]; then
    exit 0
fi

user_refresh(){
    accept=1
    [ "$logged" = "-1" ] && accept=0
    [ "$enabled" = "0" ] && accept=0
    [ "$deleted" = "1" ] && accept=0
    [ "$own_disabled_end" != -1 ] && accept=0
    redirect=0
    [ "$over_limit" = "1" ] && redirect=1

    [ "$accept" = "1" ] && user_accept
    [ "$redirect" = "0" ] && user_redirect_cancel
    [ "$redirect" = "1" ] && user_redirect
    [ "$accept" = "0" ] && user_drop
}

user_data_changed(){
    if [ -f /var/lib/event/before/$id.before ]; then
        changed=0
        for VAR in $(</var/lib/event/before/$id.before); do
            [[ "$VAR" = "*" ]] && eval ${VAR%*=}=\'${VAR#*=}\'
        done
        for val in ip mac server enabled deleted auth_type nas_ip router_ip router_vlan router_port opt82
        limit_ip acl; do
            old_val="old_$val"
            [ "${!val}" != "${!old_val}" ] && changed=1;
        done
        else
            changed=1
        fi

        if [ "$changed" = "1" ]; then
            (
                DATA_BEFORE=$(</var/lib/event/before/$id.before)
                DATA_BEFORE="${DATA_BEFORE// old_/ }"
                DATA=$DATA_BEFORE
                for VAR in $DATA_BEFORE; do
                    [[ "$VAR" = "*" ]] && eval ${VAR%*=}=\'${VAR#*=}\'
                done
                user_del
            )
            user_del
            user_add
            user_refresh
            user_rate_set
        fi
    }

if [ -f /var/lib/event/$script_name ]; then
    source /var/lib/event/$script_name
    if [ "$EVENT" = "rate_set" ]; then
        user_rate_set
    elif [ "$EVENT" = "user_disconnect" ]; then
        user_disconnect
    elif [ "$EVENT" = "user_add" ]; then
        user_add
    elif [ "$EVENT" = "user_del" ]; then
        user_del
    fi

```

```

        elif [ "$EVENT" = "user_data_changed_before" ]; then
        echo "${DATA// / old_}" >/var/lib/event/before/$id.before
    elif [ "$EVENT" = "user_data_changed" ]; then
        user_data_changed
        elif [ "$EVENT" = "radius_update_err" ]; then
        user_disconnect
        elif [ "$EVENT" = "rad_acc_timeout" ]; then
        user_disconnect
        elif [ "$EVENT" = "try_double_login" ]; then
        user_disconnect
        elif [ "$EVENT" = "user_info" ]; then
        user_info 2>&1 > /tmp/${id}_user_info
        output="$(cat /tmp/${id}_user_info | head -c 8000 | iconv -f koi8-r -t cp1251)"
        rm -f /tmp/${id}_user_info
        {
            . /etc/ics/ics.conf
            echo -e "SET SQL DIALECT 3;\n"
            echo -e "SET NAMES WIN1251;\n"
            echo -e "CONNECT '127.0.0.1:/var/db/ics_main.gdb' USER '$ISC_USER' PASSWORD
'$ISC_PASSWORD';\n"
            echo -e "DELETE FROM users_diagnostic WHERE user_id='$id';\n"
            echo -e "EXECUTE PROCEDURE KNL_USR_DIAGNOSTIC_INS('$id', '$output');\n"
            echo -e "commit;\n"

        } >/tmp/$$tmp
        isql < /tmp/$$tmp
        rm -f /tmp/$$tmp
    else
        user_refresh
        user_rate_set
    fi
else

```

```
    echo "NO SCRIPT FOUND!!!"
fi
```

xge_router.sh

```
#!/bin/bash

#. /usr/local/lib/carbon.shlib

trap __exit EXIT

TMPDIR=/tmp/${0##*/}
mkdir -p $TMPDIR

__exit() {
    local ret=$?
    rm -f $TMPDIR/*.$$
    return $ret
}

# set -eux

burst_in=''
burst_out=''
ceil_in=${ceil_in:-1000}
rate_in=${rate_in:-1000}
ceil_out=${ceil_out:-1000}
rate_out=${rate_out:-1000}

__xge_coa_send() {
    echo "Filter-Id=\"$@" | radclient -x $nas_ip:3799 coa $coa_psw >>$TMPDIR/radclient.$$
    ret=$?
    # exit 254 , todo      busy context
    grep "no response from server" $TMPDIR/radclient.$$ && exit 254
    cat $TMPDIR/radclient.$$
    return $ret
}

user_add() {
    [ "$auth_type" = "1" ] && __xge_coa_send session $ip start IPOE
}

user_del() {
    __xge_coa_send session $ip stop "user_del"
    __xge_coa_send session $ip remove
}

user_accept() {
    [ "$auth_type" = "1" ] && __xge_coa_send session $ip start IPOE
    __xge_coa_send session $ip redirect blocked cancel
    __xge_coa_send session $ip nat $snat_ip
}

user_drop() {
    __xge_coa_send session $ip redirect blocked
}

user_redirect() {
    __xge_coa_send session $ip redirect negbal
}

user_redirect_cancel() {
    __xge_coa_send session $ip redirect negbal cancel
}

user_rate_set() {
    __xge_coa_send session $ip rate set in $rate_in $ceil_in $burst_in out $rate_out $ceil_out $burst_out
}

}
```

```

user_rate_set_cancel() {
    __xge_coa_send session $ip rate remove
}

user_info() {
    __xge_coa_send session $ip info
}

user_test() {
    __xge_coa_send session $ip test
}

user_event_before() {
    :
}

user_event_after() {
    :
}

user_disconnect() {
    __xge_coa_send session $ip disconnect
}

__xge_list_local(){
    ipset -o save -l $4 | grep add | cut -d ' ' -f 3
}

__xge_ssh_send(){
    echo -e "chroot /app/xge $@\nexit\n" | ssh_send --port ${telnet_port:-33} -u
    ${telnet_login:-root} -p ${telnet_password:-servicemode} ${nas_ip:-127.0.0.1}
}

users_from_nas() {
    # coa
    local SYNCNCDIR="/var/lib/event/sync/$nas_ip"
    mkdir -p $SYNCNCDIR
    if [ "$nas_ip" != "169.1.18.12" ]; then
        __xge_ssh_send xgesh show list xge_blocked_list | grep '^^[0-9].*' >
        $SYNCNCDIR/blocked_list.nas
        __xge_ssh_send xgesh show list xge_negbal_list | grep '^^[0-9].*' > $SYNCNCDIR/negbal_list.nas
        __xge_ssh_send xgesh show list xge_auth_list | grep '^^[0-9].*' > $SYNCNCDIR/auth_list.nas
    fi
    #
    if [ "$nas_ip" = "169.1.18.12" ]; then
        __xge_list_local xgesh show list xge_blocked_list | grep '^^[0-9].*' >
        $SYNCNCDIR/blocked_list.nas
        __xge_list_local xgesh show list xge_negbal_list | grep '^^[0-9].*' >
        $SYNCNCDIR/negbal_list.nas
        __xge_list_local xgesh show list xge_auth_list | grep '^^[0-9].*' > $SYNCNCDIR/auth_list.nas
    fi
}

user_info(){
    echo '<pre>' > /tmp/${user_id}_user_info.new
    __xge_coa_send session $ip test human | grep "Reply-Message" | sed -e 's/Reply-Message =
    //g; s/\s\+//g; s/^\s//g; s/"$//g' >> /tmp/${user_id}_user_info.new
    echo '</pre>' >> /tmp/${user_id}_user_info.new

    mv -f /tmp/${user_id}_user_info.new /tmp/${user_id}_user_info
    chown apache:apache /tmp/${user_id}_user_info
}

user_get_mac() {
    local TMPDIR=/tmp/nas_event_daemon/$nas_ip/user_get_mac/
    mkdir -p $TMPDIR/

    __xge_coa_send session $ip get_mac | grep "Reply-Message" | sed -e 's/Reply-Message =//g;
    s/\s\+//g; s/^\s//g; s/"$//g' >> ${TMPDIR}/${user_id}

    chmod 777 -R /tmp/nas_event_daemon/
}

get_mac(){
    res="(echo "Filter-Id=\"session $ip get_mac\" | radclient -x $nas_ip coa $coa_psw | grep
    Reply-Message)"
    res=${res/*= \"/}

```

```
res=${res/\\"/>
```

```
}    sqlexec "SELECT * FROM KNL_USR_SETMAC('$ip', '$res')"
```