

# Початок



## Загальні домовленості конфігурацій

- *внутрішні власні назви*<sup>1)</sup> елементів конфігурації пишемо у верхньому регістрі з нижнім підчеркуванням в якості розділювача ( WHITE\_LIST )
- *зовнішні власні назви*<sup>2)</sup> елементів конфігурації пишемо у верхньому регістрі з нижнім підчеркуванням в якості розділювача ( svc-allow-pppoe )
- Назви динамічних профілів послуг записуємо в форматі **svc-<напрямок>-<тип авторизації>** наприклад:
  - svc-allow-**dhcp**
  - svc-deny-**pppoe**
  - svc-cgn-nat-**qinq**
- Назви динамічних профілів автоконфігурацій записуємо як в офіційній документації + за потреби власний постфікс, наприклад:
  - **Auto-<тип інкапсуляції>-<наявність демультіплектора>-<власний>**
  - Auto-Vlan-Demux
  - Auto-Staked-Vlan-Demux
  - Auto-Staked-Vlan-Demux-Combo
- Завжди коментуйте комміти (commit comment "change ae1 vlans")

Дотримуючись цих простих правил, ви спростите собі і іншим читання і розуміння конфігурації, що прискорить роботу з нею

## Приклади корисних команд

Подивимося поточну конфігурацію:

```
show configuration
```

Приклад скидання налаштувань та повного очищення конфігурації, лог файлів та інше:

```
load factory defaults
request system zeroize
```

Подивимося інформацію про комплектуючі, використання CPU та пам'яті, температуру, версію JunOS тощо.

```
show chassis hardware
show chassis routing-engine
show system memory
show system storage
show chassis environment
show version
show system information
```

Ще кілька прикладів команд для перегляду іншої інформації:

```
show system alarms
show chassis alarms
show system uptime
show system connections
show system errors active
show system users
show system resource-monitor summary
show system processes extensive
show subscribers summary
show subscribers extensive
show bridge mac-table
```

Приклади перегляду логів:

```
show log ?
show log messages | last 30
show log cli-comands | last 30
```

```
show log user admin
show system alarms
show system boot-messages
```

Приклади перезавантаження пристрою:

```
request system reboot
```

Щоб виконати команди CLI, з режиму редагування не повертаючись до CLI, додамо перед командою «run», наприклад:

```
run show configuration
```

## Початкове налаштування Juniper

Приклад базового налаштування Juniper серії MX (JunOS).

Підключимося до пристрою консольним кабелем та запустимо інтерфейс командного рядка (Command Line Interface, CLI):

```
cli
```

За замовчанням логін root без пароля.

Переходимо до конфігурування:

```
configure
```

Вкажемо пароль користувачеві root (після введення команди знадобиться ввести пароль):

```
set system root-authentication plain-text-password
```

Створимо нового користувача admin (наприклад, цим користувачем надалі підключатимемося по SSH):

```
set system login user admin class ?
  operator      permissions [ clear network reset trace view ]
  read-only     permissions [ view ]
  super-user    permissions [ all ]
  unauthorized  permissions [ none ]
set system login user admin class super-user
set system login user admin authentication plain-text-password
```

Вкажемо hostname та домен:

```
set system host-name "mx0"
set system domain-name "ndp.pp.ua"
```

Для пристроїв з оперативною пам'яттю менше 32 ГБ необхідно вказати розмір системної бази даних.

Для пристроїв нижньої лінійки (MX5 - MX104) макимальний розмір БД 100 мб

```
set system configuration-database max-db-size 100M
```

Для пристроїв середньої лінійки (MX240 - MX960) макимальний розмір БД 300 мб

```
set system configuration-database max-db-size 300M
```

Для вступу змін в дію може знадобитись перезавантажити пристрій.

Також добре буде увімкнути версіонування динамічних профілів до їх створення:

```
set system dynamic-profile-options versioning
```

Призначимо IP адресу для Ethernet порту управління (в нього пізніше підключимо кабель), а також вкажемо сервер DNS і маршрут за замовчуванням:

```
set interfaces fxp0 unit 0 family inet address 172.16.0.10/24
```

```
set system name-server 1.1.1.1
```

Щоб застосувати будь-які зміни в конфігурації, необхідно виконати команду «commit», наприклад перевіримо конфігурацію та застосуємо:

```
commit check
commit
commit ?
commit comment "init config"
```

Якщо ви припускаєте, що після підтвердження зникне зв'язок з пристроєм, то можна використовувати команду «commit confirm», яка застосує зміни і якщо не підтвердити другий раз, то зміни будуть скасовані наприклад через 5 хвилин (10 хвилин за замовчуванням якщо не вказувати кількість хвилин):

```
commit confirmed 5
```

Для підтвердження застосування комміту потрібно протягом вказаного часу повторно виконати:

```
commit
```

Якщо цього не зробити, автоматично виконається команда `rollback 1`

Скасувати зміни можна командою `rollback 0`, або можна повернути більш стару конфігурацію:

```
run show system commit
rollback ?
rollback 0
rollback 1
```

Перевіримо зв'язок командою `ping`:

```
run ping 192.168.24.1
```

Активуємо `ssh` або `telnet`:

```
set system services ssh
set system services ssh port 2222
run show system connections extensive | match "22|2222" | match LISTEN
set system services telnet
show system services
```

Подивитися хто ще редагує конфігурацію можна командою:

```
status
```

Налаштуємо час:

```
set system ntp server 192.168.5.5
set system time-zone Europe/Kiev
```

<https://ixnfo.com/en/configuring-juniper-mx-and-abills.html>

1)

внутрішні власні назви — ті які не передаються ззовні.

2)

зовнішні власні назви — ті які передаються наприклад через RADIUS.

From:

<https://ndp.pp.ua/> - my NoDeny Wiki

Permanent link:

[https://ndp.pp.ua/doku.php/juniper/00\\_intro?rev=1687120704](https://ndp.pp.ua/doku.php/juniper/00_intro?rev=1687120704)

Last update: **18/06/2023 20:38**

