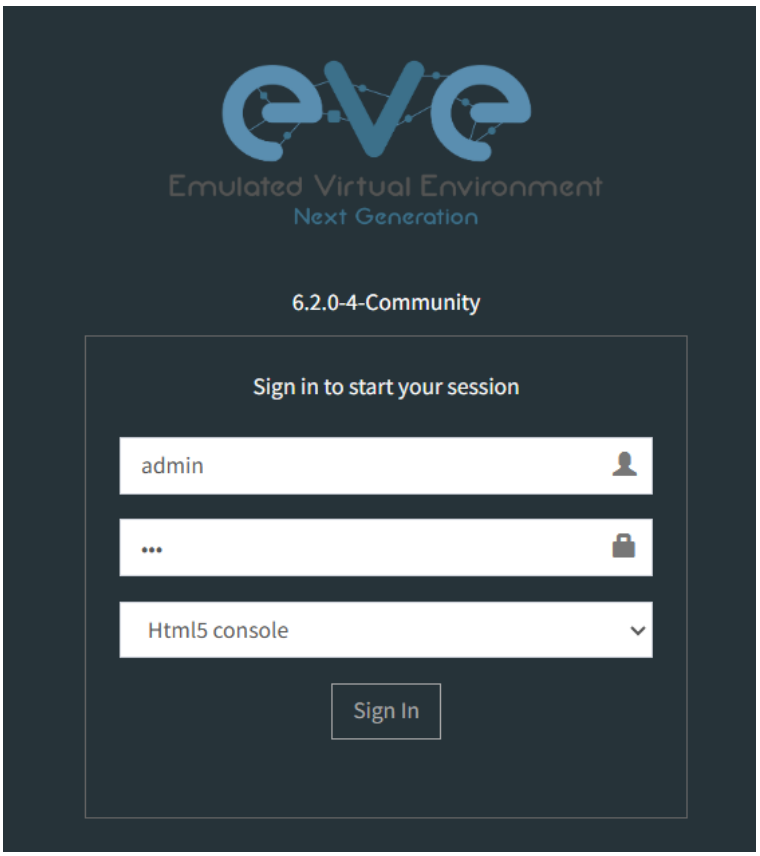
Logujemy się do EVE-NG. W panelu Google Cloud klikamy na nasz adres publiczny. Jeżeli strona się nie otwiera to zmieniamy na http://nasz adres ip

Wewnętrzny adres IP	Zewnętrzny adres IP	Połącz
10.156.0.2 (nic0)	34.40.46.231 (nic0)	SSH

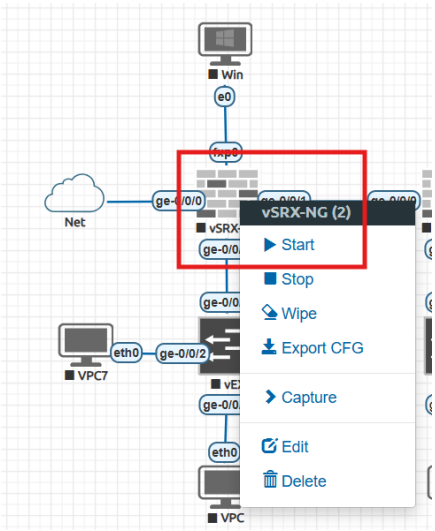
Logujemy się danymi:
Login: admin
Password: eve
Z listy wybieramy Html5 console

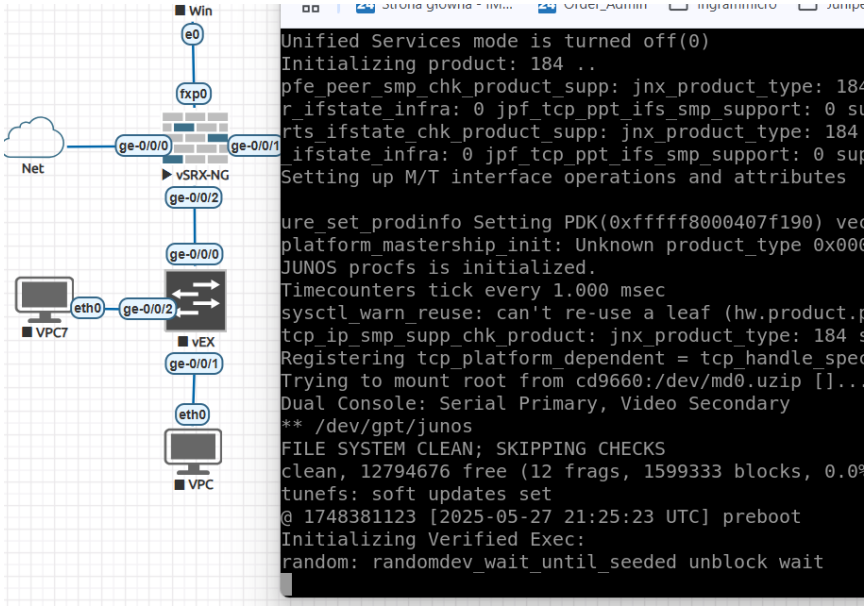


Po zalogowaniu uruchamiany nasz główny router z dostępem do Internetu.
Prawy przycisk myszki na ikonie pokaże okno do zarządzania nodem. Uruchamiamy maszynę przyciskiem Start. Dwuklik na ikonie otworzy nam terminal consoli do routera.

Uwaga 1 - jak nic się nie otwiera to proszę sprawdzić czy przeglądarka nie blokuje wyskakujących okienek.

Uwaga 2 - proszę nie uruchamiać innych nodów! Na tym etapie uruchamiamy tylko jeden router. Szybkość bootowania urządzenia zależy od używanych zasobów.
Jak router będzie gotów widoczne będzie logowanie do shella linuxa.

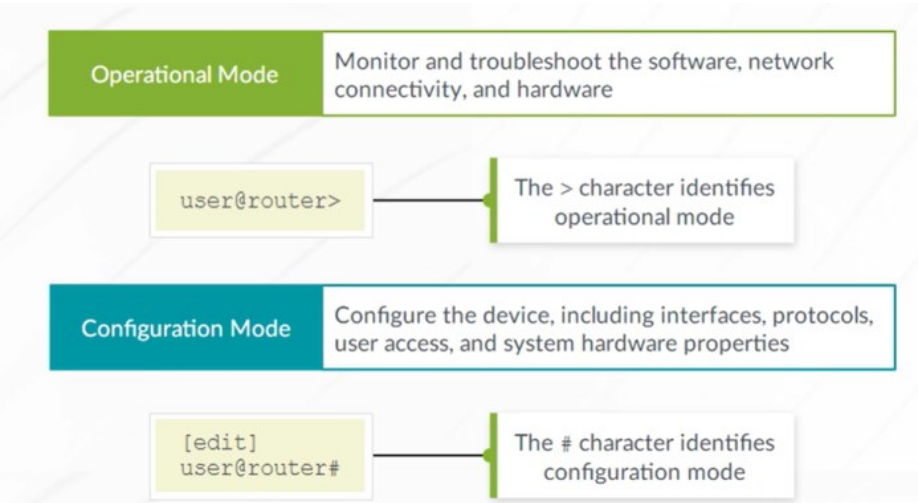




Login root bez hasła.

```
FreeBSD/amd64 (Amnesiac) (ttyu0)

login:

FreeBSD/amd64 (Amnesiac) (ttyu0)

login: root

--- JUNOS 24.4R1.9 Kernel 64-bit XEN JNPR-15.0-20241104.1ed86e6_buil
root@:~ # cli
root> configure
Entering configuration mode

[edit]
root#
```

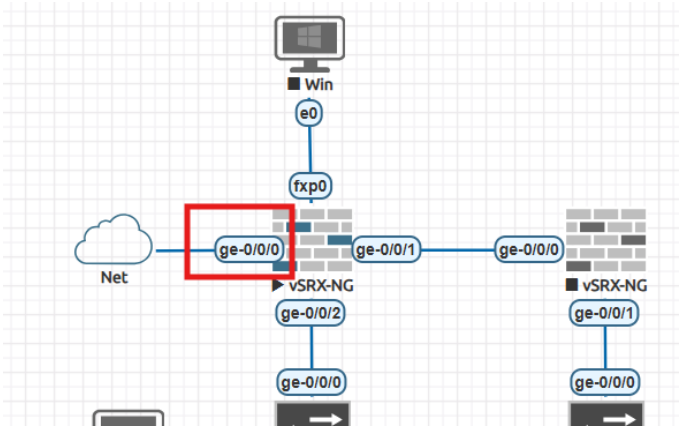
Do Junosa wchodzimy komendą CLI. Do poziomu konfigurowania używamy configure



Krok pierwszy to zawsze ustawienie hasła root:
cli
configure
set system root-authentication plain-text-password
New password:
Retype new password:
commit

Teraz możemy wprowadzać i zatwierdzać (commit) wprowadzane zmiany z configu kandydata, automatycznie do running config (czyli wchodzą od razu w życie).

Do niektórych ustawień routera przyda się dostęp do sieci wiec zaczniemy od ustawienia Interfejsu WAN na ge-0/0/0



Z poziomu trybu konfiguracji czyli znacznika # wykonujemy następujące komendy:

set interfaces ge-0/0/0 unit 0 family inet address 172.16.100.101/24

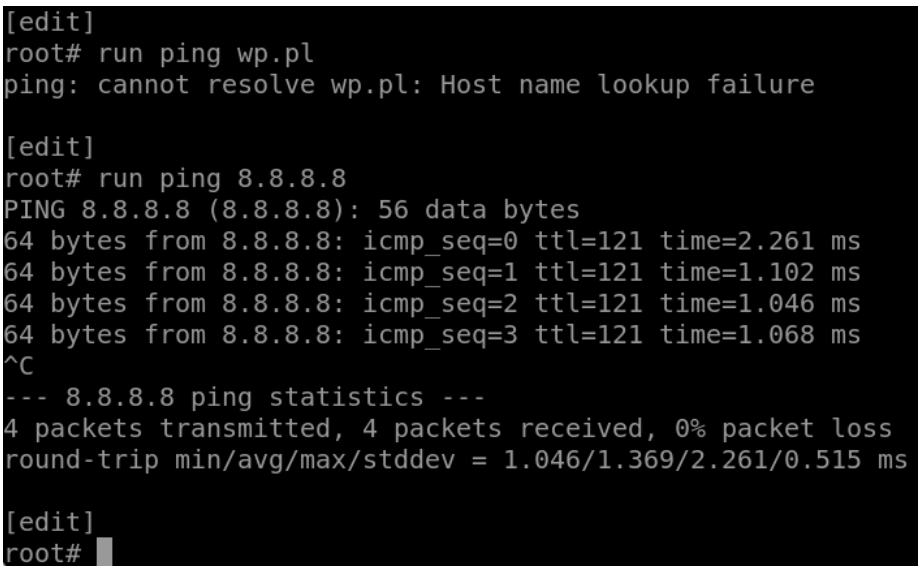
set routing-options static route 0.0.0.0/0 next-hop 172.16.100.100

set security zones security-zone untrust interfaces ge-0/0/0.0

Możemy teraz zweryfikować czy **brama na pnet2** nam odpowiada na ping.

```
root> configure
Entering configuration mode

[edit]
root# run ping 172.16.100.100
PING 172.16.100.100 (172.16.100.100): 56 data bytes
64 bytes from 172.16.100.100: icmp_seq=0 ttl=64 time=98.261 ms
64 bytes from 172.16.100.100: icmp_seq=1 ttl=64 time=0.776 ms
^C
--- 172.16.100.100 ping statistics ---
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.776/49.518/98.261/48.742 ms

[edit]
root#
```

Możemy również spingować adresy w internecie. Do rozwiązywania nazw url potrzebujemy jeszcze ustawić DNS.



Teraz możemy dokonać niektórych podstawowych rzeczy.
Proszę uważać na znaczniki w jakim trybie wykonujemy komendy.
Zmiany, aby weszły w życie muszą być zapisane przez commit

Ustawienie hostname w Junos
> configure
set system host-name NAZWA_HOSTA

Ustawienie domain-name w Junos:
set system domain-name example.local

Ustawienie DNS w Junos:
set system name-server 8.8.8.8
set system name-server 1.1.1.1

Ręczne ustawienie daty i czasu:
run set date YYYYMMDDHHMM (set date 202505281030)
run show system uptime

Ustawienie automatyczne – przez NTP:
set system ntp server 0.pool.ntp.org
set system ntp server 1.pool.ntp.org

Tworzenie nowego użytkownika:
set system login user admin class super-user
set system login user admin authentication plain-text-password
commit

Możemy sprawdzić ja nasze ustawienia wyglądają w konfigu:

```
root@INGRAM# run show configuration
## Last commit: 2025-05-27 22:18:13 UTC by root
version 24.4R1.9;
system {
    host-name INGRAM;
    root-authentication {
        encrypted-password "$6$btmGH.Fj$geZfLPtK
YNh4P7Gf48."; ## SECRET-DATA
    }
    login {
        user admin {
            uid 2000;
            class super-user;
            authentication {
                encrypted-password "$6$.m9FJoNe$
kZMdgmohu4/38vHZc21"; ## SECRET-DATA
            }
        }
    }
    services {
        ssh;
        web-management {
            http {
                interface fxp0.0;
```

```
[edit]
root@INGRAM# show interfaces
ge-0/0/0 {
    unit 0 {
        family inet {
            address 172.16.100.101/24;
        }
    }
}
fxp0 {
    unit 0;
}

[edit]
root@INGRAM#
```

```
[edit]
root@INGRAM# run show route

inet.0: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

0.0.0.0/0          *[Static/5] 00:25:12
                    >  to 172.16.100.100 via ge-0/0/0.0
172.16.100.0/24    *[Direct/0] 00:25:12
                    >  via ge-0/0/0.0
172.16.100.101/32  *[Local/0] 00:25:12
                       Local via ge-0/0/0.0

inet6.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

ff02::2/128        *[INET6/0] 00:29:26
                       MultiRecv

[edit]
root@INGRAM# run ping wp.pl
PING wp.pl (212.77.98.9): 56 data bytes
64 bytes from 212.77.98.9: icmp_seq=0 ttl=60 time=21.828 ms
64 bytes from 212.77.98.9: icmp_seq=1 ttl=60 time=20.953 ms
 64 bytes from 212.77.98.9: icmp_seq=2 ttl=60 time=20.966 ms
 64 bytes from 212.77.98.9: icmp_seq=3 ttl=60 time=20.996 ms
```

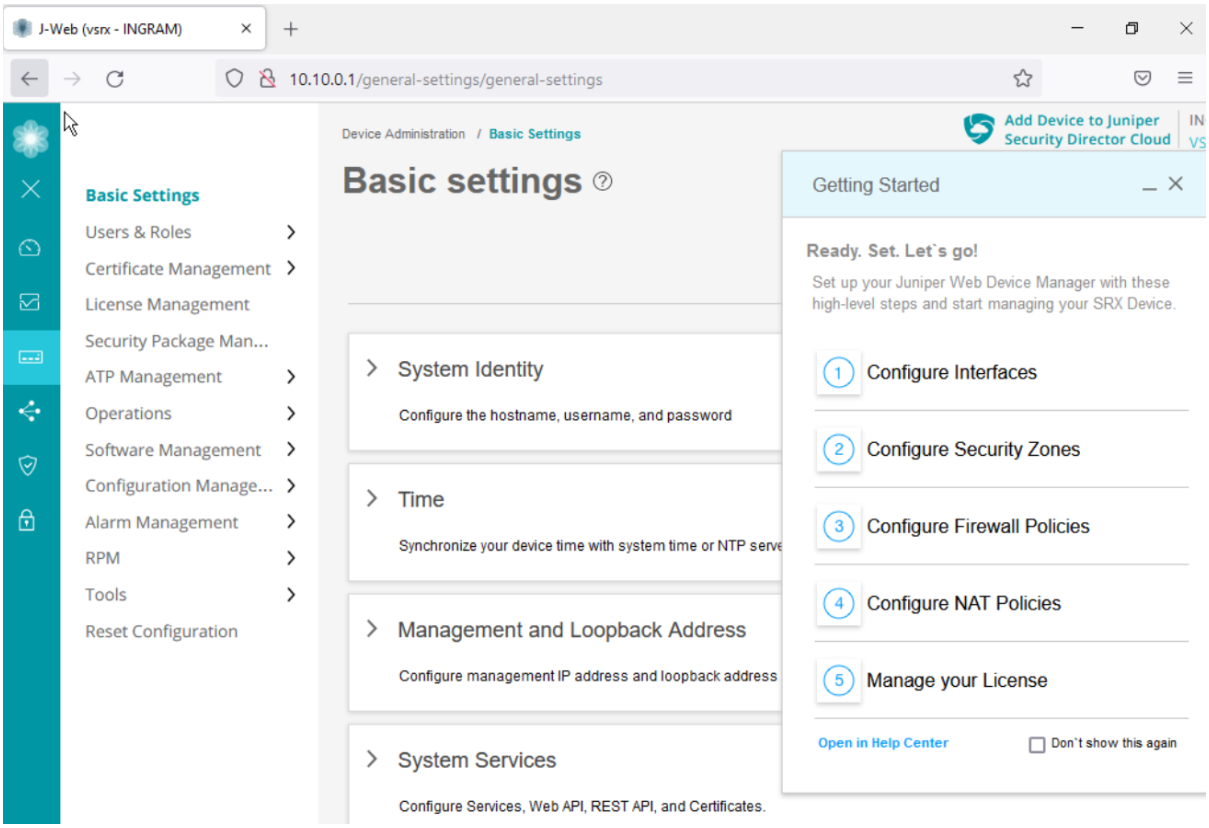
Przygotujmy sobie teraz interfejs zarzadzania Out of band. Użyjemy go do połączenia się do SRX poprzez J-WEB.

```
root@INGRAM# set interfaces fxp0 unit 0 family inet address 10.10.0.1/24

[edit]
root@INGRAM# commit
```

Na karcie sieciowej PC ustawiamy statycznie adresację: 10.10.0.2/24
Z poziomu windowsowego CLI możemy wykonać ping do 10.10.0.1
Ten adres wpisujemy w przeglądarce PC.
Możemy się zalogować na dane utworzonego użytkownika: admin

Po zalogowaniu domyślnie pokaże się wizzard do wstępnej konfiguracji.
Omijamy go aby zapoznać się z każdą sekcja w GUI oraz jakie są tam opcje.

Zobaczymy, że dane które już wprowadziliśmy są w J-web. Cześć dalej konfiguracji możemy wykonać już z tego poziomu lub dalej ćwiczymy w konsoli komendy 😊



Wprowadźmy sobie teraz przez J-WEB adresy na pozostałe interfejsy SRX