

# Nowości w Red Hat Enterprise Linux 8

Radosław Kujawa – [radoslaw.kujawa@osec.pl](mailto:radoslaw.kujawa@osec.pl)





# Red Hat

- ▶ 8 Maja 2019 – wydanie Red Hat Enterprise Linux 8
- ▶ Co zmienia się od strony technicznej?
- ▶ Ogromna ilość zmian – część backportowana do RHEL 7
- ▶ Zarządzanie konfiguracją
  - Klasyczna Unixowa linia komend
  - Interfejs web (cockpit)
  - Ansible – sukcesywne wdrażanie *Infrastructure as a Code*
- ▶ Satellite + Insights + Ansible = Admin 2.0
- ▶ Unixowe korzenie, nowe narzędzia

# Cockpit v185

- ▶ Zaawansowany interfejs webowy oparty na pluginach
- ▶ Zmiany wykonane za pomocą cockpita lądują we właściwych miejscach w systemie
- ▶ Wykonywanie zmian w konfiguracji przez DBus, API, biblioteki – cockpit unika opakowywania poleceń powłoki
- ▶ Standardowo włączony w RHEL 8 (poza instalacją minimal)
- ▶ Integracja z Red Hat IdM (IPA)
- ▶ `dnf install cockpit\*`
- ▶ `systemctl enable --now cockpit.socket`

# YUM v4 (dnf)

- ▶ Dandified YUM
- ▶ Python 3
- ▶ Kompatybilność wsteczna z klasycznym yum
- ▶ Modularyzacja systemu
- ▶ *Application Streams* – odpowiedź na potrzeby twórców aplikacji, aby dostarczać wspierane alternatywne wersje oprogramowania
- ▶ Wiele niezależnych od siebie wersji oprogramowania w ramach jednej dystrybucji OS
- ▶ Zachowanie kompatybilności na poziomie API i ABI w ramach konkretnej wersji strumienia
- ▶ Budowanie paczek zgodnie z ideologią SCL w dalszym ciągu obsługiwane

# YUM v4 (dnf)

- ▶ Usługi dostępne w alternatywnych wersjach
- ▶ Np. PostgreSQL 9.6 oraz 10
- ▶ Np. MariaDB 10.3 oraz MySQL 8
- ▶ Modularyzacja dotyczy nie tylko usług ale też języków programowania i frameworków
- ▶ Pojawią się strumienie z nowszymi wersjami paczek w ramach życia RHEL 8 (np. php)
- ▶ Indywidualny czas **wspierania** każdego strumienia
- ▶ System bazowy dalej objęty **10-letnim wsparciem** (+ ELS)

# YUM v4 (dnf)

- ▶ `dnf module enable postgresql:10`
- ▶ `dnf module install postgresql:10`

---

```
- name: Install PostgreSQL 10
  hosts: servers
  tasks:
    - dnf:
      name: '@postgresql:10'
```

# Konfiguracja uwierzytelniania

- ▶ `authselect` zastępuje `authconfig`
- ▶ Ułatwiona konfiguracja PAM, `nsswitch`, `sssd` oparta o profile
- ▶ Obsługa *Smart Card*, tworzenia katalogów domowych przy pierwszym logowaniu, czytników linii papilarnych...
- ▶ `authselect` **nie** zastępuje `ipa-client` dla Red Hat IdM ani `realmd` dla Microsoft Active Directory
- ▶ Kompatybilność wsteczna: `authselect-compat`, `authselect-migration(7)`

- ▶ BTRFS nie jest już wspierany, ZFS nigdy nie był... .
- ▶ **Stratis** – zarządzanie przestrzenią blokową i systemami plików w jednym narzędziu
- ▶ *Thin provisioning*, cache na dyskach SSD, snapshoty, w przyszłości redundancja
- ▶ `stratis pool create testpool /dev/vdb`
- ▶ `stratis filesystem create testpool testfs1`
- ▶ `stratis filesystem create testpool testfs2`
- ▶ `stratis filesystem list`
- ▶ `mount /stratis/testpool/testfs1 /mnttest1`
- ▶ Warstwa systemu plików zachowuje się jak XFS... bo to jest XFS



## RHEL 8 jako host wirtualizacji oraz chmur

- ▶ Nowe wersje kvm, qemu, libvirt
- ▶ cockpit preferowanym GUI do zarządzania lokalnymi VMkami
- ▶ klasyczna wirtualizacja – Red Hat Virtualization 4.x
- ▶ IaaS – Red Hat OpenStack 15
- ▶ PaaS – Red Hat OpenShift Container Platform 4

# RHEL 8 jako gość chmur IaaS i klasycznych hypervisorów

- ▶ `cloud-init` 18.2
- ▶ Customizacja obrazów RHEL dla chmur – `lorax-composer`
- ▶ Pełna obsługa deploymentów IPv6 single stack (włącznie z PXE boot)

# Budowanie własnych obrazów maszyn wirtualnych RHEL 8

- ▶ `lorax-composer` – usługa budująca obrazy, oferująca REST API
- ▶ Interfejs użytkownika – `composer-cli`, `cockpit-composer`, `welder-web`
- ▶ `systemctl enable --now lorax-composer.socket`
- ▶ `composer-cli blueprints list, push, save...`
- ▶ `composer-cli compose start example-development qcow2`

# Nowe narzędzia do konteneryzacji

- ▶ RHEL 8 nie dostarcza Dockera
- ▶ Container tools – podman, buildah, skopeo
- ▶ Silnik dla instalacji lokalnych, developmentu: podman
- ▶ Silnik dla dużej skali (Kubernetes, OpenShift): CRI-O
- ▶ Red Hat Universal Base Images (UBI)
- ▶ udica – generator polityk SELinuxa dla kontenerów (RHEL 8.1)

- ▶ Nowoczesne narzędzie do zarządzania lokalnymi kontenerami
- ▶ W pełni wspierany, to nie jest *tech preview*!
- ▶ Bardzo intensywny rozwój
- ▶ Prostota konfiguracji, brak daemonów
- ▶ Kontenery działające na kontach użytkowników nieuprzywilejowanych
- ▶ Dobra integracja z systemd
- ▶ Ułatwienie migracji do Kubernetesa (`podman generate kube`, `podman play kube`)

## podman – integracja z systemd

- ▶ Kontenery podman mogą działać jako usługi systemd
- ▶ Także na koncie nieuprzywilejowanego użytkownika
- ▶ ExecStart może uruchamiać pojedynczy kontener lub wykonywać `play kube`
- ▶ Mechanizm *socket activation* w połączeniu z kontenerami

```
[Unit]
```

```
Description=My containerized service
```

```
[Service]
```

```
ExecStart=/usr/bin/podman start -a mycontainer
```

```
ExecStop=/usr/bin/podman stop -t 10 mycontainer
```

```
[Install]
```

```
WantedBy=multi-user.target
```

# Kompatybilność wsteczna z Docker

- ▶ Paczka `podman-docker` dostarcza polecenie `docker`
- ▶ Implementacja Open Container Initiative (OCI) Runtime Specification – obrazy zbudowane w RHEL container tools działają z Dockerem i na odwrót
- ▶ `podman` nie stara się implementować API Dockera - jest oparty o własną bibliotekę `libpod`

Tabela 1: Docker vs podman compatibility vs container tools native.

docker	rhel 8 compat	rhel 8 native
docker run	podman run	podman run
docker build	podman build	buildah
docker pull	podman pull	skopeo copy
...	...	...

# Instalacja narzędzi do konteneryzacji za pomocą Ansible

---

```
- name: Install container tools
  hosts: servers
  tasks:
    - name: Install podman, buildah and skopeo
      dnf:
        name:
          - podman
          - podman-docker
          - buildah
          - skopeo
```



# Instalacja narzędzi do konteneryzacji za pomocą Ansible

```
- name: Pull images to unprivileged account
  hosts: servers
  become: false
  tasks:
    - podman_image:
      name: registry.fedoraproject.org/fedora-
        minimal
      tag: 30
    - podman_image:
      name: registry.fedoraproject.org/fedora
      tag: 30
```

# Budowanie kontenera za pomocą buildah

```
#!/bin/bash
# bc-as-a-service
# No more Dockerfile!
FEDORA_RELEASE=30
FEDORA_IMAGE=registry.fedoraproject.org/fedora-minimal:${FEDORA_RELEASE}

container=$(buildah from ${FEDORA_IMAGE})
DNF_TOOL=microdnf

buildah run $container -- $DNF_TOOL -y install nmap-ncat bc
buildah run $container -- $DNF_TOOL clean all

buildah run $container -- mkdir /logs
buildah config --volume /logs $container

buildah add $container entrypoint.sh /
buildah config --entrypoint /entrypoint.sh $container

buildah commit $container bcaas:latest
```

- ▶ `mkdir -p $HOME/tmp/logs`
- ▶ `sudo chcon -t container_file_t $HOME/tmp/logs`
- ▶ `podman run -p 7777:7777 -v $HOME/tmp/logs/://logs/localhost/bcaas`
- ▶ `-p` dla nieuprzywilejowanych kontenerów – libpod issue [2081](#)

# Budowanie obrazu kontenera za pomocą Ansible

- ▶ `ansible-bender` vs `ansible-container`?
- ▶ `ansible-bender` nie jest jeszcze dostępny jako paczka systemowa – jest w `pip` i na GitHub
- ▶ `ansible-bender build container-provisioning.yml`

# Instalacja ansible-bender



- 
- name: Install ansible
  - hosts: servers
  - tasks:
    - name: Enable ansible RHSM repo
    - rhsm\_repository:
      - name: ansible-2.8-for-rhel-8-x86\_64-rpms
    - name: Install ansible
    - dnf:
      - name: ansible

# Instalacja ansible-bender

```
- name: Install unprivileged ansible-bender and
  example playbooks
  become: false
  hosts: servers
  tasks:
    - pip:
      name: ansible-bender
      extra_args: --user
    - git:
      repo: 'https://github.com/OSEC-pl/
        osecforum-rhel8-news.git'
      dest: osecforum-rhel8-news
```

# Tworzenie obrazów kontenerów z ansible-bender

---

```
- hosts: all
  vars:
    ansible_bender:
      base_image: registry.fedoraproject.org/
        fedora:30

      target_image:
        name: myimage
        labels:
          built-by: '{{ ansible_user }}'
        environment:
          FOO: bar

  tasks:
    - dnf:
      name: mc
      state: latest
```

# Aktualizacja stosu graficznego

- ▶ Wayland domyślnym serwerem dla aplikacji graficznych
- ▶ Powrót do X11 w dalszym ciągu obsługiwany
- ▶ GNOME 3.28

Koniec...

<https://github.com/OSEC-pl/osecforum-rhel8-news>



Dziękuję!  
Czy są pytania?