Inne hypervisory i porównanie

oVirt, OLVM, XCP-ng i porównanie wszystkich mechanizmów HA.

- Inne hypervisory z HA: oVirt, Oracle Linux Virtualization Manager i XCP-ng
- Porównanie mechanizmów HA: Windows, Linux i hypervisory

Inne hypervisory z HA: oVirt, Oracle Linux Virtualization Manager i XCP-ng

Poziom: średnio

Dotyczy: oVirt 4.5.7, Oracle Linux Virtualization Manager 4.5 (hosty Oracle Linux 8 i 9), XCP-ng 8.3 LTS z Xen Orchestra i XOSTOR, VMware vSphere HA jako punkt odniesienia, klaster Hyper-V hv01 i hv02

Czas: ~60 min

Zielona ma dziś jeden hypervisor, Hyper-V, a książka opisuje jego klaster i klaster Proxmox VE. Na tej stronie przeglądamy trzy inne rodziny: oVirt, jego komercyjną odmianę od Oracle (OLVM) i XCP-ng. Dla każdej mówimy, kim jest, w jakim stanie jest projekt w październiku 2026, jak działa jej wysoka dostępność i na jakie pułapki trafia administrator, a na końcu zestawiamy je z Hyper-V i Proxmox VE.

> Nic z tej strony nie było uruchomione w Zielonej ani w naszym laboratorium. Opisujemy dokumentację producentów i projektów, a komendy z Fiszki tylko odczytują stan i podajemy je bez przykładowych wyników, bo nie mamy działającego środowiska, z którego moglibyśmy je zebrać. Stan projektów sprawdziliśmy 3 października 2026.

ZAŁOŻENIA SCENARIUSZA

Zielona nie ma dziś ani oVirt, ani XCP-ng: strona zestawia dokumentację producentów z naszym klastrem `hv01` i `hv02` (dwa hosty i świadek poza klastrem) w jednej serwerowni. Żadnego z opisanych układów nie uruchamialiśmy w laboratorium.

Skróty użyte na tej stronie (28)

Skrót	Co znaczy
BMC	Baseboard Management Controller: osobny malutki komputer na płycie serwera, który działa nawet przy wyłączonym serwerze i odczytuje m.in. temperatury czujników

Skrót	Co znaczy
CHAP	Challenge-Handshake Authentication Protocol: uwierzytelnianie hasłem przy połączeniach iSCSI; XCP-ng nie pozwala użyć takiego magazynu na heartbeat HA
CI	Continuous Integration: automatyczne budowanie i testowanie kodu po każdej zmianie; w ITIL także configuration item, czyli element konfiguracji: każdy składnik usługi zapisany w CMDB
dom0	domena sterująca w XCP-ng i Xen: specjalna maszyna na hoście, w której działają sterowniki i narzędzia zarządzania
DRBD	Distributed Replicated Block Device: replikowane urządzenie blokowe w jądrze Linuksa, w którym każdy zapis trafia na dwa węzły
ESXi	hypervisor firmy VMware instalowany na serwerze, na którym działają maszyny wirtualne zarządzane z vCenter
GFS2	Global File System 2: klastrowy system plików Linuksa, który kilka serwerów montuje naraz na tym samym dysku
GiB	gibibajt: 1024 MiB, czyli około 1,07 GB; tej jednostki używa dokumentacja MailCow przy wymaganiach pamięci
HA	High Availability: wysoka dostępność, czyli działanie mimo awarii jednego elementu
ICMP	Internet Control Message Protocol: protokół diagnostyczny sieci, którym działa między innymi ping
iSCSI	Internet Small Computer System Interface: sposób udostępniania dysku z macierzy po zwykłej sieci IP, tak że serwer widzi go jak własny dysk
KVM	Keyboard, Video, Mouse: konsola albo przełącznik, który daje jedną klawiaturę, ekran i mysz do wielu serwerów (w Linuksie KVM to też technika wirtualizacji)
LINSTOR	program do zarządzania replikowanymi woluminami DRBD na wielu serwerach, na którym stoi XOSTOR
LTS	Long Term Support: wersja z długim wsparciem poprawkami
NFS	Network File System: udostępnianie katalogów po sieci w świecie Linuksa
OLVM	Oracle Linux Virtualization Manager: komercyjnie wspierana odmiana oVirt od Oracle, której hosty działają na Oracle Linux
POSIX	Portable Operating System Interface: wspólny zestaw zasad, dzięki któremu polecenie działa tak samo na różnych Uniksach
RAM	Random Access Memory: pamięć operacyjna
RHEL	Red Hat Enterprise Linux: komercyjna dystrybucja Linuksa
SMB	Server Message Block: protokół udostępniania plików i drukarek w sieci Windows, na nim stoi zwykły udział sieciowy
UDP	User Datagram Protocol: szybki protokół sieciowy bez potwierdzeń
UEK	Unbreakable Enterprise Kernel: jądro Linuksa dostarczane przez Oracle w Oracle Linux, osobne od jądra z Red Hat

Skrót	Co znaczy
VCF	VMware Cloud Foundation: pakiet Broadcom, który łączy vSphere z magazynem, siecią i zarządzaniem w którym Broadcom sprzedaje dziś licencje vSphere
VDSM	Virtual Desktop and Server Manager: agent na każdym hoście KVM w oVirt i OLVM, który wykonuje polecenia silnika, takie jak uruchom maszynę albo podłącz magazyn
VM	Virtual Machine: maszyna wirtualna
XAPI	Xen API: interfejs, przez który narzędzia zarządzają hostami i maszynami w XCP-ng; w puli odpowiada na nim host master
XOA	Xen Orchestra Appliance: gotowa maszyna wirtualna z Xen Orchestra, z którą Vates sprzedaje wsparcie i dodatkowe funkcje
XOSTOR	magazyn hiperkonwergentny w XCP-ng: łączy lokalne dyski hostów w jeden współdzielony magazyn z replikacją przez LINSTOR i DRBD

Fiszka

Komendy wykonujemy w powłoce bash z uprawnieniami root (`sudo`), a te z `xe` na dowolnym hoście puli XCP-ng. Żadna z nich niczego nie zmienia. W Zielonej nie ma dziś ani oVirt, ani XCP-ng, więc tę Fiszkę traktujemy jako ściągę na dzień, w którym trafimy na taki system w innej firmie albo w laboratorium. Przy komendach opisujemy, na co patrzymy w wyniku, bo kształtu odpowiedzi nie znamy z własnego uruchomienia.

◫◫ TYLKO ODCZYT

Odczyt stanu silnika oVirt na hoście: hosted-engine --vm-status

```
sudo hosted-engine --vm-status
```

Dotyczy układu z silnikiem w maszynie na hostach (self-hosted engine) w oVirt i OLVM. Polecenie wypisuje sekcję dla każdego hosta ze stanem silnika (pole `Engine status`), a Oracle pokazuje je w swoim opisie kopii zapasowej silnika z filtrowaniem po tym polu [14].

◫◫ **OLVM** = Oracle Linux Virtualization Manager: komercyjnie wspierana odmiana oVirt od Oracle, której hosty działają na Oracle Linux · **hosted-engine** = silnik wbudowany: narzędzie oVirt i OLVM do wdrożenia silnika w maszynie na hostach, z opcją --vm-status, która pokazuje jego stan na każdym hoście

◫◫ TYLKO ODCZYT

Odczyt puli XCP-ng i jej mastera: xe pool-list

```
xe pool-list params=name-label,master,ha-enabled
```

Pokazuje nazwę puli, identyfikator hosta, który jest masterem, i to, czy HA jest włączone. Parametr `ha-enabled` należy do parametrów puli opisanych w dokumentacji interfejsu wiersza poleceń [18], a samo `xe` to narzędzie, którym rozmawiamy z interfejsem XAPI puli.

HA = High Availability: wysoka dostępność, czyli działanie mimo awarii jednego elementu · **XAPI** = Xen API: interfejs, przez który narzędzia zarządzają hostami i maszynami w XCP-ng; w puli odpowiada na nim host master

TYLKO ODCZYT

Odczyt hostów puli XCP-ng: xe host-list

```
xe host-list params=name-label,address,enabled
```

Wypisuje hosty z adresem zarządzania i informacją, czy host przyjmuje nowe maszyny. Host z `enabled` ustawionym na `false` jest wyłączony z przyjmowania maszyn, na przykład po `xe host-disable` [18].

TYLKO ODCZYT

Obliczenie, ile awarii hostów pula zniesie: xe pool-ha-compute-max-host-failures-to-tolerate

```
xe pool-ha-compute-max-host-failures-to-tolerate
```

Polecenie tylko liczy: dokumentacja XCP-ng mówi, że wynik uwzględnia liczbę hostów i dostępną pamięć RAM [16]. Wartość zapisaną w puli odczytujemy parametrem `ha-host-failures-to-tolerate` z `xe pool-param-list`.

RAM = Random Access Memory: pamięć operacyjna

ZMIENIA SYSTEM

Odczyt priorytetów restartu maszyn w XCP-ng: xe vm-list

```
xe vm-list is-control-domain=false params=name-label,power-state,ha-restart-priority
```

Wypisuje maszyny z priorytetem restartu przy awarii hosta. Wartości `restart`, `best-effort` i `disabled` opisuje dokumentacja HA [16], a `is-control-domain=false` pomija maszynę sterującą hosta (dom0).

HA = High Availability: wysoka dostępność, czyli działanie mimo awarii jednego elementu · **dom0** = domena sterująca w XCP-ng i Xen: specjalna maszyna na hoście, w której działają sterowniki i narzędzia zarządzania

TYLKO ODCZYT

Odczyt węzłów XOSTOR: linstor node list

```
linstor node list
```

Dotyczy tylko XCP-ng z XOSTOR. Pokazuje węzły z adresem i stanem połączenia, a kolejne polecenia z dokumentacji XOSTOR to `linstor resource list` i `linstor storage-pool list` [17].

XOSTOR = magazyn hiperkonwergentny w XCP-ng: łączy lokalne dyski hostów w jeden współdzielony magazyn z replikacją przez LINSTOR i DRBD · **linstor** = narzędzie wiersza poleceń LINSTOR: pokazuje węzły, zasoby i pule magazynu XOSTOR, na przykład linstor node list

Z życia wzięte

Z ŻYCIA WZIĘTE

Środa, 11:15. Pan Zbigniew kładzie na biurku wydruk artykułu o zmianach cen i licencji u jednego z dużych dostawców wirtualizacji i pyta, czy `hv01` i `hv02` to jedyna droga. Ola odpowiada, że klaster Hyper-V zostaje, bo działa, ale dla porządku zrobią listę innych rozwiązań z drugim hostem. Tomek dopisuje na tablicy trzy nazwy, które słyszał na forach, i od razu pyta, czy któraś z nich żyje.

Pytanie Tomka jest najważniejsze, bo wirtualizacja bez rozwoju projektu jest długiem na kilka lat. Dlatego dla każdej rodziny zaczynamy od stanu projektu, a dopiero potem od architektury. Niczego nie rekomendujemy: strona ma pokazać, jak te systemy podchodzą do tego samego problemu co klaster Hyper-V i Proxmox VE.

Jak to działa

Rozwiń: jak to działa pod spodem

Każdy hypervisor z wysoką dostępnością składa się z tych samych czterech części. Potrzebujemy elementu, który wie, gdzie co działa, hostów, na których maszyny pracują, magazynu widocznego dla wszystkich hostów i mechanizmu, który po awarii pewnie odcina host, zanim ktoś inny uruchomi jego maszyny.

> **Pojęcie: hypervisor.** Oprogramowanie, które uruchamia na jednym serwerze wiele maszyn wirtualnych i dzieli między nie procesor, pamięć i dysk. Hyper-V, KVM i Xen to hypervisory, a oVirt, OLVM i Xen Orchestra to narzędzia, którymi nimi zarządzamy.

> **Pojęcie: silnik zarządzania.** Program, który trzyma spis hostów, maszyn i magazynów oraz wydaje polecenia: uruchom, przenieś, zrestartuj po awarii. W klastrze Hyper-V jego rolę pełni sam klaster, w oVirt osobna aplikacja (Engine), a w XCP-ng jeden z hostów puli (master).

> **Pojęcie: fencing.** Pewne odcięcie hosta, o którym nie wiemy, że jest martwy, zanim jego maszyny wystartują gdzie indziej. Gdyby host żył i dalej zapisywał na dysk, a drugi uruchomił tę samą maszynę, dane byłyby uszkodzone. Szerzej o tym mówi strona Wysoka dostępność: pojęcia, liczenie dostępności i poziomy rezerwy.

Punkt odniesienia: VMware vSphere HA

Jako punkt odniesienia, bez oceny i bez rekomendacji, bierzemy vSphere HA, bo wiele pojęć z dalszych sekcji ma tam odpowiedniki. Hosty ESXi tworzą klaster, w którym jeden host jest głównym (primary), a pozostałe pomocniczymi, a host główny sprawdza pozostałe przez heartbeaty sieciowe wymieniane co sekundę i pingi ICMP na adres zarządzania [21]. Gdy host milczy w sieci, ale dalej aktualizuje heartbeat na datastore, jest uznany za odizolowany albo odcięty od hosta głównego, a nie za martwy, więc jego maszyny nie startują drugi raz [21]. HA wymaga co najmniej dwóch hostów, wspólnego magazynu i licencji na funkcję, a vSphere 8 ma koniec ogólnego wsparcia 11 października 2027, przy czym dostawca kieruje klientów do pakietów VMware Cloud Foundation (VCF) 9.x [21].

oVirt

> **Pojęcie: oVirt.** Otwarty system zarządzania maszynami wirtualnymi na KVM, który założył Red Hat jako projekt społeczności. Red Hat Virtualization (RHV) był jego komercyjną odmianą [6]. Red Hat przeniósł RHV w tryb utrzymaniowy i przestał wnosić wkład do oVirt, więc od tamtej pory projekt trzyma społeczność [4].

Stan projektu w październiku 2026

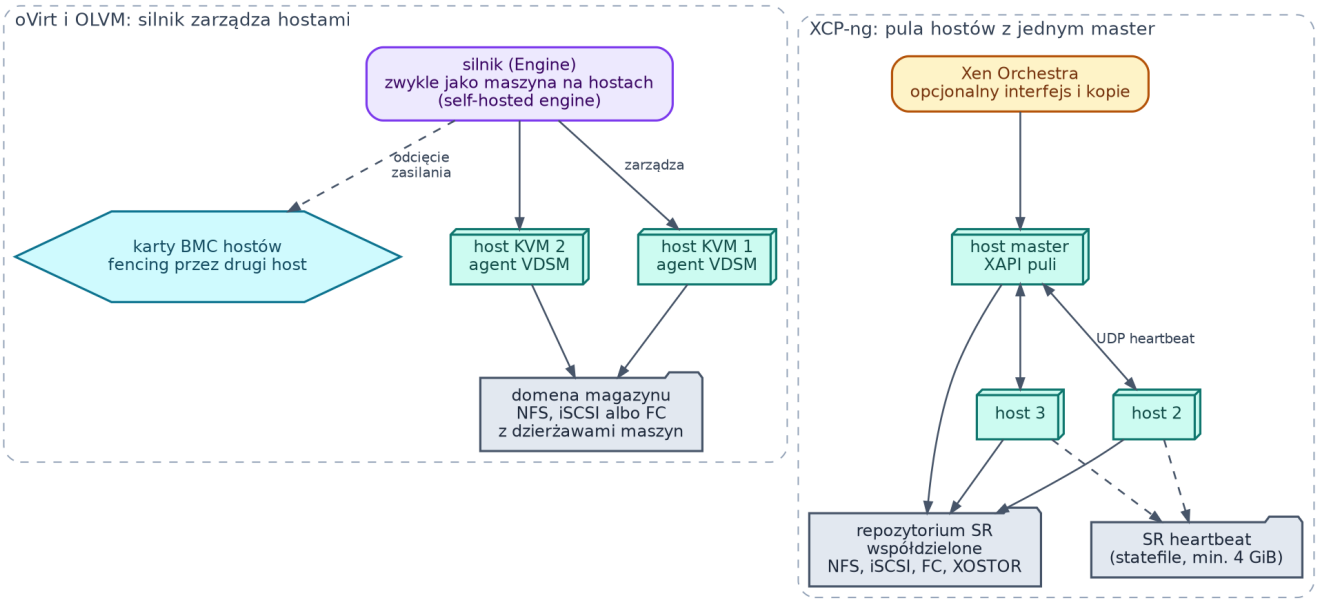
Wersja 4.5.7 została ogłoszona jako ogólnie dostępna 13 stycznia 2026, po około dwóch latach bez nowego wydania [2][4]. Obsługuje silnik i hosty na CentOS Stream 9 i 10 oraz RHEL 9 i 10 z pochodnymi (AlmaLinux 10), dodała systemy Windows 2025 i RHEL 10 do listy gości i nowy poziom klastra 4.8 [1]. Na CentOS Stream 10 wypadły Ceph, Cinderlib i collectd, a obrazy oVirt Node NG są wycofywane na rzecz innych sposobów wdrożenia [1][2].

We wrześniu 2025 zespół, który przejął rozwój, opisał naprawę CI, przejście na Javę 21 i budowanie na CentOS Stream 10 [3]. Mimo to aktywność jest mała. Na liście użytkowników w czerwcu 2026 były trzy wątki, we wrześniu jeden (pytanie o przyszłe wersje z 2 października), a lista deweloperów w sierpniu 2026 nie miała żadnego wątku [5]. Nowszego wydania niż 4.5.7 na stronie wydań nie znaleźliśmy, a pytań o następną wersję na liście użytkowników (czerwiec i październik 2026) nie rozstrzygnęliśmy, bo nie przeczytaliśmy odpowiedzi [5].

Nasza ocena: oVirt jest aktywny w sensie wydań raz na dwa lata i opieki społeczności, a nie w sensie comiesięcznych poprawek. Historii zmian w repozytorium na GitHubie nie mogliśmy odczytać w tej sesji, więc sprawdzamy ją przed jakąkolwiek decyzją.

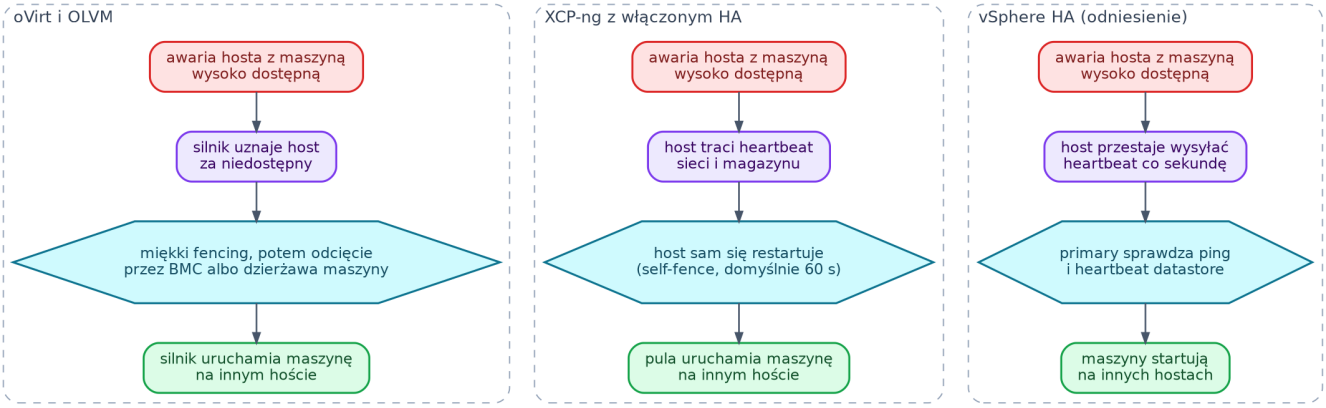
Architektura

Silnik oVirt (Engine) to aplikacja Java na serwerze WildFly z bazą PostgreSQL, a na każdym hoście KVM działa demon VDSM, który wykonuje jego polecenia [6][10]. Hosty zgrupowane są w klastry i centra danych, a magazyn to domeny magazynu: NFS (wersje 3 i 4), iSCSI, Fibre Channel, systemy plików zgodne z POSIX (GFS2), magazyn lokalny i hiperkonwergentny z Gluster [7]. W zalecanym układzie silnik działa jako maszyna na samych hostach (self-hosted engine) i jest wysoko dostępny, a dokumentacja wymaga co najmniej dwóch takich hostów [7].



Rysunek 1. Architektura oVirt, OLVM i XCP-ng. W oVirt silnik zarządza hostami z agentami VDSM, wspólną domeną magazynu i kartami BMC do fencingu. W XCP-ng pula ma jednego mastera z XAPI, hosty wymieniają heartbeat po sieci i przez osobne repozytorium SR, a Xen Orchestra jest opcjonalną nakładką. (kliknięcie otwiera pełny rozmiar)

HA maszyny włączamy na jej właściwościach (pole Highly Available), a silnik po awarii hosta uruchamia ją na innym hoście klastra [8][9]. Wymaga to skonfigurowanego zarządzania zasilaniem (power management) hostów, czyli dostępu do ich kart BMC, a odcięcia wykonuje inny host jako pośrednik, więc w klastrze lub centrum danych potrzebujemy co najmniej dwóch hostów w stanie Up albo Maintenance [8]. Silnik musi mieć pewność, że maszyna nigdzie nie działa, dlatego oVirt zna też dzierżawę maszyny (VM lease) na domenie magazynu, która pozwala uruchomić ją na innym hoście nawet po utracie zasilania przez pierwszy i chroni przed dwiema instancjami naraz [9]. Migracja na żywo działa przy pracach planowych, a przy awarii hosta maszyna startuje od nowa [8][13].



Rysunek 2. Przebieg awarii hosta z maszyną wysoko dostępną w oVirt i OLVM, w XCP-ng z włączonym HA oraz we vSphere HA jako punkcie odniesienia. Różnica leży w tym, kto podejmuje decyzję: silnik, sam host albo host główny. (kliknięcie otwiera pełny rozmiar)

Minimum, kopie i pułapki

Minimum to dwa hosty, wspólny magazyn spoza hostów i dostęp do kart BMC; w zwykłej pracy nie ma tu głosowania hostów jak w klastrze Windows, bo decyduje silnik. Przy dwóch hostach awaria jednego zostawia drugiemu cały ruch, więc podczas prac planowych na drugim hoście nie mamy żadnej rezerwy. Kopie maszyn robimy migawkami i narzędziami do kopii, a silnik zabezpieczamy osobno: poradnik Oracle pokazuje ręczną kopię dysku maszyny silnika w trybie globalnej konserwacji, a upstream ma narzędzie `engine-backup`, którego opcje sprawdzamy w dokumentacji [14].

Oracle Linux Virtualization Manager (OLVM)

> **Pojęcie: OLVM.** Komercyjnie wspierana odmiana oVirt od Oracle, oparta na tym samym silniku i agencie VDSM, ale z własnymi pakietami i hostami wyłącznie na Oracle Linux [10]. Wsparcie jest częścią subskrypcji Oracle Linux Premier Support, a według karty produktu z 2024 roku obejmuje KVM i OLVM, bez dopłaty za hosta KVM i gości Oracle Linux [13]. Terminy końca wsparcia dla wydania 4.5 podajemy jako do potwierdzenia w polityce Oracle.

Stan i wersje

Dokumentacja wydania 4.5 jest utrzymywana na bieżąco: Informacje o wydaniu mają oznaczenie z sierpnia 2026 [10]. Oracle 10 lutego 2026 dodał obsługę hostów KVM z Oracle Linux 9 (co najmniej 9.6, jądro UEK 8 oparte na Linuksie 6.12) oraz obraz silnika wbudowanego na Oracle Linux 9, z zachowaniem migracji na żywo, migawek i HA [11]. Hosty Oracle Linux 8 wymagają wydania 8.8 lub nowszego, a wszystkie hosty w środowisku muszą mieć to samo wydanie systemu, żeby HA działało [12]. Na jakiej wersji oVirt jest dziś oparty OLVM 4.5, nie ustaliliśmy: w sierpniu 2024 było to 4.5.5 [11].

Architektura i HA

Architektura jest taka jak w oVirt, więc korzystamy z rysunku 1: silnik na WildFly, agent VDSM na hostach KVM, centrum danych, klastry i domeny magazynu [10]. Magazyny to lokalny, NFS, iSCSI i Fibre Channel, a GlusterFS jest dostępny tylko na hostach Oracle Linux 8 i w układzie replica 3 [12]. Dokumentacja Oracle opisuje HA tak samo jak upstream: maszyna wysoko dostępna startuje na innym hoście po awarii, ale ręcznie wyłączony host nie przenosi jej sam, a zarządzanie zasilaniem wymaga co najmniej dwóch hostów KVM w stanie Up albo Maintenance [8]. Dochodzą dwa scenariusze odtwarzania po katastrofie, aktywny-aktywny i aktywny-pasywny, opisane w podręczniku administratora [10].

Minimum i pułapki

Minimum to dwa hosty KVM dla silnika wbudowanego [12]. Pułapki to wspólne wydanie systemu na wszystkich hostach, brak Glustera poza Oracle Linux 8 i to, że hosty OLVM działają na Oracle Linux, więc nasze serwery z Rocky Linux nie wchodzą w grę. Na czas kopii dysku silnika włączamy tryb globalnej konserwacji, który wstrzymuje automatyczne przenoszenie silnika [14].

XCP-ng z Xen Orchestra i XOSTOR

> **Pojęcie: XCP-ng.** Otwarta platforma wirtualizacji na hypervisorze Xen, rozwijana przy wsparciu firmy Vates. Hosty tworzą pulę, w której jeden host to master z interfejsem XAPI, a klienci łączą się właśnie z nim [15].

Stan projektu w październiku 2026

XCP-ng 8.3 wyszedł 7 października 2024, a status wydania z długim wsparciem (LTS) dostał 16 czerwca 2025 i ma wsparcie do 30 listopada 2028 [15]. Wydanie 8.2 LTS skończyło życie 16 września 2025 [15]. Dokumentacja wymienia Xen 4.17.5, jądro 4.19 w dom0 i bazę CentOS 7 z łatkami bezpieczeństwa przenoszonymi przez zespół, a o XCP-ng 9.0 mówi tylko jako o możliwym przyszłym wydaniu bez terminów [15]. Dokumentacja Vates pokazuje aktualizację z 2 sierpnia 2026 [19].

Dodatki: Xen Orchestra to interfejs webowy i narzędzie do kopii, XO Lite to lekka konsola jednej puli, a XOSTOR to magazyn hiperkonwergentny z lokalnych dysków hostów [19]. Licencje istnieją dwojakie: XCP-ng i Xen Orchestra ze źródeł są otwarte, a firma Vates sprzedaje subskrypcje (Essential, Essential+, Pro i Enterprise) z wsparciem i urządzeniem wirtualnym Xen Orchestra (XOA), przy czym profesjonalne wsparcie jest dostępne tylko z XOA [19]. Szczegółów licencji i cen nie podajemy.

Architektura

Pula nie wymaga osobnego serwera zarządzania: klienci łączą się z masterem przez XAPI, a Xen Orchestra jest opcjonalną nakładką [15][19]. Po utracie mastera inny host przejmuje jego rolę automatycznie [16]. Maszyny z HA muszą leżeć na magazynie współdzielonym (NFS, iSCSI, Fibre Channel albo XOSTOR) i dać się migrować na żywo, a do heartbeatu potrzebne jest osobne repozytorium SR o rozmiarze co najmniej 4 GiB, przy czym SMB i iSCSI z uwierzytelnianiem CHAP nie nadają się na heartbeat [16].

Hosty wymieniają dwa rodzaje heartbeatu: pakiety UDP przez interfejs zarządzania i zapisy do pliku stanu (`ha-statefile`) w repozytorium heartbeatu [16]. Host, który nie odzyskuje sieci albo magazynu w wyznaczonym czasie (domyślnie 60 sekund), sam się restartuje twardym wyłączeniem, a to jest jego fencing [16]. Pula liczy, ile awarii hostów zniesie, i dla każdej maszyny ma priorytet restartu: `restart`, `best-effort` albo `disabled` [16]. Migracja na żywo służy do opróżniania hosta przed pracami (`xe host-evacuate`) [18].

> **Pojęcie: XOSTOR.** Sterownik magazynu stworzony przez Vates, który łączy lokalne dyski hostów w jeden magazyn z replikacją przez LINSTOR i DRBD [17]. Wymaga co najmniej trzech hostów, bo DRBD potrzebuje kworum, pula może mieć najwyżej siedem hostów, a liczba kopii (1, 2 albo 3) jest ustalana przy tworzeniu i nie można jej zmienić [17]. Dokumentacja traktuje XOSTOR jako gotowy do produkcji i dodaje, że przed aktualizacją wersji trzeba wyłączyć HA w puli [17].

Minimum, kopie i pułapki

Dokumentacja mówi, że HA da się uruchomić na dwóch hostach, ale zdecydowanie zaleca trzy, ze względu na ryzyko split brain [16]. Przed każdą aktualizacją i restartem host przechodzi w tryb konserwacji [16]. Kopie robi Xen Orchestra: migawki rotacyjne, pełne kopie, kopie przyrostowe, replikację pełną i przyrostową, kopie lustrzane repozytorium oraz kopię metadanych puli i konfiguracji XO [20]. Które z tych typów wymagają płatnej edycji, dokumentacja nie mówi wprost, więc sprawdzamy to przed zakupem.

Zestawienie z Hyper-V i Proxmox VE

Tabele mają te same wiersze i dzielimy je na dwie połowy, bo jedenaście kolumn nie mieści się na ekranie. Dla Hyper-V i Proxmox VE bierzemy dane z naszych stron i z dokumentacji producentów, a vSphere zostaje poza tabelą, bo opisaliśmy go wyżej tylko jako punkt odniesienia. Czasy to rzędy wielkości z dokumentacji, a nie wyniki naszych testów.

Stan, zarządzanie i magazyn

System	Stan w październiku 2026	Silnik zarządzania	Magazyn współdzielony	Migracja na żywo
Hyper-V z klastrem (Zielona)	Windows Server 2022 w Zielonej, rozwijany przez Microsoft	sam klaster pracy awaryjnej, konsola i PowerShell, bez osobnego serwera	S2D (`hv01`, `hv02`), CSV	tak
Proxmox VE	wersja 9.2 z 21 maja 2026, Debian 13 [22]	brak osobnego serwera, każdy węzeł ma interfejs, stan trzyma klaster Corosync	Ceph, NFS, iSCSI, ZFS z replikacją, szczegóły w książce	tak
oVirt	4.5.7 z 13 stycznia 2026, rozwój przez społeczność, mała aktywność [1][5]	Engine (Java, WildFly, PostgreSQL), często jako maszyna na hostach	NFS, iSCSI, FC, POSIX, Gluster [7]	tak
OLVM	wydanie 4.5, dokumentacja z sierpnia 2026, hosty Oracle Linux 8 i 9 [10][11]	Engine jak w oVirt, wsparcie Oracle	lokalny, NFS, iSCSI, FC, Gluster tylko na OL8 [12]	tak [13]
XCP-ng	8.3 LTS do 30 listopada 2028 [15]	master puli z XAPI, Xen Orchestra opcjonalnie	NFS, iSCSI, FC, XOSTOR [16][17]	tak [18]

HA, czasy, kopie i pułapki

System	Minimum hostów	HA i fencing	Czas po awarii hosta	Kopie wbudowane	Typowa pułapka

Hyper-V z klastrem (Zielona)	dwa i świadek poza klastrem	maszyna jako rola klastra, bez osobnego fencingu: decyduje kworum klastra	wykrycie po 10 sekundach przy domyślnych ustawieniach [23], potem rozruch maszyny: minuty	Hyper-V Replica (DR), punkty kontrolne	świadek na maszynie z tego samego klastra
Proxmox VE	trzy, albo dwa z urządzeniem QDevice [22]	menedżer HA, host sam się restartuje przez watchdog po 60 sekundach [22]	około 2 minut według dokumentacji [22]	zadania kopii i Proxmox Backup Server	dwa węzły bez trzeciego głosu
oVirt	dwa, plus magazyn i dostęp do BMC [7][8]	HA maszyny przez silnik, power management i dzierżawa maszyny [8][9]	do zmierzenia, minuty: wykrycie, odcięcie, start	migawki, `engine-backup` dla silnika, odtwarzanie po katastrofie opisuje Oracle [10]	silnik poza układem wysoko dostępnym albo brak dostępu do BMC
OLVM	dwa [12]	jak oVirt, wspólne wydanie systemu na hostach [12]	jak oVirt	jak oVirt	różne wydania systemu na hostach, brak Glustera poza OL8
XCP-ng	dwa, zalecane trzy, XOSTOR trzy [16][17]	HA puli, heartbeat sieci i SR, samodzielny restart hosta (domyślnie 60 s) [16]	limit 60 sekund plus start maszyn	Xen Orchestra: migawki, kopie, replikacja [20]	HA na lokalnym SR, dwa hosty bez trzeciego

> Tabela nie jest rankingiem. Hyper-V jest wyborem Zielonej z powodu istniejących licencji i wiedzy zespołu, a przy zmianie hypervisora liczymy nie tylko HA, ale też migrację maszyn, kopie zapasowe i szkolenie dwóch administratorów.

Częste błędy

Błąd	Skutek	Jak unikamy
Wybór oVirt jako długoterminowego hypervisora bez sprawdzenia aktywności projektu	jedno wydanie na dwa lata i mała liczba wątków, poprawki mogą przychodzić powoli	sprawdzamy repozytorium, listy i terminy wsparcia producenta przed decyzją, a przy potrzebie umowy wsparcia bierzemy OLVM

Błąd	Skutek	Jak unikamy
HA maszyny w oVirt bez skonfigurowanych kart BMC	silnik nie może pewnie odciąć hosta, więc maszyna nie startuje na drugim	power management na każdym hoście, test odcięcia, a dla maszyn ważnych dzierżawa na domenie magazynu [8][9]
Silnik oVirt na jednej maszynie poza hostami	awaria tej maszyny zatrzymuje HA i zarządzanie, choć maszyny dalej pracują	silnik wbudowany (self-hosted engine) na co najmniej dwóch hostach [7]
Hosty OLVM na różnych wydaniach Oracle Linux	HA nie działa, bo silnik nie przeniesie maszyny na inne wydanie [12]	jedno wydanie systemu na wszystkich hostach
Pula XCP-ng z dwoma hostami i włączonym HA	split brain przy awarii sieci, o którym ostrzega dokumentacja [16]	trzy hosty albo świadome przyjęcie ryzyka
Maszyny HA w XCP-ng na lokalnym magazynie hosta	po awarii nie ma skąd ich uruchomić na drugim hoście [16]	współdzielony SR (NFS, iSCSI, FC, XOSTOR)
XOSTOR z liczbą kopii wybraną na oko	zmiany nie da się cofnąć po utworzeniu [17]	decyzja o liczbie kopii przed utworzeniem, trzy hosty minimum
Aktualizacja XCP-ng z HA włączonym i restart hosta bez trybu konserwacji	host wyłączany bez opróżnienia, możliwe samodzielne restarty innych	zawsze tryb konserwacji przed pracami [16], przed aktualizacją XOSTOR HA wyłączone [17]
Brak kopii samego silnika zarządzania	po jego utracie odbudowa środowiska trwa dniami	kopia konfiguracji (Xen Orchestra: metadane [20], oVirt: `engine-backup`) obok kopii maszyn

□□ Ćwiczenia

1. Dlaczego oVirt dla maszyny wysoko dostępnej potrzebuje kart BMC albo dzierżawy na magazynie, a XCP-ng nie potrzebuje żadnego z nich?

Odpowiedź

W oVirt decyzję podejmuje silnik na zewnątrz hosta, więc musi mieć pewność, że host nie pisze już na dysk, i zdobywa ją odcięciem zasilania przez BMC albo dzierżawą maszyny na magazynie. W XCP-ng decyzję o odcięciu podejmuje sam host: gdy przez 60 sekund (domyślnie) nie ma sieci albo magazynu, restartuje się twardo, więc pula wie, że jego maszyny nie zapisują.

1. Pula XCP-ng ma dwa hosty i włączone HA. Co mówi dokumentacja i co robimy?

Odpowiedź

Dokumentacja dopuszcza HA na dwóch hostach, ale zdecydowanie zaleca trzy, żeby uniknąć split brain. Gdy trzeciego hosta nie mamy, świadomie akceptujemy ryzyko, testujemy awarię sieci i przy pracach planowych pamiętamy, że po wyłączeniu jednego hosta nie ma już rezerwy.

1. W OLVM jeden z trzech hostów działa na Oracle Linux 8.8, a dwa na 9.6. Czy HA zadziała i co zmieniamy?

Odpowiedź

Dokumentacja Oracle wymaga tego samego wydania systemu na wszystkich hostach, żeby HA działało, a silnik wbudowany migruje się tylko na host z tym samym wydaniem. Ujednolicamy wszystkie hosty do jednej wersji (8.8 albo nowszej ósemki, albo 9.6 i nowszej) przed poleganiem na HA.

1. Pan Zbigniew pyta, czy oVirt jest „martwy”. Co odpowiadamy na podstawie tej strony?

Odpowiedź

Że projekt ma wydanie z 13 stycznia 2026 (4.5.7) i jest prowadzony przez społeczność po odejściu Red Hata, ale aktywność jest mała: jedno wydanie od około dwóch lat i kilka wątków miesięcznie na listach. Nie jest martwy, ale ryzyko długiego wsparcia jest wyższe niż w produkcie z umową producenta, więc przy potrzebie umowy bierzemy pod uwagę OLVM, a decyzję poprzedzamy przeglądem repozytorium.

Powiązane: Wysoka dostępność: pojęcia, liczenie dostępności i poziomy rezerwy · Replikacja danych między hostami: DRBD, GlusterFS, Ceph, ZFS i replikacja w aplikacji · Wzorce projektowania wysokiej dostępności: jak dobrać sposób do rodzaju usługi · Klaster Windows dla własnych usług: rola, zasób, zależności, kworum i testy przełączenia · Macierz wysokiej dostępności usług Zielonej: poziom rezerwy każdej usługi, zależności i kolejność prac · Porównanie mechanizmów HA: Windows, Linux i hypervisory · Proxmox VE: klaster trzech węzłów i HA · Proxmox VE a Hyper-V z Failover Cluster: porównanie i przenosiny · Hyper-V na co dzień: przełącznik, maszyny, punkty kontrolne i migracja

Źródła: (odczyt 3 października 2026)

1. ovirt.org: oVirt 4.5.7 Release Notes, data wydania, obsługiwane systemy, usunięte funkcje
2. blogs.ovirt.org: oVirt 4.5.7 is now generally available, 20 stycznia 2026
3. blogs.ovirt.org: oVirt Project Update, 24 września 2025

4. phoronix.com: oVirt 4.5.7 Released After Two Years With New OS & CPU Support
5. lists.ovirt.org: archiwa users@ovirt.org, czerwiec 2026, wrzesień 2026 i devel@ovirt.org, sierpień 2026; wątek Next major release of oVirt, odpowiedzi nieprzeczytane
6. en.wikipedia.org: oVirt, historia, składniki, wersja 4.5.7
7. ovirt.org: Planning and Prerequisites Guide, minimum dwóch hostów, typy magazynu, fencing
8. docs.oracle.com: High Availability and Optimization, HA maszyn, fencing, power management, minimum dwóch hostów
9. ovirt.org: Highly Available VMs i sanlock fencing; docs.redhat.com: Improving Uptime with Virtual Machine High Availability, dzierżawa maszyny
10. docs.oracle.com: OLVM Release Notes, Release 4.5 (F52195-20, sierpień 2026) i Architecture and Planning Guide
11. blogs.oracle.com: Oracle Virtualization adds support for Oracle Linux 9 compute, 10 lutego 2026, i OLVM delivers the latest oVirt community update, 14 sierpnia 2024
12. docs.oracle.com: Self-Hosted Engine Deployment i Deploy the Self-Hosted Engine
13. oracle.com: Oracle Linux KVM and Virtualization Manager, karta produktu, maj 2024
14. blogs.oracle.com: oVirt OLVM Engine VM Backup, polecenia `hosted-engine`
15. docs.xcp-ng.org: Releases i XCP-ng 8.3 LTS; xcp-ng.org: XCP-ng 8.3 is now LTS
16. docs.xcp-ng.org: High Availability
17. docs.xcp-ng.org: XOSTOR
18. docs.xcp-ng.org: CLI reference
19. docs.vates.tech: Vates VMS i VMS bundle overview; docs.xen-orchestra.com: Support and licensing
20. docs.xen-orchestra.com: Backup overview
21. techdocs.broadcom.com: Host Failure Types i vSphere HA Checklist; msspalert.com: VMware pushes customers to VCF 9.1 ahead of vSphere 8 end of support
22. pve.proxmox.com: High Availability, Cluster Manager i Roadmap
23. learn.microsoft.com: Tuning failover cluster network thresholds

Porównanie mechanizmów HA: Windows, Linux i hypervisory

Poziom: średnio

Dotyczy: Windows Failover Cluster, Hyper-V Replica, Storage Replica, Pacemaker i Corosync, keepalived i HAProxy, DRBD, Ceph, ZFS, Proxmox VE, oVirt i OLVM, XCP-ng, Docker Swarm, k3s, replikacja baz

Czas: ~40 min

Ta strona zbiera w jednej tabeli wszystkie mechanizmy wysokiej dostępności opisane w książce i mówi, co każdy z nich chroni, ile ma minimum węzłów, jak rozstrzyga spór i gdzie o nim czytamy. Pod tabelą stoi przewodnik „jeśli masz to, wybierz tamto". Wracamy tu, gdy ktoś pyta, który mechanizm pasuje do konkretnego układu serwerów, i przed wpisem w macierz usług.

> Liczby w tabeli to rzędy wielkości z dokumentacji producentów albo z prób w piaskownicy opisanych na stronach książki. Nie są to pomiary Zielonej ani obietnice, a prawdziwy czas przełączenia poznajemy tylko z testu.

ZAŁOŻENIA SCENARIUSZA

porównanie jest ogólne, a Zielona to w nim jedna serwerownia z dwoma hostami `hv01` i `hv02`, trzecim głosem na fizycznym `bkp01` i bez drugiej lokalizacji. Mechanizmy działające między miejscami, jak Hyper-V Replica, zakładają łącze do osobnej lokalizacji, na przykład do

Swarzędza: IPsec site-to-site: Pamiątkowo i Swarzędz między FortiGate a routerem innej firmy.

Skróty użyte na tej stronie (10)

Skrót	Co znaczy
API	Application Programming Interface: sposób, w jaki programy rozmawiają ze sobą

Skrót	Co znaczy
Corosync	usługa komunikacji klastra, która pilnuje członkostwa węzłów i kworum; Proxmox VE i Pacemaker używają jej do sprawdzania, które węzły żyją
DRBD	Distributed Replicated Block Device: replikowane urządzenie blokowe w jądrze Linuksa, w którym każdy zapis trafia na dwa węzły
etcd	etc distributed: rozproszony magazyn par klucz-wartość, w którym Kubernetes trzyma cały stan klastra, a zmiany uzgadnia protokołem Raft
HAProxy	High Availability Proxy: program, który rozdziela połączenia między kilka serwerów i omija te, które nie odpowiadają
OLVM	Oracle Linux Virtualization Manager: komercyjnie wspierana odmiana oVirt od Oracle, której hosty działają na Oracle Linux
QDevice	Quorum Device: trzeci głos dla klastra z dwoma węzłami, działający na osobnym hoście; w Proxmox VE i Pacemakerze rozstrzyga remis
S2D	Storage Spaces Direct: programowa macierz Windows Server z dysków wielu serwerów
UPS	Uninterruptible Power Supply: zasilacz awaryjny, podtrzymuje zasilanie urządzeń na czas zaniku napięcia z sieci
ZFS	system plików z migawkami i replikacją przyrostową zbudowany w jednym narzędziu z menedżerem wolumenów (dawniej Zettabyte File System)

Z życia wzięte

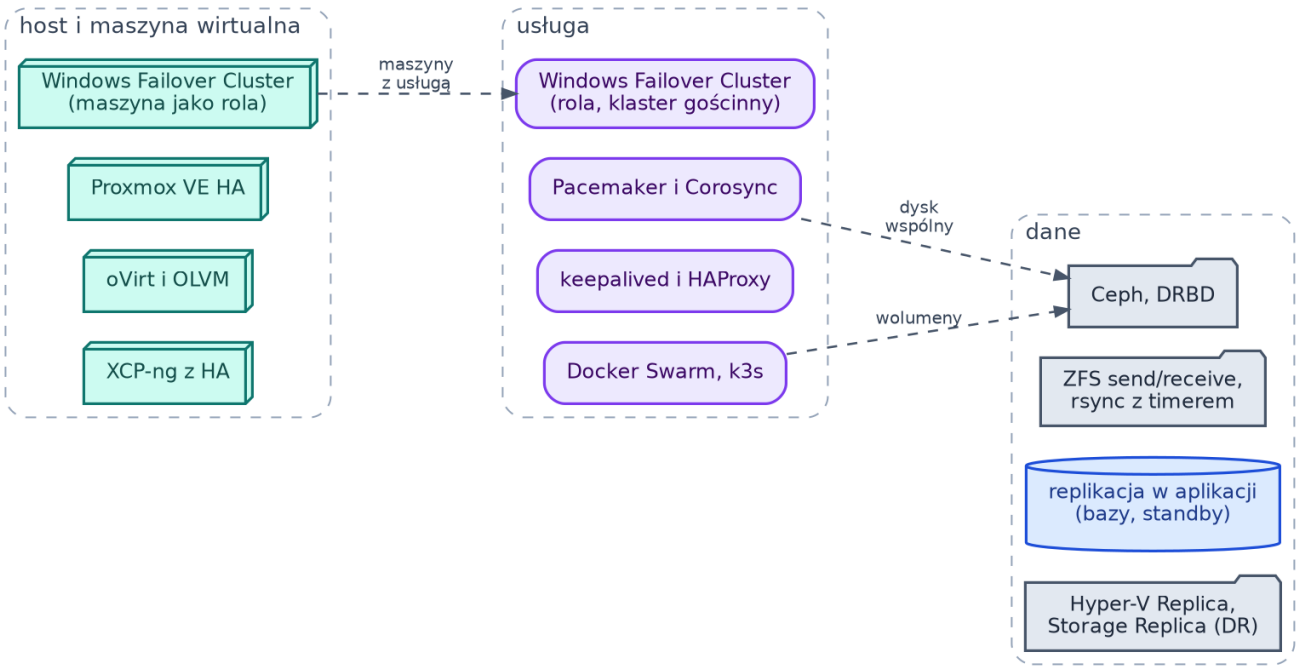
> **□□ Z ŻYCIA WZIĘTE**
>
> Wtorek, 14:05. Pan Zbigniew przychodzi po tygodniu czytania całej książki o wysokiej dostępności i prosi o jedną kartkę: który z tych mechanizmów jest do czego. Tomek wymienia z pamięci cztery nazwy i wszystkie myli, a Ola kreśli na tablicy tabelę z kolumnami „co chroni”, „ile węzłów” i „kto rozstrzyga spór”. Gdy tabela ma szesnaście wierszy, pan Zbigniew mówi, że rozumie już, dlaczego odpowiedź na pytanie „zróbmy klaster” brzmi „jaki”.

Kartka z wtorku jest tą stroną. Nie wybiera za nas, tylko pokazuje, że każdy mechanizm odpowiada na inne pytanie i ma inną cenę.

Jak czytać tabelę

Mechanizmy łatwo pomylić, bo wszystkie obiecują „drugi serwer przejmie pracę”. Różnią się tym, co chronią: samego hosta, całą maszynę, usługę albo dane. Różnią się też tym, kto po awarii

decyduje, że węzeł nie żyje, i czy potrafi go pewnie odciąć.



Rysunek 1. Mapa mechanizmów według warstwy, którą chronią. Hypervisory chronią host i maszynę wirtualną, klastry usług i równoważenie chronią usługę, a replikacja chroni dane. Warstwy się nakładają: maszyny z usługą stoją na hypervisorze, a usługa potrzebuje danych. (kliknięcie otwiera pełny rozmiar)

Pojęcie: warstwa ochrony. Część układu, której awarię mechanizm potrafi przeżyć: host, system w maszynie, usługa albo dane. Mechanizm z jednej warstwy nie widzi awarii w innej, na przykład klaster hostów nie wie, że system w maszynie się zawiesił.

Pojęcie: czas przełączenia. Czas od awarii do pierwszej poprawnej odpowiedzi usługi. Składa się z wykrycia awarii, pewnego odcięcia węzła (fencing) i startu usługi na drugim węźle, a dla maszyn także z rozruchu systemu.

Pojęcie: kworum i świadek. Kworum to zasada większości głosów, a świadek to dodatkowy głos spoza węzłów, który rozstrzyga remis. Szerzej opisuje je strona Wysoka dostępność: pojęcia, liczenie dostępności i poziomy rezerwy.

Tabela ma szesnaście wierszy, więc dzielimy ją na dwie połowy z tymi samymi wierszami. Pierwsza mówi, co mechanizm chroni i ile kosztuje w węzłach, druga, jak rozstrzyga spór i czego potrzebuje do pracy.

Co chroni, jak szybko i ile węzłów

Mechanizm	System	Co chroni	Czas przełączenia (rząd)	Minimum węzłów	Gdzie w książce
Windows Failover Cluster: maszyna wirtualna	Windows Server z Hyper-V	host: maszyna startuje na drugim hoście, wnętrza systemu nie widzi	planowo migracja na żywo, pauza rzędu sekund; po awarii wykrycie po 10 sekundach przy domyślnych ustawieniach [1] i rozruch maszyny: minuty	dwa hosty i świadek	Perforce i Lore w klastrze: cztery drogi, rekomendacja dla Zielonej i test przełączenia, Klaster Windows od zera: budowa, sieć, świadek i S2D, Klaster Windows w utrzymaniu: aktualizacje CAU, monitoring i problemy
Windows Failover Cluster: rola z usługą (klaster gościnny)	Windows Server	usługę z adresem, nazwą i dyskiem	sekundy do minuty, połączenia zrywane	dwie maszyny gościnne na różnych hostach i świadek	Klaster Windows dla własnych usług: rola, zasób, zależności, kworum i testy przełączenia
Hyper-V Replica (odtwarzanie po katastrofie)	Windows Server z Hyper-V	kopię maszyny na drugim hoście, zwykle w innym miejscu; nie przejmuje pracy sama	start ręczny: godziny; dane sprzed 30 sekund, 5 lub 15 minut [2]	dwa hosty (źródło i cel)	Perforce i Lore: porównanie dróg do wysokiej dostępności na Windowsie

Mechanizm	System	Co chroni	Czas przełączenia (rząd)	Minimum węzłów	Gdzie w książce
Storage Replica (odtwarzanie po katastrofie)	Windows Server	wolumen: replikacja synchroniczna (do 5 ms) albo asynchroniczna [2]	start ręczny: godziny	dwa serwery lub dwa klastry	Perforce i Lore: porównanie dróg do wysokiej dostępności na Windowsie
Pacemaker i Corosync	Linux (Rocky Linux 9)	usługę i dane na wspólnym dysku	dziesiątki sekund	dwa węzły i trzeci głos (qnetd na `bkp01`)	Pacemaker na Linuksie: własna usługa z adresem wirtualnym, Pacemaker w utrzymaniu: zarządzanie, testy i aktualizacje
keepalived i HAProxy	Linux	usługę bezstanową i adres wirtualny	sekundy: około 3 sekund po utracie ogłoszeń VRRP i około 9 sekund po trzech porażkach skryptu (piaskownica)	dwie maszyny równoważące i dwie instancje usługi	Równoważenie i adres wirtualny: keepalived i HAProxy na parze lb01 i lb02
rsync z timerem i keepalived (ciepła rezerwa Dockera)	Linux z Dockerem	pliki aplikacji na drugim hoście i adres wirtualny	minuty; dane sprzed 15 minut	dwa hosty	Przenoszenie kontenerów z docker01 na docker02: zimna i ciepła rezerwa, przełączenie
DRBD	Linux, moduł spoza dystrybucji	dane na poziomie bloków	samo nie przełącza: promocja i montowanie ręcznie albo przez Pacemaker	dwa węzły i trzeci głos	Replikacja danych między hostami: DRBD, GlusterFS, Ceph, ZFS i replikacja w aplikacji

Mechanizm	System	Co chroni	Czas przełączenia (rząd)	Minimum węzłów	Gdzie w książce
Ceph	Linux	dane: obiekty, bloki (RBD), pliki (CephFS), samonaprawa	OSD uznany za niedziałający po około 20 sekundach, usunięty z puli po 10 minutach (ustawienia domyślne) [3]	trzy węzły i trzy monitory	Replikacja danych między hostami: DRBD, GlusterFS, Ceph, ZFS i replikacja w aplikacji, Proxmox VE: klaster trzech węzłów i HA
ZFS send/receive	Linux z modułem OpenZFS	migawki zbioru na drugim hoście	przełączenie ręczne; dane sprzed minut (częstość migawek)	dwa hosty	Replikacja danych między hostami: DRBD, GlusterFS, Ceph, ZFS i replikacja w aplikacji, Przenoszenie kontenerów z docker01 na docker02: zimna i ciepła rezerwa, przełączenie
Proxmox VE HA	Linux (Debian)	host: maszyny i kontenery startują na innym węźle	około 2 minut według dokumentacji [4]	trzy węzły albo dwa z urządzeniem QDevice	Proxmox VE: klaster trzech węzłów i HA, Proxmox VE w utrzymaniu: kopie PBS, aktualizacje, użytkownicy, Proxmox VE a Hyper-V z Failover Cluster: porównanie i przenosiny

Mechanizm	System	Co chroni	Czas przełączenia (rząd)	Minimum węzłów	Gdzie w książce
oVirt i OLVM	Linux (CentOS Stream, RHEL, Oracle Linux)	host: silnik uruchamia maszyny wysoko dostępne na innym hoście	minuty, do zmierzenia: wykrycie, odcięcie, start	dwa hosty, wspólny magazyn i dostęp do kart BMC [5]	Inne hypervisory z HA: oVirt, Oracle Linux Virtualization Manager i XCP-ng
XCP-ng z HA	Linux (Xen)	host: pula uruchamia maszyny HA na innym hoście	limit 60 sekund (domyślnie) plus start maszyn [6]	dwa hosty, zalecane trzy	Inne hypervisory z HA: oVirt, Oracle Linux Virtualization Manager i XCP-ng
Docker Swarm	Linux z Dockerem	usługi kontenerowe: zadania startują na innych węzłach	heartbeat dispatchera co 5 sekund [7] plus start zadań, do zmierzenia	trzech menedżerów	Docker Swarm: klaster z trzema menedżerami, na którym przenosimy kolorowy nginx
k3s (Kubernetes) z etcd	Linux	pody i płaszczyznę sterowania	węzeł `NotReady` po około 40 sekundach, pody odtwarzane po 300 sekundach (ustawienia domyślne)	trzy serwery z etcd	k3s z trzema węzłami serwerowymi: etcd, adres wirtualny API i Longhorn
Replikacja w aplikacji (bazy, standby)	PostgreSQL, MongoDB, SQL Server, Perforce i inne	dane i usługę bazy	awans ręczny: czas decyzji; automat Patroni: domyślnie ttl 30 sekund [8]	dwa węzły i trzeci głos (arbiter, świadek, magazyn uzgadniania)	Replikacja danych między hostami: DRBD, GlusterFS, Ceph, ZFS i replikacja w aplikacji, Macierz wysokiej dostępności usług Zielonej: poziom rezerwy każdej usługi, zależności i kolejność prac

Jak rozstrzyga spór i czego potrzebuje

Mechanizm	Kworum i świadek	Fencing	Magazyn	Złożoność
Windows Failover Cluster: maszyna wirtualna	większość głosów, świadek: udział poza klastrem, dysk albo Cloud Witness	bez osobnego fencingu urządzeniami, decyduje kworum klastra	S2D i CSV na `hv01` i `hv02`	niska do średniej
Windows Failover Cluster: rola z usługą	świadek klastra gościnnego poza klastrem hostów (`bkp01`)	rezerwacja dysku klastrowego (wspólny VHD Set)	wspólny dysk VHD Set na S2D	wysoka
Hyper-V Replica	brak	brak	osobny magazyn na każdym hoście	średnia
Storage Replica	poza klastrem rozciągniętym nie dotyczy (sprawdzamy w dokumentacji)	brak	wolumeny po obu stronach	średnia
Pacemaker i Corosync	urządzenie kworum na trzecim hoście, algorytm `ffsplit`	obowiązkowy: `fence_scsi` na wspólnym dysku albo karty BMC; dla maszyn na Hyper-V do potwierdzenia testem	wspólny dysk (HA-LVM) albo DRBD	wysoka
keepalived i HAProxy	brak: priorytet VRRP, dwa węzły mogą przejściowo trzymać ten sam adres	brak, zabezpieczeniem są skrypty i decyzja człowieka	nie dotyczy, usługa bezstanowa	niska do średniej
rsync z timerem i keepalived	brak	brak, bezpieczniki w skrypcie synchronizacji	dyski lokalne obu hostów	niska
DRBD	kworum DRBD 9 z bezdyskowym węzłem głosującym (tiebreaker) [9]	potrzebny w klastrze (Pacemaker), samo DRBD nie odcina	dyski lokalne replikowane po sieci	wysoka
Ceph	większość monitorów	nie dotyczy: spójność daje protokół magazynu	dyski OSD, zalecana sieć 10 Gb	bardzo wysoka
ZFS send/receive	brak	brak	pule ZFS na obu hostach	średnia

Mechanizm	Kworum i świadek	Fencing	Magazyn	Złożoność
Proxmox VE HA	kworum Corosync, QDevice jako trzeci głos [4]	watchdog: węzeł bez kworum restartuje się po 60 sekundach [4]	Ceph, magazyn współdzielony albo replikacja ZFS	średnia
oVirt i OLVM	w zwykłej pracy decyduje silnik, nie głosowanie hostów	karty BMC (pośrednikiem jest drugi host) i dzierżawa maszyny na magazynie [5]	NFS, iSCSI, FC, POSIX	wysoka
XCP-ng z HA	heartbeat sieci i repozytorium SR zamiast głosowania	host sam się restartuje twardo [6]	wspólne SR (NFS, iSCSI, FC) albo XOSTOR	średnia
Docker Swarm	większość menedżerów (Raft), trzech znosi utratę jednego	brak	nie rozwiązuje danych, trzeba osobno	średnia
k3s (Kubernetes) z etcd	większość członków etcd, trzech znosi utratę jednego	brak	Longhorn (w naszym układzie dwie repliki) albo magazyn zewnętrzny	wysoka
Replikacja w aplikacji	arbiter, świadek albo magazyn uzgadniania przy automatycznym awansie	zwykle zbędny, zależy od narzędzia	każdy węzeł ma własne dane	średnia

Trzy zdania z tabel zasługują na przypomnienie osobno. Wspólny dysk wymaga fencingu, bo węzeł, który się zawiesił, może dalej pisać na dysk. Dwa węzły bez trzeciego głosu nie rozstrzygną sporu. Trzy węzły na dwóch hostach hypervisora nie dają odporności na awarię hosta, bo host z dwoma węzłami zabiera większość głosów.

Przewodnik: jeśli masz, wybierz

Przewodnik zaczyna od układu, który Zielona ma dziś, a potem idzie do układów, które spotykamy gdzie indziej. Dla każdego podajemy mechanizm domyślny i warunek, przy którym zmieniamy zdanie. Dobór według rodzaju usługi (stanowa, bezstanowa, baza) opisuje strona Wzorce projektowania wysokiej dostępności: jak dobrać sposób do rodzaju usługi, a tu patrzymy od strony sprzętu i systemu.

Jeśli masz	Wybieramy	Dlaczego i kiedy zmieniamy zdanie

Dwa hosty Windows z S2D (`hv01`, `hv02`)	maszyny jako role klastra Hyper-V i świadek poza klastrem na `bkp01`	klaster hostów chroni przed awarią sprzętu; gdy przerwa po awarii maszyny za długa dla usługi, dokładamy klaster gościnny z dwóch maszyn (wd2-01)
Te same dwa hosty i potrzebę kopii w drugim miejscu	Hyper-V Replica albo Storage Replica do trzeciego miejsca	to odtwarzanie po katastrofie, a nie HA, i nie zastępuje kopii zapasowej
Trzy hosty linuksowe i maszyny wirtualne	Proxmox VE z HA, a Ceph jako magazyn, jeśli sieć i dyski to udźwignią	trzy węzły dają kworum bez świadka; oVirt albo XCP-ng wybieramy z powodów spoza HA, na przykład wsparcia Oracle albo istniejącej wiedzy
Trzy hosty linuksowe i same kontenery	k3s z trzema serwerami na trzech hostach	trzy serwery na dwóch hostach nie przeżyją awarii hosta; Swarm tylko dla aplikacji bezstanowych, gdy zespół zna Dockera
Dwa hosty z Dockerem i aplikacje trzymające dane w plikach	rsync z timerem, restic na `bkp01` i keepalived z adresem wirtualnym	prosta ciepła rezerwa z danymi sprzed kwadransa; ZFS send/receive, gdy RPO ma spaść do minut
Jedna usługa stanowa na dwóch węzłach Linuksa	Pacemaker z trzecim głosem na `bkp01` i fencingiem	bez fencingu nie wdrażamy; gdy fencingu nie da się zrobić, maszyna jako rola klastra hypervisora jest prostsza
Usługa bezstanowa (serwer WWW, aplikacje Java za HAProxy)	keepalived i HAProxy z dwiema instancjami	niewidoczne wyłączenie jednej instancji, sekundy po awarii; dane trzymamy w bazie
Baza danych (PostgreSQL, MongoDB, SQL Server)	replikacja natywna z arbitrem albo świadkiem	silnik zna potwierdzone zapisy; awans automatyczny (na przykład Patroni) wymaga trzeciego głosu, a ręczny przynajmniej zdecydowanej osoby
Dane na wspólnym dysku bez klastra usług	DRBD z tiebreakerem albo ZFS send/receive	DRBD dla RPO bliskiego zeru i z Pacemakerem do przełączenia, ZFS dla prostszej ciepłej rezerwy
Potrzebę sekund, a nie minut	keepalived, replikacja natywna baz, klaster usług	restart maszyny po awarii hosta to minuty; sekundy daje tylko mechanizm wyżej niż hypervisor
Druga lokalizacja i brak potrzeby automatu	Hyper-V Replica, ZFS send/receive albo replikacja asynchroniczna bazy	start repliki wykonuje człowiek; ustalamy, kto i kiedy

Przewodnik nie zastępuje testu. Gdy wybieramy mechanizm, ustalamy też, kto go uruchomi o 20:00 w piątek i kto sprawdzi, że kopia zapasowa dalej istnieje.

Gdy mechanizmy nakładają się

W Zielonej prawie każda usługa stoi w warstwach, więc mechanizmy z tabeli działają jednocześnie. Maszyna `pcmk01` jest rolą klastra Hyper-V, a w środku ma Pacemaker. Baza `pg01` ma replikację natywną, a sama stoi na S2D. Dobrze jest wiedzieć, które pary ze sobą współpracują, a które sobie przeszkadzają.

Para warstw	Co się dzieje	Co robimy
maszyna jako rola klastra Hyper-V i Pacemaker w gościu	migracja na żywo wstrzymuje maszynę na chwilę, która może przekroczyć limity czasu Corosync	sprawdzamy na laboratorium; przed pracami na hoście wstrzymujemy węzeł (`pcs node standby`) Pacemaker na Linuksie: własna usługa z adresem wirtualnym
k3s z trzema węzłami na dwóch hostach Hyper-V	awaria hosta z dwoma węzłami zabiera kworum etcd do czasu restartu maszyn	trzeci fizyczny host albo świadome przyjęcie ryzyka k3s z trzema węzłami serwerowymi: etcd, adres wirtualny API i Longhorn
Longhorn nad S2D	S2D trzyma już dwie kopie bloków, więc repliki Longhorna mnożą zajętość	dwie repliki, nie trzy
replikacja bazy w maszynie klastra	klaster restartuje maszynę, a replika ma własny awans; dwa mechanizmy mogą się rozmijać	jeden mechanizm decyduje o awansie, drugi tylko restartuje maszynę, i opisujemy to w macierzy
klaster hostów i dwa kontrolery domeny na nim	rezerwa kontrolerów nie chroni przed awarią klastra, który je niesie	kontrolery na różnych hostach, kopia i awaryjne konto Macierz wysokiej dostępności usług Zielonej: poziom rezerwy każdej usługi, zależności i kolejność prac

Odniesienie do macierzy wd4-01

Po wyborze mechanizmu zapisujemy go w macierzy Macierz wysokiej dostępności usług Zielonej: poziom rezerwy każdej usługi, zależności i kolejność prac, w kolumnie „mechanizm docelowy”. Macierz pamięta poziom rezerwy (zimna, ciepła, gorąca) i to, ile dziś naprawdę mamy, a ta strona dobiera mechanizm do tego poziomu.

Zasada z macierzy obowiązuje bez zmian: poziom usługi nie przekracza poziomu jej najsłabszej zależności i podnosimy go dopiero po teście przełączenia. Gitea z ciepłą rezerwą na `docker02` i bazą na `pg01` bez repliki zostaje w macierzy zimna, bez względu na to, który mechanizm z tabeli wyżej wybraliśmy dla plików.

Częste błędy

Błąd	Skutek	Jak unikamy
Porównujemy mechanizmy po samym czasie przełączenia	wybieramy najszybszy, który nie chroni tego, co potrzebujemy	najpierw warstwa ochrony (host, usługa, dane), potem czas
Dwa węzły i brak trzeciego głosu w mechanizmie z kworum	po zerwaniu łączności żaden nie ma większości albo oba uznają się za aktywne	świadek, qdevice, arbiter albo trzeci węzeł, zależnie od mechanizmu
Wspólny dysk i klaster usług bez fencingu	zawieszony węzeł pisze na dysk razem z drugim, a system plików ulega uszkodzeniu	fencing obowiązkowy albo mechanizm bez wspólnego dysku
Trzy węzły na dwóch hostach uznane za odporność na awarię hosta	host z dwoma węzłami zabiera większość głosów	trzy hosty albo świadome przyjęcie ryzyka
Odtwarzanie po katastrofie uznane za HA	replika startuje ręcznie, a dane są sprzed 30 sekund do kwadransa	w macierzy opisujemy to jako DR i nie obiecujemy nim RTO
Replika zamiast kopii zapasowej	skasowanie i ransomware docierają do rezerwy w sekundy	kopia restic, RMAN albo pgBackRest zostaje obok każdego mechanizmu
Czas z tabeli wpisany do umowy z właścicielem usługi	w dniu awarii przerwa jest dłuższa niż obiecana	wpisujemy czas z naszego testu przełączenia, najgorszy z kilku pomiarów
Dwa mechanizmy decydują o tym samym zdarzeniu	klaster hostów restartuje maszynę, a klaster usług w niej przenosi rolę, więc przerwa się mnoży	opisujemy, kto decyduje, a drugi mechanizm ustawiamy tylko jako zabezpieczenie

□□ Ćwiczenia

1. Dwa hosty Windows z S2D i jedna usługa, której przerwa po awarii hosta nie może przekroczyć minuty. Który mechanizm z tabeli bierzemy najpierw i co dokładamy, gdy minuta nie wystarcza?

Odpowiedź

Najpierw maszyna jako rola klastra Hyper-V ze świadkiem poza klastrem, bo chroni przed awarią hosta i dla maszyny daje migrację na żywo przy pracach planowych. Po awarii hosta maszyna startuje od nowa, co daje minuty, więc gdy minuta nie wystarcza, dokładamy klaster gościnny z dwóch maszyn na różnych hostach, w którym rola przechodzi w sekundach do minuty. Wyboru nie podejmujemy bez zmierzenia przerwy w teście.

1. Dlaczego Hyper-V Replica i Storage Replica stoją w tabeli, choć nie przejmują pracy same?

Odpowiedź

Bo to mechanizmy odtwarzania po katastrofie z drugiej lokalizacji, a nie wysokiej dostępności. Chronią dane przed utratą miejsca albo hosta, ale start repliki wykonuje człowiek (godziny), a dane są sprzed 30 sekund do 15 minut (Hyper-V Replica) albo bliskie zeru (Storage Replica synchroniczna). Stoją w tabeli, żeby nikt nie wpisał ich do macierzy jako gorącej rezerwy.

1. Mamy trzy hosty linuksowe i chcemy trzech węzłów etcd dla k3s. Jak ułożyć węzły i co zmienia się, gdy hostów są tylko dwa?

Odpowiedź

Po jednym węźle na każdym z trzech hostów, wtedy awaria hosta zostawia dwa głosy z trzech i klaster działa dalej. Przy dwóch hostach jeden z nich trzyma dwa węzły, a jego awaria zabiera większość głosów, więc API przestaje przyjmować zmiany do czasu, aż hypervisor uruchomi maszyny na drugim hoście. Przy dwóch hostach lepiej wybrać prostszy mechanizm, na przykład rsync z keepalived.

1. Dla jakiej pary mechanizmów w tabeli rozstrzygnięcie sporu zależy od tego samego trzeciego elementu, jakim jest `bkp01`?

Odpowiedź

Świadek klastra Hyper-V (udział na `bkp01`, propozycja z macierzy) i trzeci głos klastra Pacemakera (qnetd na `bkp01`). Obie funkcje może pełnić ten sam fizyczny serwer poza klastrem hostów, więc jego awaria odbiera jednocześnie świadka i trzeci głos, a on sam stoi na tym samym UPS co serwery. Dlatego `bkp01` jest w macierzy punktem awarii, który trzeba zabezpieczyć.

Powiązane: Wysoka dostępność: pojęcia, liczenie dostępności i poziomy rezerwy · Przenoszenie kontenerów z docker01 na docker02: zimna i ciepła rezerwa, przełączenie · Replikacja danych między hostami: DRBD, GlusterFS, Ceph, ZFS i replikacja w aplikacji · Wzorce projektowania wysokiej dostępności: jak dobrać sposób do rodzaju usługi · Klaster Windows dla własnych usług: rola, zasób, zależności, kworum i testy przełączenia · Pacemaker na Linuksie: własna usługa z adresem wirtualnym · Równoważenie i adres wirtualny: keepalived i HAProxy na parze lb01 i lb02 · Docker Swarm: klaster z trzema menedżerami, na którym przenosimy kolorowy nginx · k3s z trzema węzłami serwerowymi: etcd, adres wirtualny API i Longhorn · Macierz wysokiej dostępności usług Zielonej: poziom rezerwy każdej usługi, zależności i kolejność prac · Inne hypervisory z HA:

oVirt, Oracle Linux Virtualization Manager i XCP-ng · Proxmox VE: klaster trzech węzłów i HA · Proxmox VE w utrzymaniu: kopie PBS, aktualizacje, użytkownicy · Proxmox VE a Hyper-V z Failover Cluster: porównanie i przenosiny · Klaster Windows od zera: budowa, sieć, świadek i S2D · Klaster Windows w utrzymaniu: aktualizacje CAU, monitoring i problemy · Pacemaker w utrzymaniu: zarządzanie, testy i aktualizacje

Źródła: (odczyt 3 października 2026)

1. learn.microsoft.com: Tuning failover cluster network thresholds, domyślnie 1000 ms razy 10 pulsów
2. learn.microsoft.com: Hyper-V Replica overview i Storage Replica overview, za Perforce i Lore: porównanie dróg do wysokiej dostępności na Windowsie
3. docs.ceph.com: Configuring Monitor/OSD Interaction, `osd_heartbeat_grace` 20 sekund, `mon_osd_down_out_interval` 10 minut
4. pve.proxmox.com: High Availability i Cluster Manager, trzy węzły albo QDevice, watchdog po 60 sekundach, około 2 minut
5. ovirt.org: Planning and Prerequisites Guide; docs.oracle.com: High Availability and Optimization
6. docs.xcp-ng.org: High Availability
7. docs.docker.com: docker swarm update, `--dispatcher-heartbeat` domyślnie 5 sekund
8. patroni.readthedocs.io: Dynamic configuration settings, `ttl` 30, `loop_wait` 10, `retry_timeout` 10
9. LINBIT: DRBD 9 User Guide, za Replikacja danych między hostami: DRBD, GlusterFS, Ceph, ZFS i replikacja w aplikacji
10. Ta baza wiedzy: strony wd1-01 do wd4-01 i wd8-01 z kolumny „Gdzie w książce”, opisy testów i piaskownicy na stronach wd3-01, wd3-02 i wd3-03