Poznajemy Home Assistant

Opowieść i wybór sposobu instalacji.

- Home Assistant: lokalny system dla serwerowni i domu
- Instalacja Home Assistant: system na maszynie wirtualnej, mini PC albo w kontenerze

Home Assistant: lokalny system dla serwerowni i domu

Poziom: podstawy **Dotyczy: Home Assistant 2026.9, ESPHome 2026.9**

Czas: ~10 min

Ta strona otwiera książkę o Home Assistant: skąd wziął się ten system, czym jest ESPHome i dlaczego Zielona woli trzymać dane z czujników u siebie, a nie w chmurze producenta. Ten sam program i te same płytki ESP32 działają w serwerowni i w domu, więc każda strona książki ma też wątek domowy. Stan wiedzy: październik 2026.

Skróty użyte na tej stronie (6)

Skrót	Co znaczy
API	Application Programming Interface: sposób, w jaki programy rozmawiają ze sobą
ESP32	Tani mikrokontroler z Wi-Fi i Bluetooth firmy Espressif: malutki komputer, z którego budujemy własne czujniki sieciowe
ESPHome	Oprogramowanie, które zamienia ESP32 w gotowy czujnik sieciowy bez pisania kodu od zera, a odczyty udostępnia przez sieć
MIT	MIT License: bardzo liberalna licencja wolnego oprogramowania, pozwala na dowolne użycie i modyfikację kodu
OS	Operating System: system operacyjny
YAML	YAML Ain't Markup Language: format plików konfiguracyjnych oparty na wcięciach

Skąd się wzięło

> **Pojęcie: Home Assistant.** Wolny program, który zbiera w jednym miejscu urządzenia z różnych źródeł (czujniki, przełączniki, lampy, liczniki energii) i pozwala nimi sterować z przeglądarki albo z telefonu. Działa na własnym serwerze i decyzje, czyli automatyzacje,

> podejmuje lokalnie.

Paulus Schoutsen zaczął pisać Home Assistant jako aplikację w Pythonie we wrześniu 2013 roku, a w listopadzie tego roku opublikował ją na GitHubie [1]. Projekt ma licencję Apache 2.0, więc każdy może go pobrać, uruchomić i zmieniać bez opłat [1]. W 2018 roku założyciele projektu powołali firmę Nabu Casa, która utrzymuje rozwój z subskrypcji opcjonalnej usługi Home Assistant Cloud i ze sprzedaży sprzętu [1][2]. W kwietniu 2024 roku własność kodu przeszła do organizacji non profit Open Home Foundation [1].

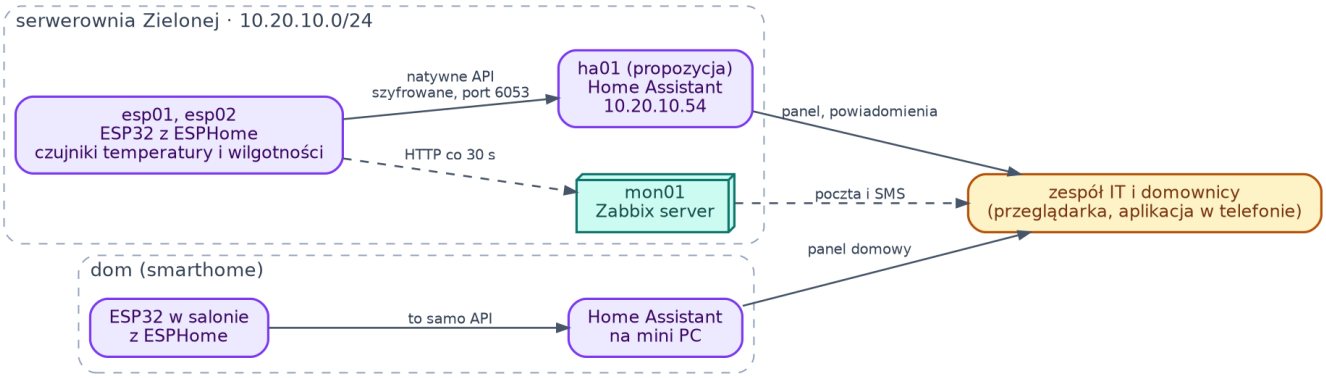
> **Pojęcie: Open Home Foundation.** Organizacja non profit, której powierzono Home Assistant, żeby jego kod pozostał wolny. Jako cele podaje prywatność, wybór i trwałość, a pod jej opieką jest ponad 250 projektów, między innymi ESPHome [3].

Nowe wydanie Home Assistant ukazuje się co miesiąc, zwykle w pierwszą środę: 2026.7 pojawiło się 1 lipca, 2026.8 5 sierpnia, a 2026.9 2 września, po drodze wychodzą poprawki (2026.9.4 z 27 września) [4].

> **Pojęcie: ESPHome.** Narzędzie, które z jednego pliku YAML buduje program dla płytki ESP32 i podobnych układów, a odczyty udostępnia przez sieć. Dziś rozwija je Open Home Foundation na licencji MIT [5][6]. Pierwotnym autorem jest Otto Winter, do dziś wśród opiekunów pakietu na PyPI [5]. Roku powstania i daty przejęcia przez Nabu Casa nie potwierdziliśmy, więc ich nie podajemy.

Jak to działa w jednym obrazku

Płytka ESP32 z ESPHome odczytuje czujnik i trzyma stałe, szyfrowane połączenie z Home Assistant, które nazywamy natywnym API [7]. Home Assistant zapamiętuje odczyty, rysuje panele, wykonuje automatyzacje i wysyła powiadomienia. Zabbix na `mon01` pyta te same płytki osobno, bo alarmy serwerowni zostają w jego rękach.



Rysunek 1. Home Assistant w Zielonej i w domu: płytki ESP32 rozmawiają z Home Assistant przez natywne API, a Zabbix na mon01 odpytuje te same płytki niezależnie. Warto zauważyć, że ten sam układ stoi w domu, tylko w osobnej instalacji. (kliknięcie otwiera pełny rozmiar)

> **Pojęcie: praca lokalna i chmura producenta.** Urządzenie lokalne rozmawia z serwerem w naszej sieci, a urządzenie w chmurze producenta z serwerami producenta, przez jego aplikację i jego konto.

Pytanie	Chmura producenta	Lokalnie (Home Assistant)
Co się dzieje, gdy zniknie internet	zwykle przestaje działać sterowanie i podgląd	automatyzacje i panele w sieci lokalnej działają dalej
Gdzie leżą dane i historia	u producenta	na naszym serwerze, w naszej kopii zapasowej
Co, gdy producent zmieni zasady lub zakończy usługę	zależymy od jego decyzji	kod jest wolny, urządzenia zostają pod naszą kontrolą
Urządzenia różnych marek w jednym panelu	zwykle osobna aplikacja na markę	jedna aplikacja i jedne reguły
Koszt oprogramowania	zwykle abonament albo cena w urządzeniu	licencja bez opłat, płatna tylko opcjonalna subskrypcja [2]

Chmury nie wykluczamy całkiem: kto chce, kupuje Home Assistant Cloud dla zdalnego dostępu [2]. Decyzja należy jednak do nas.

Dlaczego Zielona

> **Z ŻYCIA WZIĘTE**
>
> Wtorek, 8:50. Ola przynosi do pracy płytkę ESP32 z domu: w salonie mierzy nią temperaturę i wilgotność, a wynik ogląda w Home Assistant na telefonie. Tomek, w pracy od dwóch tygodni,

> rozpoznaje tę samą płytkę, na której zespół buduje czujniki serwerowni, i pyta, czy program też jest ten sam. Ola odpowiada, że ten sam ESPHome, tylko inny plik YAML. Pan Zbigniew pyta o koszt i słyszy, że oprogramowanie nic nie kosztuje, a płatna jest wyłącznie opcjonalna subskrypcja.

Zielona ma już Zabbixa i czujniki opisane na stronie Serwerownia Zielonej: gdzie stoją czujniki i dokąd płyną alarmy: dwa moduły ESP32, `esp01` i `esp02`, oraz moduł wejść `wej01`. Home Assistant dokłada panele dla ludzi, automatyzacje reagujące w sekundach (na przykład lampą przy drzwiach) i powiadomienia w aplikacji. Alarm zostaje u Zabbixa, a strona Automatyzacje i powiadomienia pokazuje, jak nie dublować alarmów.

Autor chce, żeby ten sam Home Assistant i te same płytki ESP32 służyły w firmie i w domu. Dlatego każda strona ma wątek domowy: ten sam plik ESPHome z innymi czujnikami, ten sam układ panelu z innymi kartami. Firmę i dom prowadzimy jednak w osobnych instalacjach, bo mają inne konta i inne dane.

Dla Zielonej proponujemy Home Assistant OS na maszynie wirtualnej `ha01` z adresem 10.20.10.54 (nazwa i adres to propozycja do potwierdzenia). Drugą drogą jest kontener na `docker01`, a różnice porównuje następna strona.

Mapa tej książki

Książka o Home Assistant ma siedem stron.

Strona	Na jakie pytanie odpowiada
Home Assistant: lokalny system dla serwerowni i domu	Skąd wziął się Home Assistant i po co Zielonej lokalny system
Instalacja Home Assistant: system na maszynie wirtualnej, mini PC albo w kontenerze	Którą drogę instalacji wybieramy i jak wygląda pierwsze uruchomienie
ESPHome i urządzenia: adopcja esp01 i esp02, API albo MQTT, nazwy encji	Jak dodać płytki do Home Assistant i jak nazywamy encje
Automatyzacje i powiadomienia: alarm temperatury w serwerowni bez dublowania Zabbixa	Jak napisać automatyzację i kto alarmuje, żeby nie było dubli
Panele w Home Assistant: karty, panel serwerowni, panel domowy i uprawnienia	Jak zbudować panel i komu go pokazać
Kopia, aktualizacja i dostęp zdalny: Home Assistant bez otwartego portu 8123	Jak robić kopie, aktualizować i wchodzić z zewnątrz bez otwierania portu

Strona	Na jakie pytanie odpowiada
Home Assistant: częste problemy z urządzeniami, bazą i aktualizacjami	Co robić, gdy urządzenie znika, baza rośnie albo coś nie kompiluje

Powiązane: Termometry w praktyce: wybór, montaż i własny czujnik (DIY) · Pomiar energii na ESP32 (DIY) · Zabbix: od pracy dyplomowej na Łotwie do monitoringu całej serwerowni

Źródła:

1. Wikipedia, „Home Assistant" (twórca, daty 2013, licencja, Open Home Foundation): en.wikipedia.org/wiki/Home_Assistant
2. Nabu Casa, „About" i Home Assistant Cloud: nabucasa.com/about oraz home-assistant.io/cloud
3. Open Home Foundation, strona główna (misja, projekty): openhomefoundation.org
4. Home Assistant, notatki wydań: home-assistant.io/blog/categories/release-notes
5. PyPI, pakiet esphome (licencja MIT, opiekunowie, Open Home Foundation): pypi.org/project/esphome
6. ESPHome, repozytorium projektu: github.com/esphome/esphome
7. Home Assistant, integracja ESPHome (stałe połączenie przez natywne API): home-assistant.io/integrations/esphome

Instalacja Home Assistant: system na maszynie wirtualnej, mini PC albo w kontenerze

Poziom: podstawy

Dotyczy: Home Assistant OS 18.3, Home Assistant 2026.9, Hyper-V na hv01 i hv02, Docker na docker01

Czas: ~60 min

Home Assistant instalujemy na trzy sposoby: jako system Home Assistant OS na maszynie wirtualnej albo na mini PC, albo jako kontener w Dockerze. Strona porównuje je, pokazuje komendy dla każdego i mówi, co wybieramy w Zielonej. Wracamy tu przy nowej instalacji, przy przenosinach na inny sprzęt i przy pytaniu, dlaczego w kontenerze nie ma aplikacji.

Skróty użyte na tej stronie (23)

Skrót	Co znaczy
BACnet	Building Automation and Control Networks: otwarty protokół automatyki budynkowej, którym instalacje techniczne (klimatyzacja, zasilanie, pożar) wysyłają stany do wspólnego panelu
BMS	Building Management System: system zarządzania budynkiem, który zbiera odczyty z czujników (temperatura, wilgotność) i steruje klimatyzacją oraz wentylacją
CLI	Command Line Interface: praca przez wpisywanie komend w terminalu
CSV	w Windowsie: Cluster Shared Volume, dysk widoczny naraz dla wszystkich węzłów klastra; w danych: Comma-Separated Values, plik z wartościami oddzielonymi przecinkami
DCIM	Data Center Infrastructure Management: dokumentacja tego, co fizycznie stoi w serwerowni, czyli szafy, urządzenia i okablowanie
DHCP	Dynamic Host Configuration Protocol: usługa, która sama rozdaje komputerom adresy IP
DNS	Domain Name System: książka telefoniczna internetu, zamienia nazwy na adresy IP

Skrót	Co znaczy
ESP32	Tani mikrokontroler z Wi-Fi i Bluetooth firmy Espressif: malutki komputer, z którego budujemy własne czujniki sieciowe
ESPHome	Oprogramowanie, które zamienia ESP32 w gotowy czujnik sieciowy bez pisania kodu od zera, a odczyty udostępnia przez sieć
GB	Gigabyte: miliard bajtów, jednostka pojemności pamięci i plików, tu najczęściej rozmiar modelu albo pamięci karty graficznej
HTTP	Hypertext Transfer Protocol: protokół stron WWW
IPv4	Internet Protocol wersja 4: adresy zapisane czterema liczbami z kropkami, na przykład 10.20.10.77
IT	Information Technology: dział albo praca związana z komputerami i infrastrukturą firmy
mDNS	Multicast DNS: sposób, w którym urządzenia w jednej sieci lokalnej ogłaszają swoją nazwę (np. esp01.local) zapytaniami na adres multicast, bez serwera DNS; nie przechodzi między podsieciami
Modbus	Prosty protokół komunikacji przemysłowej, którym liczniki energii i moduły pomiarowe wysyłają odczyty
Mosquitto	popularny serwer pośredniczący (broker) MQTT, który przekazuje wiadomości między urządzeniami a Home Assistant
OS	Operating System: system operacyjny
SSD	Solid State Drive: dysk półprzewodnikowy
UEFI	Unified Extensible Firmware Interface: nowoczesne oprogramowanie firmowe płyty głównej albo maszyny wirtualnej, następca starszego BIOS-u, obsługuje między innymi Secure Boot
USB	Universal Serial Bus: standard portów i urządzeń podłączanych do komputera, tu pendrive'y i inne nośniki wymienne
VHDX	Virtual Hard Disk v2: format pliku dysku wirtualnego maszyny Hyper-V, następca starszego VHD, obsługuje większe rozmiary i jest odporniejszy na uszkodzenie przy nagłym zaniku zasilania
YAML	YAML Ain't Markup Language: format plików konfiguracyjnych oparty na wcięciach
Zigbee	Radiowy protokół sieci domowych o małym poborze prądu, używany w czujkach, czujnikach i przełącznikach na baterie

Fiszka

Komendy Dockera wykonujemy w CLI bash na `docker01` jako user z uprawnieniami sudo, a komendy PowerShell na `hv01` w podniesionym PowerShellu. Interfejs Home Assistant mamy po polsku, ale ścieżki podajemy po angielsku, jak w dokumentacji, bo polskie nazwy pozycji bywają inne w kolejnych wydaniach (Settings to Ustawienia).

ZMIENIA SYSTEM

Założenie katalogów aplikacji: mkdir -p

```
sudo mkdir -p /docker/apps/homeassistant /docker/compose/homeassistant
```

Pierwszy katalog trzyma dane Home Assistant (w kontenerze to katalog `/config`), drugi plik Compose.

mkdir = make directory: utwórz katalog

PLIK KONFIGURACYJNY

Plik Compose Home Assistant: /docker/compose/homeassistant/compose.yaml

```
services:
  homeassistant:
    container_name: homeassistant
    image: ghcr.io/home-assistant/home-assistant:2026.9.4
    restart: unless-stopped
    stop_grace_period: 60s
    network_mode: host
    environment:
      TZ: Europe/Warsaw
    volumes:
      - /docker/apps/homeassistant:/config
      - /etc/localtime:/etc/localtime:ro
```

Przykład z dokumentacji ma jeszcze `privileged: true` i dostęp do `/run/dbus`, potrzebne dla urządzeń USB i Bluetooth [1]. Na `docker01` nie podłączamy takich urządzeń, więc tych dwóch rzeczy nie dajemy, a po starcie sprawdzamy dziennik. Wersję obrazu przypinamy numerem, nigdy `stable` ani `latest`: numer 2026.9.4 sprawdziliśmy 3 października 2026 w dokumentacji i przy każdej instalacji sprawdzamy go na nowo [1].

> W naszym środowisku dane aplikacji trzymamy w `/docker/apps/<nazwa>`, a plik Compose w `/docker/compose/<nazwa>/compose.yaml`. To konwencja, nie wymóg Dockera: można ją zmienić, byle w całej firmie była jedna. Katalog `/docker` to osobny wolumen `vgDocker/lvDocker`, żeby pełny dysk aplikacji nie zatrzymał systemu.

USB = Universal Serial Bus: standard portów i urządzeń podłączanych do komputera, tu pendrive'y i inne nośniki wymienne

ZMIENIA SYSTEM

Otwarcie portu 8123 dla sieci serwerów: firewall-cmd

```
sudo firewall-cmd --add-port=8123/tcp --permanent
sudo firewall-cmd --reload
```

Kontener w trybie hosta słucha bezpośrednio na portach `docker01`, więc zapora hosta musi go przepuścić. Port otwieramy tylko na hoście: `fw01` nie wystawia go do internetu (strona Kopia zapasowa, aktualizacje i bezpieczny dostęp zdalny).

firewall-cmd = sterowanie zaporą firewalld

TYLKO ODCZYT

Uruchomienie kontenera: docker compose up -d

```
docker compose -f /docker/compose/homeassistant/compose.yaml up -d
```

docker = narzędzie do kontenerów

TYLKO ODCZYT

Sprawdzenie dziennika kontenera: docker compose logs

```
docker compose -f /docker/compose/homeassistant/compose.yaml logs --tail 50
homeassistant
```

Po pierwszym starcie szukamy w dzienniku komunikatów o błędach integracji. Pierwsze uruchomienie trwa kilka minut, bo Home Assistant dociąga biblioteki.

docker = narzędzie do kontenerów

TYLKO ODCZYT

Sprawdzenie konfiguracji w kontenerze: check_config

```
docker exec homeassistant python -m homeassistant --script check_config --config /config
```

Polecenie z dokumentacji sprawdza pliki YAML, zanim restartujemy kontener [2].

YAML = YAML Ain't Markup Language: format plików konfiguracyjnych oparty na wcięciach · **docker** = narzędzie do kontenerów

TYLKO ODCZYT

Aktualizacja kontenera po zmianie numeru wersji: docker compose pull

```
docker compose -f /docker/compose/homeassistant/compose.yaml pull
docker compose -f /docker/compose/homeassistant/compose.yaml up -d
```

Najpierw zmieniamy numer w `compose.yaml` i robimy kopię zapasową katalogu `/docker/apps/homeassistant` (strona o kopii zapasowej). Dokumentacja przypomina też o przejrzeniu zgłoszeń Repairs i dziennika po aktualizacji [2].

docker = narzędzie do kontenerów

ZMIENIA SYSTEM

Utworzenie maszyny wirtualnej w Hyper-V: New-VM

```
New-VM -Name "ha01" -Generation 2 -MemoryStartupBytes 4GB -VHDPath
"C:\ClusterStorage\Volume1\ha01\NazwaPlikuHaos.vhdx" -SwitchName "NazwaPrzelacznika"
Set-VMProcessor -VMName "ha01" -Count 2
Set-VMFirmware -VMName "ha01" -EnableSecureBoot Off
Start-VM -Name "ha01"
```

(gdzie `NazwaPlikuHaos.vhdx` to rozpakowany obraz Hyper-V z wydania Home Assistant OS, a `NazwaPrzelacznika` to przełącznik wirtualny w trybie zewnętrznym, podłączony do sieci serwerów)

Komendy można wykonywać całościowo jako blok tekstu. Dokumentacja wymaga maszyny drugiej generacji, co najmniej 2 GB pamięci i 2 procesorów wirtualnych, przełącznika w trybie mostu i wyłączonego Secure Boot [3]. Pamięć 4 GB i dysk z obrazu to nasz zapas, nie wymóg.

OS = Operating System: system operacyjny · **GB** = Gigabyte: miliard bajtów, jednostka pojemności pamięci i plików, tu najczęściej rozmiar modelu albo pamięci karty graficznej

□□ ZMIENIA SYSTEM

Dodanie maszyny do klastra: Add-ClusterVirtualMachineRole

```
Add-ClusterVirtualMachineRole -VirtualMachine "ha01"
```

Dzięki temu `ha01` jest rolą klastra i przy awarii `hv01` uruchomi się na `hv02` [4]. Plik VHDX musi leżeć na wolumenie klastra (CSV), jak pozostałe maszyny.

□□ **VHDX** = Virtual Hard Disk v2: format pliku dysku wirtualnego maszyny Hyper-V, następca starszego VHD, obsługuje większe rozmiary i jest odporniejszy na uszkodzenie przy nagłym zaniku zasilania · **CSV** = w Windowsie: Cluster Shared Volume, dysk widoczny naraz dla wszystkich węzłów klastra; w danych: Comma-Separated Values, plik z wartościami oddzielonymi przecinkami

□□ TYLKO ODCZYT

Zapis obrazu na mini PC: Balena Etcher

```
1. Pobieramy z wydań Home Assistant plik haos_generic-x86-64-NumerWersji.img.xz
2. Zapisujemy go na nośniku programem Balena Etcher albo narzędziem Disks z Ubuntu
3. W BIOS włączamy tryb UEFI i wyłączamy Secure Boot
4. Uruchamiamy mini PC z kablem Ethernet i czekamy kilka minut
```

Dokumentacja podaje dwie drogi: rozruch z Ubuntu na pendrivie i odtworzenie obrazu na dysk wewnętrzny narzędziem Disks (droga zalecana) albo zapis na nośnik rozruchowy programem Balena Etcher [5]. Nazwa pliku w wydaniu 18.3 to `haos_generic-x86-64-18.3.img.xz` [5].

□□ KLIKAMY W INTERFEJSIE

Ustawienie adresu i nazwy hosta w Home Assistant OS: Settings

```
Settings → System → Network
```

Tam odczytujemy nazwę hosta i ustawiamy adres IPv4 jako statyczny: dla `ha01` 10.20.10.54, brama 10.20.10.1, DNS 10.20.10.11 [6]. Alternatywą jest rezerwacja adresu w DHCP na `dc01` . Dokładny układ okna zależy od wersji, więc go sprawdzamy po instalacji.

□□ ZRZUT EKRANU DO UZUPEŁNIENIA

Zrzut do uzupełnienia: ekran Settings → System → Network w `ha01` z ustawionym statycznym adresem 10.20.10.54, bramą 10.20.10.1 i DNS 10.20.10.11.

OS = Operating System: system operacyjny · **IPv4** = Internet Protocol wersja 4: adresy zapisane czterema liczbami z kropkami, na przykład 10.20.10.77 · **DNS** = Domain Name System: książka telefoniczna internetu, zamienia nazwy na adresy IP · **DHCP** = Dynamic Host Configuration Protocol: usługa, która sama rozdaje komputerom adresy IP

TYLKO ODCZYT

Pierwsze logowanie i konto właściciela: onboarding

```
Maszyna wirtualna ha01 (nowa instalacja HAOS od 2026.8): http://10.20.10.54
Kontener na docker01:                                    http://10.20.10.42:8123
Stary adres w dokumentacji (zależy od mDNS):             http://homeassistant.local:8123
```

Na ekranie powitalnym zakładamy konto właściciela, podajemy nazwę domu i położenie. Pierwszy utworzony użytkownik zostaje właścicielem (owner) i zarządza pozostałymi [7]. Konta administratorów IT zakładamy od razu osobno, a właściciela zostawiamy jako konto awaryjne.

ZRZUT EKRANU DO UZUPEŁNIENIA

Zrzut do uzupełnienia: ekran powitalny Home Assistant z formularzem konta właściciela (imię, nazwa użytkownika, hasło) na `ha01`.

IT = Information Technology: dział albo praca związana z komputerami i infrastrukturą firmy

TYLKO ODCZYT

Sprawdzenie konfiguracji w Home Assistant OS: ha core check

```
ha core check
```

Polecenie wykonujemy w konsoli systemu albo w aplikacji terminala. Ten sam test daje w interfejsie ścieżka Settings → Tools → YAML → Check configuration [8].

OS = Operating System: system operacyjny · **YAML** = YAML Ain't Markup Language: format plików konfiguracyjnych oparty na wcięciach

Z życia wzięte

Z ŻYCIA WZIĘTE

Czwartek, 14:00. Tomek stawia Home Assistant jako kontener na `docker01` i chce dołożyć broker Mosquitto, ale w interfejsie nie znajduje sklepu z aplikacjami. Ola tłumaczy, że kontener nie ma Supervisora, więc nie ma też aplikacji: Mosquitto i ESPHome trzeba uruchomić jako osobne kontenery. Tomek notuje, że przy Home Assistant OS ten sam broker instaluje się dwoma kliknięciami, i do końca dnia ustalają z Olą, który wariant naprawdę pasuje do Zielonej.

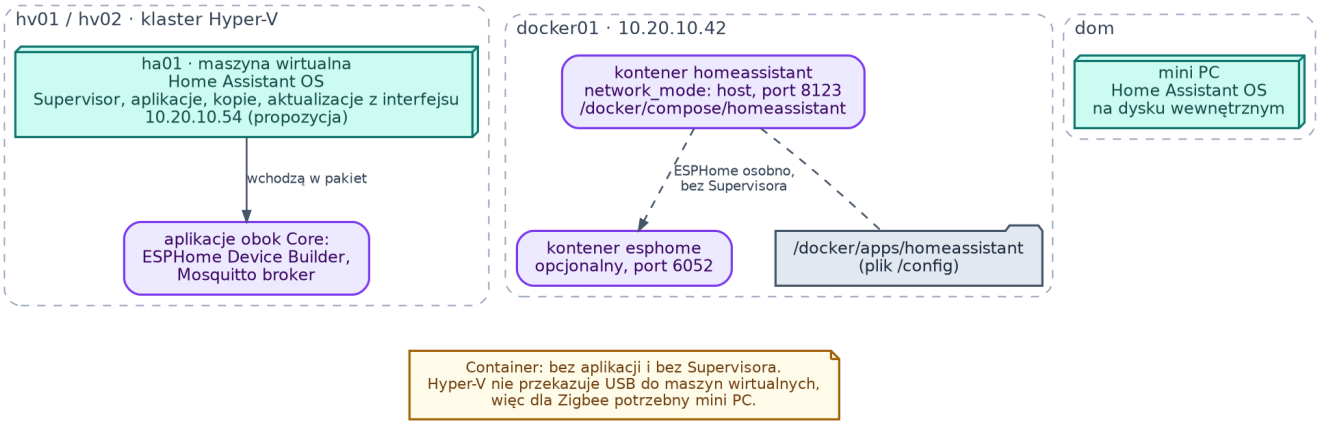
Jak to działa

Rozwiń: jak to działa pod spodem

Pojęcie: Home Assistant OS. Minimalny system operacyjny zbudowany wyłącznie po to, żeby uruchamiać Home Assistant. Ma jednokliknięciowe aktualizacje i kopie zapasowe, a dokumentacja nazywa go zalecanym rodzajem instalacji dla większości użytkowników [9].

Pojęcie: Supervisor i aplikacje. Supervisor to usługa wbudowana w Home Assistant OS, która zarządza systemem i uruchamia aplikacje, czyli samodzielne programy działające obok Home Assistant, takie jak broker Mosquitto albo ESPHome Device Builder. Do wydania 2026.2 aplikacje nazywały się dodatkami (add-ons) i stąd ta nazwa wciąż pojawia się w starszych poradnikach [10].

Pojęcie: Container. Sam program Home Assistant w kontenerze Dockera, bez Supervisora, bez aplikacji i bez zarządzania systemem. Hosta, Dockera, aktualizacje obrazu i kopie katalogu danych prowadzimy sami [2].



Rysunek 1. Trzy drogi instalacji: Home Assistant OS na maszynie wirtualnej ha01 w klastrze Hyper-V, kontener na docker01 z danymi w /docker/apps/homeassistant i mini PC w domu. Warto zauważyć, że aplikacje (ESPHome, Mosquitto) wchodzą w pakiet tylko w Home Assistant OS, a w kontenerze trzeba je uruchomić osobno. (kliknięcie otwiera pełny rozmiar)

Cecha	Home Assistant OS (VM `ha01` albo mini PC)	Container na `docker01`
Aplikacje (ESPHome, Mosquitto)	tak, instalowane z sklepu w interfejsie	nie, osobne kontenery
Supervisor	tak	nie
Aktualizacje	z interfejsu, z kopią przed aktualizacją	zmiana numeru obrazu i `compose up -d`
Kopie zapasowe	wbudowane, z aplikacjami i szyfrowaniem	wbudowane działają też w kontenerze, ale aplikacji nie ma; plus kopia katalogu `/config`
System i Docker	obraz zarządzany przez Home Assistant (dwa sloty rozruchowe, wycofanie po nieudanym starcie)	nasza praca: łatki Rocky Linux i Dockera
Urządzenia USB (Zigbee)	mini PC tak, Hyper-V nie [3]	host `docker01` musi mieć fizyczny dostęp do USB
Zasoby	cała maszyna wirtualna dla jednej aplikacji	współdzieli `docker01` z Gitea, MailCow i resztą
Kiedy wybieramy	gdy chcemy zalecaną, kompletną instalację	gdy Docker już jest, a aplikacje dołożymy sami

Do porównania warto dodać jeszcze dwa wyjaśnienia. Dokumentacja Home Assistant wymienia dwie drogi (Home Assistant OS i Container) i nie opisuje już rodzajów Supervised i Core, dlatego ich nie omawiamy [9]. W maszynie wirtualnej Hyper-V nie ma przekazywania USB, więc Zigbee wymaga mini PC albo koncentratora na Ethernecie [3].

Sieć hosta

W kontenerze ustawiamy `network_mode: host`. Home Assistant odkrywa urządzenia domowe przez rozgłaszanie w sieci lokalnej (mDNS), a taki ruch nie przechodzi przez domyślną sieć mostkową Dockera. Kontener w trybie hosta słucha bezpośrednio na adresie `docker01`, co oznacza, że nie mapujemy portów, a zaporę ustawiamy na hoście.

W maszynie wirtualnej wybieramy przełącznik wirtualny w trybie zewnętrznym, żeby `ha01` miał własny adres w sieci serwerów 10.20.10.0/24 i widział te same rozgłoszenia co płytki ESP32. Port interfejsu zależy od rodzaju instalacji: od wydania 2026.8 nowa instalacja Home Assistant OS słucha na porcie 80, a Container zostaje przy 8123 [11]. Istniejące instalacje nie zmieniają portu, co potwierdzają notatki wydania 2026.8 [12]. Po instalacji sprawdzamy, na którym porcie nasłuchuje `ha01`, i ten port wpisujemy w nginx i w Zabbixa.

Pierwsze uruchomienie

Po pierwszym starcie Home Assistant prowadzi przez kreator: konto właściciela, nazwa domu, położenie, strefa czasowa, jednostki i zgoda na statystyki (opcjonalna). Dalej w interfejsie dodajemy integracje: ESPHome dla płytek (następna strona), a w Zielonej także Ping dla `mon01`. Przywrócenie kopii zapasowej jest także możliwe na ekranie powitalnym, co przydaje się przy przeprowadzce na nowy sprzęt [13].

Co wybieramy w Zielonej

Dla `ha01` wybieramy maszynę wirtualną z Home Assistant OS. Powody są trzy: kompletna instalacja z aplikacjami (ESPHome Device Builder i Mosquitto) bez prowadzenia ich osobno, wbudowane kopie zapasowe i aktualizacje, oraz oddzielenie od `docker01`, na którym stoją Gitea, MailCow i inne usługi, których awaria nie powinna zatrzymywać panelu. Cena to jedna maszyna wirtualna więcej na klastrze i brak USB, a ten drugi minus nie dotyczy nas, bo płytki ESP32 łączą się Ethernetem.

Kontener na `docker01` zostaje drugą drogą, kiedy cały Home Assistant ma być prostym panelem bez aplikacji. Adres wtedy to `docker01` 10.20.10.42, port 8123. Adres 10.20.10.54 sprawdziliśmy w tabeli serwerów Zielonej: kolejne po `gpu01` (.45) jest wolne i nie używa go żadna strona bazy. Nazwa `ha01` i ten adres są propozycją do potwierdzenia przez autora.

W domu

W domu zwykle wybieramy Home Assistant OS na mini PC albo na gotowym urządzeniu: dokumentacja wymienia Home Assistant Green jako sprzęt z preinstalowanym systemem oraz Raspberry Pi i maszyny x86-64 [9]. Mini PC musi obsługiwać UEFI i dyski z sektorem logicznym 512 bajtów [5]. Dla Zigbee i Z-Wave wybieramy mini PC albo Green z kluczem USB, bo przez Hyper-V takiego klucza nie przekażemy [3]. Firmę i dom trzymamy w osobnych instalacjach, każda z własnymi kontami i własną kopią zapasową.

> □□ Więcej możliwości albo gotowe rozwiązanie od ręki (out of the box): gotowy kontroler domowy z preinstalowanym systemem (jak Home Assistant Green) albo mini PC z dyskiem SSD i zasilaczem awaryjnym, a w serwerowni maszyna wirtualna w klastrze, która sama przenosi się na drugi węzeł. W dużych centrach danych stosuje się systemy BMS i DCIM z klastrami wysokiej dostępności i redundantnymi serwerami zarządzania, a automatykę budynku prowadzą wyspecjalizowane sterowniki z protokołami przemysłowymi (BACnet, Modbus).

Częste błędy

Objaw	Przyczyna	Co robimy
Maszyna wirtualna nie startuje, ekran zatrzymuje się na rozruchu	włączony Secure Boot	`Set-VMFirmware -VMName "ha01" -EnableSecureBoot Off` [3]
`homeassistant.local` się nie otwiera	mDNS nie działa na urządzeniu albo w sieci, a nowa instalacja słucha na porcie 80, nie 8123	wchodzimy po adresie IP i sprawdzamy port; po zmianie portu dla zakładek patrz strona o częstych problemach [11][12]
Kontener działa, ale nie widzi urządzeń w sieci	brak `network_mode: host`	dopisujemy tryb hosta i uruchamiamy `compose up -d`
Port 8123 nie odpowiada z innego komputera	zapora `firewalld` na `docker01`	`firewall-cmd --add-port=8123/tcp --permanent` i `--reload`
W interfejsie brak sklepu z aplikacjami	Container nie ma Supervisora i aplikacji	uruchamiamy Mosquitto i ESPHome jako osobne kontenery albo przechodzimy na Home Assistant OS [2]
Wykresy i dzienniki mają złą godzinę	brak strefy czasowej	`TZ: Europe/Warsaw` w Compose, a w Home Assistant OS strefa w Settings → System → General
Brak klucza USB w Home Assistant na Hyper-V	Hyper-V nie obsługuje USB	mini PC albo koncentrator sieciowy [3]

□□ Ćwiczenia

1. Dlaczego w kontenerze Home Assistant nie znajdziemy aplikacji, a w Home Assistant OS tak?

Odpowiedź

Aplikacje uruchamia Supervisor, a kontener jest samym programem Home Assistant bez Supervisora. Dokumentacja mówi wprost, że ten rodzaj instalacji nie ma dostępu do aplikacji, a jeśli ich potrzebujemy, wybieramy inny rodzaj instalacji [2].

1. Maszyna `ha01` nie startuje po utworzeniu w Hyper-V. Co sprawdzamy najpierw?

Odpowiedź

Generację maszyny (musi być druga) i Secure Boot, który dokumentacja każe wyłączyć w ustawieniach zabezpieczeń maszyny. Poleceniem jest `Set-VMFirmware -VMName "ha01" -EnableSecureBoot Off` [3].

1. Dlaczego w pliku Compose nie wpisujemy `image: ...:stable`?

Odpowiedź

Bo `stable` zmienia się po każdym wydaniu i `docker compose pull` pobrałby nową wersję bez naszej decyzji. Numer, na przykład `2026.9.4`, pozwala aktualizować świadomie, po przeczytaniu notatek wydania, i wycofać zmianę.

1. Gdzie w Zielonej wskazujemy adres i port `ha01`, gdy nowa instalacja Home Assistant OS słucha na porcie 80?

Odpowiedź

W konfiguracji nginx (cel `proxy_pass`), w elemencie Zabbixa sprawdzającym port i w zakładkach zespołu. Port ustalamy po instalacji, a nie z pamięci, bo zależy od wydania Home Assistant OS [11][12].

Powiązane: Jak trzymamy aplikacje w Dockerze: konwencje Zielonej · Hyper-V: maszyny wirtualne · Serwerownia Zielonej: gdzie stoją czujniki i dokąd płyną alarmy

Źródła:

1. Home Assistant, instalacja Container na Linuksie (plik Compose, `privileged`, wymagania): home-assistant.io/installation/linux
2. Home Assistant, polecenia dla Container (check_config, aktualizacja, tagi): home-assistant.io/common-tasks/container
3. Home Assistant, instalacja w Hyper-V (generacja 2, Secure Boot, brak USB): home-assistant.io/installation/windows
4. Microsoft, Add-ClusterVirtualMachineRole: learn.microsoft.com
5. Home Assistant, instalacja na x86-64: home-assistant.io/installation/generic-x86-64
6. Home Assistant, zadania w systemie (Settings, System, Network): home-assistant.io/common-tasks/os
7. Home Assistant, dostawcy uwierzytelniania (pierwszy użytkownik jako właściciel): home-assistant.io/docs/authentication/providers
8. Home Assistant, sprawdzenie konfiguracji w Home Assistant OS: home-assistant.io/common-tasks/os
9. Home Assistant, rodzaje instalacji: home-assistant.io/installation
10. Home Assistant 2026.2, „add-ons are now called apps": home-assistant.io/blog/2026/02/04/release-20262
11. Home Assistant, integracja HTTP (port 80 dla Home Assistant OS od 2026.8): home-assistant.io/integrations/http
12. Home Assistant 2026.8, notatki wydania (nowe instalacje na porcie 80): home-assistant.io/blog/2026/08/05/release-20268
13. Home Assistant, zadania ogólne (kopie zapasowe, przywracanie przy onboardingu): home-assistant.io/common-tasks/general