Windows Sandbox

Przygotowanie skryptów pod konkretne wymagania, wiążą się z potrzebą dokładnego sprawdzenia ich działania. W większości przypadków skrypty te wykonują pewne operacje na systemach Windows. Ich testowanie na stacji wiązałoby się z potrzebą nie jednej reinstalacji systemu. Rozwiązanie, które przechodzi pierwsze na myśl to oczywiście Hyper-V i wirtualna maszyna, ale czy jedyne?
Poniżej, pokaże Ci funkcjonalność Windows Sandbox w "dziesiątce", która może być świetną alternatywą dla tego typu potrzeb (ale nie tylko).
Czym jest Windows Sandbox?
Windows Sandbox to rozwiązanie umożliwiające w szybki sposób uruchomienie zwirtualizowanego środowiska na swojej stacji, które co prawda użyje lokalnego systemu operacyjnego, ale w izolowany sposób. Dzięki temu podejrzane pliki, szkodliwe aplikacje, wadliwe skrypty nie będą miały wpływu na system.
Po zamknięciu piaskownicy całe oprogramowanie wraz ze wszystkimi plikami i stanem zostanie trwale usunięte. Jest to szybszy i prostszy sposób niż konfiguracja Hyper-V i wirtualizacja systemu.
Dodatkowo co jest zaletą, piaskownica zajmie tylko około 100MB przestrzeni dyskowej.

A. Instalacja Windows Sandbox

Przed instalacja spójrz na wymagania wstępne:
Co najmniej Windows 10 Pro lub Enterprise Insider build 18342 lub nowszy
Architektura 64-bitowa
Funkcje wirtualizacji włączone w systemie BIOS
Co najmniej 4GB RAM (rekomendowane 8GB)
Co najmniej 1 GB wolnej przestrzenidyskowej (rekomendowany dysk SSD)
Co najmniej 2 rdzenie CPU (rekomendowane 4 rdzenie z hyperthreadingiem)
Jeżeli chcesz wykonać kolejne czynność w maszynie wirtualnej Hyper-V to musisz włączyć wirtualizację zagnieżdżoną (jeśli będziesz je wykonywał w szkole to jest ona włączona):

Jeśli sprawdziłeś i nie ma przeciwskazań to wykonaj poniższe polecenie PowerShell:

Powyższy skrypt zadziała, jeśli masz Windows 11 z numerem wersji Build większym niż 18305.
Komenda Enable-WindowsOptionalFeature włączy funkcję Windows Sandbox o nazwie Containers-DisposableClientVM. Po uruchomieniu tej komendy jest wymagane ponowne uruchomienie komputera.
Startujemy z Windows Sandbox

B. Personalizacja Windows Sandbox

Standardowo piaskownica za każdym razem uruchamia się czysta jako nowa instalacja systemu Windows. Jednak nie musi tak być, za pomocą plików konfiguracyjnych możesz wpływać na ustawienia w 4 zakresach:

1. vGPU (zwirtualizowany procesor graficzny)

Włącz lub wyłącz zwirtualizowany procesor GPU. Jeśli vGPU jest wyłączone, Sandbox użyje WARP (rasteryzatora oprogramowania).

2. Sieć

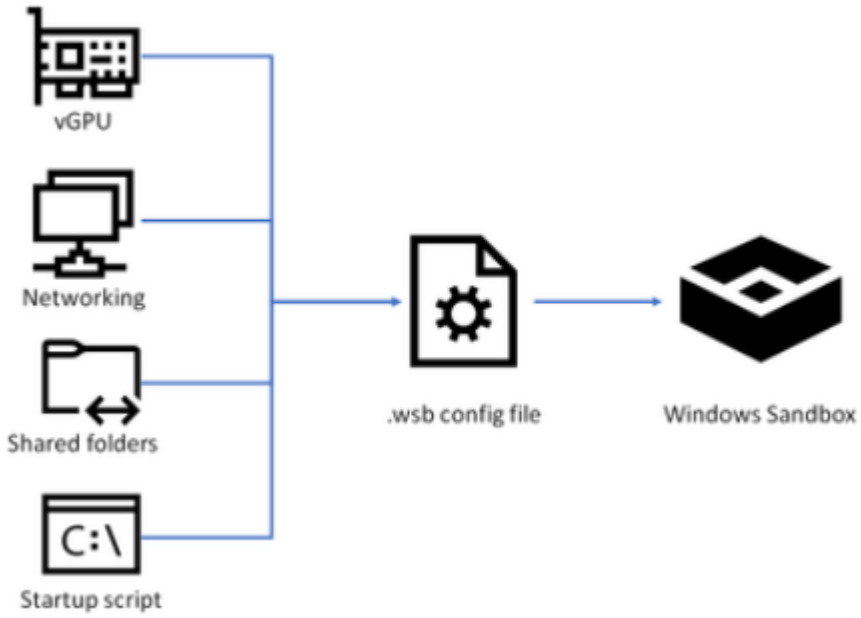
Włącz lub wyłącz dostęp sieciowy do piaskownicy.

3. Udostępnione foldery

Udostępniaj foldery z hosta z uprawnieniami do odczytu lub zapisu. Należy pamiętać, że ujawnienie katalogów hostów może pozwolić złośliwemu oprogramowaniu na wpłynięcie na system lub kradzież danych (widoczne na pulpicie – C:\Users\WDAGUtilityAccount\Desktop).

4. Skrypt startowy

Akcja logowania do piaskownicy.



Pliki konfiguracyjne Windows Sandbox są sformatowane jako XML i używają rozszerzenia pliku .wsb. Przetestuj przykładowe pliki konfiguracyjne.

Przykładowy plik konfiguracyjny - 0.

Poniżej przykład pliku konfiguracyjnego .wsb, który przygotowuję od razu piaskownicę do pracy z Visual Studio Code.

Dzięki takiej konfiguracji folderów mam dostęp (tylko odczyt) do niezbędnych zasobów ze skryptami oraz dodatkowy katalog Out(zapis), który jest moim miejscem pracy w VSCode.

Przetestuj przykładowy plik konfiguracyjny - 1.

Poniższego pliku konfiguracyjnego można użyć do łatwego testowania pobranych plików w piaskownicy. W tym celu skrypt wyłącza sieć i vGPU, a w kontenerze ogranicza udostępniony folder Pobrania do dostępu tylko do odczytu. Dla wygody polecenie logowania otwiera folder pobierania w kontenerze podczas uruchamiania.

Utwórz Downloads1.wsb

Przywróć pierwszy punk kontrolny wykonaj instalacje Windows Sandbox

Przetestuj przykładowy plik konfiguracyjny - 2.

Poniższy plik konfiguracyjny instaluje kod Visual Studio w kontenerze, co wymaga nieco bardziej skomplikowanej konfiguracji LogonCommand.

W C:\ są dwa foldery; pierwszy (SandboxScripts) zawiera VSCodeInstall.cmd, który instaluje i uruchamia VSCode. Zakłada się, że drugi folder (CodingProjects) zawiera pliki projektu, które deweloper chce zmodyfikować za pomocą VSCode.

Za pomocą skryptu instalatora VSCode, który jest już zmapowany do kontenera, LogonCommand może się do niego odwoływać.

Utwórz VSCodeInstall.cmd

Utwórz VSCode2.wsb

Teraz wystarczy dwukrotnie kliknąć VSCode2.wsb i uruchomić Windows Sandbox. Zaletą tego jest to, że można utworzyć kilka konfiguracji dla sposobu działania piaskownicy. Właściwie całkiem praktyczne. Można mieć tylko nadzieję, że Microsoft z czasem rozszerzy możliwości pliku konfiguracyjnego. Może być konieczne odsunięcie komunikatu piaskownicy i ręczne uruchomienie pliku cmd

Zgłoszenie 2

Przywróć pierwszy punk kontrolny wykonaj instalacje Windows Sandbox

Przetestuj przykładowy plik konfiguracyjny - 3

Utwórz skrypt PowerShell korzystając z poniższego kodu i zapisz go w katalogu C:\sandbox jako SwapMouse3.ps1.

Utwórz SwapMouse3.wsb

Wyjaśnienie:

SwapMouse.ps1: Ten skrypt PowerShell zmienia podstawowy przycisk myszy, aby dostosować go dla użytkowników leworęcznych. Wczytuje niezbędne biblioteki, definiuje funkcję SwapMouseButton z biblioteki user32.dll i wywołuje tę funkcję, aby zamienić przyciski myszy.
SwapMouse.wsb: Jest to plik konfiguracyjny Windows Sandbox, który mapuje katalog C:\sandbox na hoście do katalogu C:\sandbox w sandboxie jako tylko do odczytu. Po zalogowaniu się do sandboxa uruchamia skrypt SwapMouse.ps1 za pomocą PowerShell z pominięciem polityki wykonywania skryptów.
Użycie:
Stwórz katalog C:\sandbox na komputerze lokalnym i umieść tam skrypt SwapMouse.ps1.
Utwórz plik SwapMouse.wsb z powyższą zawartością i zapisz go w dowolnym miejscu.
Uruchom plik SwapMouse.wsb poprzez dwukrotne kliknięcie, co spowoduje uruchomienie Windows Sandbox z zmapowanym katalogiem i wykonanie skryptu przy logowaniu.
To rozwiązanie automatycznie skonfiguruje przyciski myszy w środowisku Windows Sandbox dla leworęcznych użytkowników.

Zgłoszenie 3

Przywróć pierwszy punk kontrolny wykonaj instalacje Windows Sandbox

C. Instalacja Visual Studio Code w Windows Sandbox

Napisz skrypt PowerShell, który instaluje Visual Studio Code, przenosi ustawienia oraz wybrane rozszerzenia z lokalnego hosta w Windows Sandbox, należy uwzględnić specyfikę środowiska Sandbox.
Skrypt PowerShell do instalacji Visual Studio Code w Windows Sandbox oraz przenoszenia ustawień i rozszerzeń z lokalnego hosta.
Ustawienia ścieżek i listy rozszerzeń
Zmiana polityki wykonywania skryptów
Instalacja Chocolatey
Instalacja Visual Studio Code
Przenoszenie ustawień
Przenoszenie rozszerzeń
Uruchomienie Visual Studio Code

Jak używać:

1. Utwórz plik .ps1:

Skopiuj powyższy skrypt do pliku .ps1, np. install_vscode.ps1.

2. Uruchom skrypt w Windows Sandbox:

Skrypt musi być uruchomiony w Windows Sandbox z odpowiednimi uprawnieniami administracyjnymi.

Wykonaj instalację PowerShell w VSC uruchamiającą powyższy skrypt:

Podaj efekt:

Zgłoszenie 4

Przywróć pierwszy punk kontrolny wykonaj instalacje Windows Sandbox

3. Instalacja w Windows Sandbox aplikacji Visual Studio Code z pobraniem z przeglądarki

Pobranie https://code.visualstudio.com/docs/?dv=win i zainstalowalnie

Dodanie rozszerzenia do PowerShell

Zgłoszenie 5

Przywróć pierwszy punk kontrolny.

Podaj wnioski z powyższych czynności.

Zgłoszenie 6