

Projekt: Hostowanie serwera gry na VPS

Parametry:

Host: hostinger.com (KVM 4 VPS)

OS: Debian 11 64bit

Maszyna: 4vCPU, 16GB RAM, 200GB NVMe

Gra: PalWorld v0.1.2.0-v0.1.3.0 EarlyAccess (na samym starcie pisania projektu v1.2, ale 25/01/2024 o godzinie 14:00 wypuścili nowego patcha v1.3 więc dodałem część o aktualizowaniu serwera)

Łączenie z terminalem oraz wstępna konfiguracja OS

1. Łączymy się przez SSH (w moim przypadku łączyłem się przez Windows cmd.exe) na nowo stworzony serwer VPS z systemem Debian 11 64bit

 OpenSSH SSH client

```
Microsoft Windows [Version 10.0.19045.3930]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\system32>ssh root@62.72.35.63
```

2. Po zalogowaniu aktualizujemy wszystkie istniejące pakiety w systemie do najnowszej wersji za pomocą poniższych komend (akceptujemy ewentualne komunikaty) :

apt upgrade && apt dist-upgrade

apt upgrade: Komenda ta służy do aktualizacji istniejących pakietów na systemie do ich najnowszych dostępnych wersji. Jest to proces polegający na pobraniu i zainstalowaniu nowych wersji oprogramowania, które już jest zainstalowane na systemie. Jest stosunkowo bezpieczna i zachowawcza, ponieważ nie usuwa ani nie instaluje nowych pakietów, a jedynie aktualizuje istniejące.

apt dist-upgrade: Ta komenda idzie krok dalej i aktualizuje system operacyjny wraz z jego pakietami. Jeśli nowa wersja systemu jest dostępna, to apt dist-upgrade będzie próbować zaktualizować cały system do najnowszej wersji. Może to obejmować usuwanie niepotrzebnych pakietów, instalowanie nowych lub zmiany w konfiguracji systemu w celu spełnienia nowych wymagań. Jest bardziej radykalna niż apt upgrade i może prowadzić do większych zmian w systemie.

3. Instalacja pakietu **software-properties-common**:

apt install software-properties-common

apt install: To polecenie systemu zarządzania pakietami APT (Advanced Package Tool), które służy do instalacji pakietów oprogramowania na systemie. Działanie tego polecenia polega na pobraniu, rozpakowaniu i zainstalowaniu określonego pakietu.

software-properties-common: To jest nazwa konkretnego pakietu, który chcemy zainstalować. Pakiet "software-properties-common" zawiera narzędzia i biblioteki umożliwiające dodawanie, usuwanie oraz zarządzanie repozytoriami oprogramowania na systemie. Repozytoria to miejsca, z których system pobiera oprogramowanie do instalacji i aktualizacji.

4. Dodawanie repozytorium "non-free":

apt-add-repository non-free

add-apt-repository lub **apt-add-repository:** To narzędzie lub polecenie, które umożliwia dodawanie nowych repozytoriów do listy źródeł pakietów systemu. Używa się go, aby ułatwić dostęp do oprogramowania dostępnego w zewnętrznych repozytoriach.

non-free: Jest to kategoria repozytorium oprogramowania w systemach Debian i Ubuntu. Oznacza ona oprogramowanie, które nie jest wolne i otwarte, co oznacza, że może zawierać ograniczenia lub warunki użytkowania, które nie są zgodne z zasadami wolnego oprogramowania. W kategorii "non-free" znajdują się na przykład sterowniki sprzętowe o ograniczonej dostępności lub oprogramowanie komercyjne.

5. Dodawanie architektury i386 do wsparcia aplikacji 32bitowych:

dpkg --add-architecture i386

dpkg: To jest narzędzie do zarządzania pakietami na systemie Debian oraz jego pochodnych, takich jak Ubuntu. Umożliwia ono instalację, usuwanie, aktualizację i zarządzanie pakietami oprogramowania na systemie.

--add-architecture i386: To opcja, która mówi dpkg, aby dodać architekturę "i386" do listy architektur obsługiwanych przez system. W praktyce oznacza to, że system będzie w stanie obsłużyć i uruchamiać 32-bitowe aplikacje i biblioteki, co może być przydatne w przypadku, gdy potrzebujesz uruchamiać starsze oprogramowanie 32-bitowe na systemie 64-bitowym.

6. Odświeżamy listę pakietów/repozytoriów:

apt update

apt: To narzędzie zarządzania pakietami, które jest często używane do instalowania,

aktualizowania i usuwania oprogramowania na systemach opartych na Debianie.

update: Jest to subkomenda narzędzia apt, która wykona operację odświeżania. Głównym celem tej operacji jest zaktualizowanie lokalnej bazy danych o dostępnych pakietach z repozytoriów systemu.

7. Instalacja narzędzia **SteamCMD** (trzeba potwierdzić STEAM LICENSE AGREEMENT):

apt install steamcmd

steamcmd: Jest to nazwa konkretnego pakietu, który chcesz zainstalować. "steamcmd" to narzędzie stworzone przez firmę Valve, które umożliwia administrację i zarządzanie serwerami gier na platformie Steam.

8. Instalacja pakietu **sudo**:

apt install sudo

sudo: Jest to nazwa konkretnego pakietu, który chcesz zainstalować. "sudo" to skrót od "Superuser Do" lub "Substitute User Do". Jest to narzędzie, które pozwala użytkownikom wykonywać polecenia jako superużytkownik (administrator) lub innego użytkownika o wyższych uprawnieniach, jeśli mają odpowiednie uprawnienia. Dzięki temu narzędziu, normalni użytkownicy systemu mogą wykonywać zadania, które zazwyczaj wymagają dostępu administratora, pod warunkiem, że zostaną uwierzytelnieni i autoryzowani przez system.

9. Tworzenie użytkownika "**steam**", który będzie odpalać i operować serwerem PalWorld:

useradd -m steam

useradd: To polecenie systemowe używane do tworzenia nowego użytkownika na systemie. Polecenie to jest zazwyczaj używane przez administratorów systemu w celu dodania nowego konta użytkownika.

-m: To opcja polecenia useradd, która wskazuje, że użytkownikowi ma być utworzony domowy katalog (home directory) o nazwie użytkownika. Domowy katalog jest miejscem, gdzie użytkownik będzie przechowywał swoje pliki i gdzie będzie miał dostęp do swojego środowiska pracy.

steam: Jest to nazwa nowego użytkownika, który ma być utworzony. W tym przypadku użytkownik będzie nazywał się "steam".

Oczywiście trzeba też ustawić hasło dla tego użytkownika za pomocą komendy "**passwd steam**".

Tworzenie oraz konfiguracja serwera PalWorld

1. Logowanie na naszego użytkownika "steam":

sudo -u steam -s

-s: To opcja polecenia "sudo", która wskazuje, że chcemy uruchomić interaktywną powłokę (zazwyczaj domyślnie bash) jako użytkownik "steam". Po wprowadzeniu tej opcji, system może poprosić o potwierdzenie hasła użytkownika "steam", jeśli jest ono wymagane, aby uruchomić powłokę.

2. Nawigacja do "/home/steam/" oraz download plików gry PalWorld (steamID: 2394010)

cd /home/steam/

***/usr/games/steamcmd +login anonymous
+app_update 2394010 validate +quit***

/usr/games/steamcmd: To ścieżka do pliku wykonywalnego narzędzia "steamcmd" na systemie. Jest to miejsce, w którym znajduje się program "steamcmd" na większości systemów Linux.

+login anonymous: Ta część polecenia wskazuje, że chcemy zalogować się do narzędzia "steamcmd" jako anonimowy użytkownik. Nie jest wymagane podawanie konkretnego konta użytkownika i hasła, co jest przydatne do pobierania publicznie dostępnych treści z platformy Steam.

+app_update 2394010: To polecenie służy do aktualizacji określonej aplikacji lub gry na platformie Steam. Numer "2394010" jest identyfikatorem **PalWorld** na platformie Steam. Wykonanie tego polecenia spowoduje, że narzędzie "steamcmd" pobierze i zaktualizuje tę aplikację lub grę do najnowszej dostępnej wersji.

validate: Ta część polecenia wskazuje, że po zaktualizowaniu aplikacji lub gry, narzędzie "steamcmd" powinno sprawdzić, czy pliki gry są poprawne i nieuszkodzone.

+quit: To polecenie mówi narzędziu "steamcmd", aby po zakończeniu aktualizacji i walidacji aplikacji lub gry, wyłączyło się samo. Jest to użyteczne, gdy chcesz, aby narzędzie automatycznie zakończyło działanie po ukończeniu zadań.

3. Tworzenie katalogów i tworzenie dowiązań symbolicznych w celu konfiguracji środowiska gry na platformie Steam

```
mkdir /home/steam/.steam && cd  
/home/steam/.steam && ln -s  
/home/steam/.local/share/Steam/steamcmd/linux32  
sdk32 && ln -s  
/home/steam/.local/share/Steam/steamcmd/linux64  
sdk64
```

mkdir /home/steam/.steam: Komenda "mkdir" służy do tworzenia nowego katalogu (folderu) o nazwie ".steam" w katalogu domowym użytkownika "steam". Tworzy ona strukturę katalogów, która będzie wykorzystywana przez klienta Steam do przechowywania pewnych danych.

&&: Jest to operator logiczny "i", który umożliwia jednocześnie wykonywanie dwóch poleceń w jednym wierszu poleceń. W tym przypadku pozwala na wykonanie pierwszej części komendy (zmiana katalogu) i kontynuowanie z drugą częścią.

cd /home/steam/.steam: Komenda "cd" jest używana do zmiany bieżącego katalogu na "/home/steam/.steam". Po wykonaniu tego polecenia terminal przenosi się do tego katalogu.

ln -s /home/steam/.local/share/Steam/steamcmd/linux32 sdk32: To polecenie "ln -s" tworzy dowiązanie symboliczne (symlink) o nazwie "sdk32" w katalogu "/home/steam/.steam", które wskazuje na ścieżkę "/home/steam/.local/share/Steam/steamcmd/linux32". Takie dowiązanie pozwala aplikacjom na odwoływanie się do lokalizacji "/home/steam/.local/share/Steam/steamcmd/linux32" za pośrednictwem "/home/steam/.steam/sdk32".

ln -s /home/steam/.local/share/Steam/steamcmd/linux64 sdk64: Podobnie jak poprzednie polecenie, to tworzy dowiązanie symboliczne o nazwie "sdk64" w katalogu "/home/steam/.steam", które wskazuje na ścieżkę "/home/steam/.local/share/Steam/steamcmd/linux64". Dzięki temu aplikacje będą mogły odwoływać się do lokalizacji "/home/steam/.local/share/Steam/steamcmd/linux64" za pomocą "/home/steam/.steam/sdk64".

4. Pierwsze odpalenie serwera w celu stworzenia plików konfiguracyjnych:

```
cd ~/Steam/steamapps/common/PalServer &&  
./PalServer.sh -useperfthreads -NoAsyncLoadingThread
```

-UseMultithreadForDS

cd ~/Steam/steamapps/common/PalServer: Polecenie "cd" służy do zmiany bieżącego katalogu na ścieżkę "/home/{nazwa_użytkownika}/Steam/steamapps/common/PalServer". Znak "~" jest skrótem oznaczającym katalog domowy bieżącego użytkownika. Komenda ta przemieszcza się do katalogu, w którym znajduje się plik "PalServer.sh".

./PalServer.sh: To jest polecenie uruchamiające skrypt o nazwie "PalServer.sh". Plik "PalServer.sh" jest wykonywany z bieżącego katalogu (czyli katalogu "PalServer"), ponieważ używamy "." przed jego nazwą. Jest to typowy sposób uruchamiania plików skryptów w systemie Unix.

-useperfthreads, -NoAsyncLoadingThread, -UseMultithreadForDS: Te są opcjami lub flagami przekazywanymi do skryptu "PalServer.sh". Są one używane do konfiguracji i uruchomienia serwera gry "PalServer" z określonymi ustawieniami. Każda flaga ma swoje własne znaczenie i wpływa na zachowanie serwera w różny sposób. Na przykład, "-useperfthreads" może włączać wydajne wątki, "-NoAsyncLoadingThread" wyłącza wątek asynchronicznego ładowania, a "-UseMultithreadForDS" włącza wielowątkowość dla systemu detekcji kolizji.

Po pierwszym odpaleniu zamykamy serwer kombinacją klawiszy **CTRL+C**

5. Kopiowanie konfiguracji domyślnej oraz wstępna konfiguracja serwera:

***cp DefaultPalWorldSettings.ini
Pal/Saved/Config/LinuxServer/PalWorldSettings.ini
&& nano
Pal/Saved/Config/LinuxServer/PalWorldSettings.ini***

cp DefaultPalWorldSettings.ini Pal/Saved/Config/LinuxServer/PalWorldSettings.ini: To polecenie "cp" służy do kopiowania pliku. W tym przypadku kopiuje plik "DefaultPalWorldSettings.ini" do lokalizacji "Pal/Saved/Config/LinuxServer/PalWorldSettings.ini". To oznacza, że zawartość pliku "DefaultPalWorldSettings.ini" zostaje skopiowana i zapisana jako "PalWorldSettings.ini" w podanej lokalizacji.

nano Pal/Saved/Config/LinuxServer/PalWorldSettings.ini: To jest polecenie używane do otwarcia pliku "PalWorldSettings.ini" w edytorze tekstu o nazwie "nano". Edytor "nano" jest często stosowany w systemach Unix do edycji plików tekstowych w terminalu.

W tym pliku przeprowadzamy wstępną konfigurację jak np. Max ilość graczy, nazwa serwera, opis serwera, hasło admina, hasło dostępu na serwer itp.

```
steam@srv467669: ~/Steam/steamapps/common/PalServer
GNU nano 5.4 Pal/Saved/Config/LinuxServer/PalWorldSettings.ini
; Changes to this file will NOT be reflected on the server.
; To change the server settings, modify Pal/Saved/Config/LinuxServer/PalWorldSettings.ini.
[Script/Pal.PalGameWorldSettings]
UseAuth=True,BanListURL="https://api.palworldgame.com/api/banlist.txt")
onSelectByMap=True,bExistPlayerAfterLogout=False,bEnableDefenseOtherGuildPlayer=False,CoopPlayerMaxNum=4,ServerPlayerM

[ Read 7 lines (Converted from DOS format) ]

^X Exit      ^R Read File  ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line M-E Redo    M-C Copy

steam@srv467669: ~
GNU nano 5.4 Steam/steamapps/common/PalServer/Pal/Saved/Config/LinuxServer/PalWorldSettings.in
[Script/Pal.PalGameWorldSettings]
layerMaxNum=32,ServerName="BetterPals Polska",ServerDescription="BetterPals to tutaj",AdminPassword="

^G Help      ^O Write Out  ^W Where Is  ^K Cut       ^T Execute   ^C Location  M-U Undo    M-E Redo
^X Exit      ^R Read File  ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line M-E Redo
```

nano

**Steam/steamapps/common/PalServer/Pal/Saved/Config/LinuxServer/PalWorldSettings.in
i**

Zapisujemy konfiguracje.

6. Odpalamy serwer:

***cd ~/Steam/steamapps/common/PalServer &&
./PalServer.sh -useperfthreads -NoAsyncLoadingThread
-UseMultithreadForDS***

Ewentualne aktualizacje gry

1. Zamknięcie serwera (**CTRL+C**)

```
starting serial terminal on interface serial0

^CFUnixPlatformMisc::RequestExitWithStatus
FUnixPlatformMisc::RequestExit
Exiting abnormally (error code: 130)
Shutdown handler: cleanup.

steam@srv467669:~/Steam/steamapps/common/PalServer$
```

2. Odpalenie komendy na update:

***/usr/games/steamcmd +login anonymous
+app_update 2394010 validate +quit***

```
steam@srv467669:/root$ cd /home/steam/
date +quit67669:~$ /usr/games/steamcmd +login anonymous +app_update 2394010 vali
(491) burning pthread_key_t == 0 so we never use it
Redirecting stderr to '/home/steam/Steam/logs/stderr.txt'
[OK] directory: '/home/steam/Steam/logs'
[OK] Checking for available updates...
[OK] Verifying installation...
Steam Console Client (c) Valve Corporation - version 1705108307
Type 'quit' to exit --
Using Steam API...dlmopen steamservice.so failed: steamservice.so: cannot open shared objec
t file: No such file or directory

[OK] Connecting anonymously to Steam Public...OK
[OK] Waiting for client config...OK
[OK] Waiting for user info...OK
Update state (0x5) verifying install, progress: 1.07 (24117248 / 2248053389)
Update state (0x5) verifying install, progress: 41.58 (934753345 / 2248053389)
Update state (0x5) verifying install, progress: 97.87 (2200121548 / 2248053389)
Success! App '2394010' fully installed.
steam@srv467669:~$ █
```


3. Ponowne odpalenie serwera:

***cd ~/Steam/steamapps/common/PalServer &&
./PalServer.sh -useperfthreads -NoAsyncLoadingThread
-UseMultithreadForDS***

```
erfthreads -NoAsyncLoadingThread -UseMultithreadForDSver && ./PalServer.sh -usep  
Shutdown handler: initialize.  
Increasing per-process limit of core file size to infinity.  
sh: 1: xdg-user-dir: not found  
dlopen failed trying to load:  
steamclient.so  
with error:  
steamclient.so: cannot open shared object file: No such file or directory  
[S_API] SteamAPI_Init(): Loaded '/home/steam/.local/share/Steam/steamcmd/linux64/steamclient.  
so' OK. (First tried local 'steamclient.so')  
CAppInfoCacheReadFromDiskThread took 3 milliseconds to initialize  
dlopen steamservice.so failed: steamservice.so: cannot open shared object file: No such file  
or directory  
Setting breakpad minidump AppID = 2394010  
[S_API FAIL] Tried to access Steam interface SteamUser021 before SteamAPI_Init succeeded.  
[S_API FAIL] Tried to access Steam interface SteamFriends017 before SteamAPI_Init succeeded.  
[S_API FAIL] Tried to access Steam interface STEAMAPPS_INTERFACE_VERSION008 before SteamAPI_I  
nit succeeded.  
[S_API FAIL] Tried to access Steam interface SteamNetworkingUtils004 before SteamAPI_Init suc  
ceeded.
```

Wynik końcowy - mój publiczny serwer na 32 sloty o nazwie
“BetterPals Polska”



Filip Smol – Informatyka niest. Sem. 3. GC01

Filip Smol – Informatyka niest.
Sem. 3. GC01