

Dokumentacja migracji na środowisko linux.

Maciej Tracz

Technikum Mechatroniczne nr 1 w Warszawie

Rok 2020

Rozdział 1: Wstęp

1.1 Omówienie tematu

Każdego dnia korzystamy z urządzeń elektronicznych. Pomagają nam wykonywać najróżniejsze zadania. Potrafimy mocno się do nich przywiązać, lecz pod koniec dnia są tylko narzędziem. Właśnie to przywiązanie często przeszkadza nam w spostrzeżeniu innych sposobów, aplikacji czy urządzeń mogących poprawić naszą produktywność. Podczas kupna laptopa czy nowego komputera często nie zastanawiamy się z jakim oprogramowaniem będzie nam dane pracować. Zostawiamy to na później wierząc, że jeśli pojawi się jakiś problem, będziemy w stanie go rozwiązać.

Podstawowym mostem między nami, **Użytkownikiem**, a częściami które tworzą komputer, **Hardwarem**, jest system operacyjny, **OS**.

To właśnie OS pozwala nam komunikować się z procesorem czy pamięcią aby uzyskać zasoby lub je zapisać. Tworzy dla nas wygodne środowisko pracy oraz oferuje różne narzędzia, które mają nam ją ułatwić.

Zazwyczaj temat systemu operacyjnego jaki zamierzamy zainstalować na nowo kupionej maszynie jest omijany lub na ty tyle oczywisty że pomijany zupełnie. **OEMi (Original Equipment Manufacturer)** przyzwyczaili nas, że jesteśmy wyřeczani w tym zakresie. Prawie każdy laptop i gotowy komputer osobisty mają dołączoną jedną z wersji Windowsa lub w przypadku Apple, MacOsa.

Te przyzwyczajenia nie wzięły się z nikąd. Rynek systemów operacyjnych jest zdominowany przez dwie firmy: Microsoft i Apple. To one stanowią filary infrastruktury biurowych oraz są nieodłącznymi kompanami każdego użytkownika. Ta sytuacja zamknęła oczy konsumentom i zabrała wybór, który im się należy.

Ta dokumentacja ma właśnie na celu skierowanie światła na alternatywę w postaci **LINUX** dla powyżej wymienionych rozwiązań oraz wyjaśnienia w przystępny sposób migracji między systemami, niuansów z nią związanych i najważniejszych elementów użytkowania.

1.2 Zanim zaczniesz

Dokumentacja stworzona jest dla osób lekko lub średnio zaawansowanych w technologiach informatycznych. Będzie zawierać opisy technologii, zagadnień i terminów dotyczących procesu migracji i samych systemów operacyjnych. Część rozdziałów jest przystosowana do samodzielnej implementacji. Poniżej przedstawiam metodykę odczytywania treści, ze względu na stylistykę:

- **Pogrubięne zostaną ważne pojęcia, które warto zapamiętać**
- *Kursywą zostaną zapisane cytaty z dokumentacji twórców lub forów społeczności*
- Podkreślone zostaną słowa grające kluczową rolę w zrozumieniu zagadnienia

Spis treści

1	Wstęp	1
1.1	Omówienie tematu	1
1.2	Zanim zaczniesz	1
2	Czym jest linux?	3
2.1	Systemy operacyjne - rodzaje, budowa, cechy	3
2.2	Różnice między Linuxem a Windowsem	4
3	Korzyści i problemy migracji	5
3.1	Podstawowe problemy migracji	5
3.2	Zalety zmiany środowiska	6
3.3	Przed migracją	6
4	Rodzaje dystrybucji	8
4.1	Modele licencjonowania	8
4.2	Dystrybucje Linuxa	9
5	Zmiana systemu	12
5.1	Instalacja	12
5.1.1	Tworzenie nośnika	12
5.1.2	Instalacja	13
5.1.3	Konfiguracja poinstalacyjna	18
5.2	Nauka obsługi systemu oraz dobre nawyki	19
6	Aplikacje i środowiska alternatywne	20
6.1	Multimedia	20
6.2	Twórczość	20
6.3	Programowanie	21
7	Automatyzacja zadań	22
7.1	Skrypty	22
7.1.1	Tworzenie skryptów	22
7.2	Kontenery	23
8	Podsumowanie	25
9	Źródła	26

Rozdział 2: Czym jest linux?

2.1 Systemy operacyjne - rodzaje, budowa, cechy

Razem z narodzinami pierwszych komputerów przyszła wizja coraz to większej kontroli nad zasobami maszyny. Pierwsze systemy operacyjne były raczej namiastką dzisiejszych i potrafiły jedynie kolejkować procesy aby zwiększyć wydajność. Na początku lat '60 zaczęto poważniej brać ich rozwój. Dodano wiele ważnych elementów, które zostały się do dziś. Od systemy dla indywidualnych maszyn, przez IBM DOS oraz Windows 1.0, do MacOS i Windows 10, tak ciągnęła się historia. A u boku tego rozwoju stał jeden człowiek, który miał trochę inną wizję. Linus Torvalds, historyczny twórca rdzenia Linuxa, postanowił dać pasjonatom to czego najbardziej potrzebowali. Otwartości, bezpieczeństwa i przystępności. To właśnie zapewnia **Open Source** natura linuxa, dystrybuowanego na licencji **GNU GPL**.

Ale czym właściwie jest system operacyjny?

Jest to oprogramowanie systemowe, które zapewnia kontrolę i komunikację nad podzespołami urządzenia, zasobami systemowymi oraz umożliwia korzystanie z serwisów tworzących środowisko robocze dla użytkownika.

Składa się on z:

- Jądra systemowego - Służy on do wykonywania i kontrolowania zadań przydzielonych systemowi. Zarządza wszystkimi zasobami maszyny i komunikacją wszystkich elementów jednostki.
- Powłoki - zapewnia komunikację między użytkownikiem a systemem operacyjnym
 - Tekstowe - tak zwane 'terminale'. Pozwalają za pomocą komend zarządzać zasobami i wykonywać operacje
 - Graficzne(GUI) - interfejsy pozwalające użytkownikowi zobaczyć zasoby w bardziej przystępnej formie. Wspierają operacje myszką i ograniczają potrzebę znania komend do minimum.
- Systemu plików - pozwala zapewnić dostęp i stworzyć strukturę plików dla użytkownika. Dostarcza zabezpieczenia prywatności oraz szyfrowanie jeśli odpowiednio dobrane i skonfigurowane.
- Aplikacje wbudowane - zestaw aplikacji zawartych w obrazie instalowanego systemu. To one dostarczają możliwości Out-Of-The-Box, pozwalające używać komputera już przy pierwszym logowaniu. Ten zestaw zależy zupełnie od twórców.

Ważnym jest tu zauważenie, że w przeciwieństwie do Windowsa i MacOS, które są jednolitymi systemami, Linux jest jądrem. Oznacza to w praktyce, że referując do Systemu operacyjnego linux mówimy o jego dystrybucjach, o czym więcej w późniejszych rozdziałach.

2.2 Różnice między Linuxem a Windowsem

Na samym początku szukania różnic między tymi produktami ważne jest, żeby zrozumieć czym one są. **Microsoft Windows** to rodzina systemów operacyjnych bazująca na jądrze NT. Zawierają standardowe aplikacje systemowe, których nie można odinstalować. Posiada wbudowaną powłokę Eksplorator Windows, która dostarcza niezbędnych narzędzi do komputera. Linux, jak już wspomniałem, jest samym jądrem dostarczającym kluczowych funkcjonalności. Jest dystrybuowany na otwartej licencji pozwalającej samemu ingerować w jego strukturę jak i integrację.

Systemy te wyróżniają się następującymi cechami:

- Licencjonowanie - w momencie kupna licencji na Windows 10 faktycznie dostajemy prawo tylko do użytkowania produktu na jednej maszynie. Dystrybucje linuxa są natomiast z reguły darmowe poza specjalistycznymi edycjami przeznaczonymi dla profesjonalistów.
- Zawartość systemu:
 - Aplikacje wbudowane - Zestaw Windowsa jest ubogi i wymaga od użytkownika samodzielnej instalacji kluczowych aplikacji. Większości z nich nie można odinstalować, lecz jedynie zastąpić.
Linux pozwala nam wybrać pakiety wedle naszego zapotrzebowania. Te same dystrybucje posiadają różne wersje z względu na własnie zawartość aplikacji wbudowanych. Takie listy też można edytować i tworzyć własne obrazy systemowe, zaoszczędzając czas i miejsce dyskowe.
 - Powłoki - Produkt Microsoftu udostępnia nam dwie powłoki tekstowe cmd i Powershell oraz interfejs graficzny Eksplorator Windows. Są one nieodłączną częścią systemu i nie podlegają wymianie. Linux dostarcza użytkownikowi takie powłoki jakie potrzebuje. Pozwala tworzyć własne lub modyfikować istniejące w razie takiej potrzeby.
 - Kod źródłowy - Dostępny jest tylko na linuxie. Umożliwia dogłębną analizę działania jądra oraz modyfikacji go wedle własnych wymagań.
- Systemy plików - Microsoft słynie z swojego systemu o nazwie NTFS. Pozwala on na zarządzanie uprawnieniami dostępu, pozwala szyfrować dane w przyjazny sposób oraz współgra z Zasadami Grup, które są potężnym narzędziem. To sprawia, że Windows a za nim NTFS są najczęściej stosowane w szkołach, biurach, kafejkach czy innych miejscach z dużą liczbą jednostek. Linux natomiast pozwala na większą różnorodność. Systemy te są stworzone zazwyczaj pod jednego użytkownika, zapewniają większą wydajność oraz formy możliwych zabezpieczeń.
- Sterowniki i oprogramowanie - jako użytkownik Linuxa musisz przygotować się na konieczność kombinowania i szukania pomocy u społeczności, gdy okaże się że program lub urządzenie, które tak bardzo chesz mieć, wspiera tylko Windowsa. Coraz rzadziej widzimy takie sytuacje lecz zawsze pozostaje ryzyko. Dzieje się tak ponieważ Windows jest bardzo popularny i stanowi procentową większość w biurach czy szkołach, przez co niektóre firmy po prostu nie czują potrzeby wydawania sterowników wspieranych przez dystrybucje Linux.
- Zabezpieczenia - pomimo iż można uzbroić się w najróżniejsze antywirusy lub zasady firewall, niezmiennym pozostaje fakt, że przez swoją popularność Windows przyciąga większą uwagę hakerów, twórców szkodliwego oprogramowania czy złodziei. Linux natomiast swoją otwartością mógłby się wydawać prostrzy do atakowania. Ma to swoje podłoża, lecz równie prostsze jest szukanie rozwiązań przez społeczność i łatanie tych problemów. Zapewnia też dużo większe zabezpieczenia i dokładniejszy wgląd na to co dzieje się z maszyną.

Rozdział 3: Korzyści i problemy migracji

3.1 Podstawowe problemy migracji

Nie ważne czy jest to nowy samochód, nowa myszka czy buty, każdy z nas musiał kiedyś kupić nową rzecz, która była zupełnie inna od tej do której byliśmy przyzwyczajeni. Gdy jeszcze nie zdążyliśmy przyzwyczać się do narzędzia z jakim pracujemy zdarza się, że czujemy dyskomfort lub zakłopotanie. Nasza pamięć mięśniowa, ani nawyki motoryczne zaczynają wariować. Podobnie bywa przy przechodzeniu na nowy system operacyjny, który de facto jest narzędziem. W tym rozdziale przedstawię z jakimi problemami można się zetknąć i jak sobie z nimi radzić.

Najważniejsze problemy:

- Proces instalacji systemu
 - Zmiana interfejsu graficznego
 - Potrzeba używania terminalu
 - Instalowanie oraz aktualizacja aplikacji
 - Aplikacje użytkowe
1. Proces instalacji systemu - podczas prób instalowania dystrybucji linuxa, zależnie od jej zaawansowania, możemy natrafić na kroki konfiguracyjne technologie nie spotykane w instalatorze windowsa. Brak ich zrozumienia może prowadzić do późniejszych błędów, a takie potrafią popsuć pierwsze wrażenia z systemem. Warto jest postępować wedle poradników od samych twórców lub osób zaawansowanych.
 2. Zmiana interfejsu graficznego - pomimo, że zarówno interfejs Windowsa jak i GNOME/MATE/Cinnamon/Xfce (najpopularniejsze Interfejsy stosowane w dystrybucjach linux) były tworzone ramię w ramię, czerpiąc od siebie co rusz to nowe elementy, używając obu można zauważyć sporo różnic. Jedne z nich są błache i można ich nie zauważyć, inne natomiast mogą wymagać pełnej zmiany przyzwyczajęń. Warto podchodzić do nich powoli i dać sobie czas aby nabrać nawyki związane z nowym interfejsem.
 3. Potrzeba używania terminalu - aplikacje cmd oraz powershell, bywają raczej zagadką dla przeciętnego użytkownika produktu Microsoftu. Są one zaawansowanymi narzędziami do zarządzania zasobami, jednak praktycznie wszystkie potrzebne operacje można wykonać przez GUI. Systemy Linuxa są jednak odmienne, gdyż wymagają korzystania z terminalu. Wiele ustawień wymaga od nas zmiany wartości w plikach systemowych. Aplikacje potrzebne do wykonywania pracy często pisane są w języku bash, który uruchamiamy w konsoli. Warto także wspomnieć, że wiele zasobów jest dostępnych praktycznie tylko przez konsolę lub bez niej wymagają dużego wysiłku.
 4. Instalowanie oraz aktualizacja aplikacji - zapewne każdy spotkał się z plikami .exe, które instalują aplikacje na urządzeniach z Windowsem. Charakteryzują się one tym, że konfiguruje pliki systemowe tak aby współgrały z aplikacją, dlatego też nie można przenosić aplikacji z

maszyny na maszynę. Są one głęboko zakorzenione. Po drugiej stronie muru mamy rozwiązanie, które nie jednego nowego użytkownika przyprawiło o zawrót głowy. Jest to manager pakietów. Pozwala on na pobieranie, instalację oraz aktualizację narzędzi przy pomocy jednej komendy. Dzięki listą repozytoriów wie ona gdzie szukać plików instalacyjnych w internecie oraz eliminuje element wyszukiwania lików z instalatorem. Aktualizacje odbywają się ręcznie lub automatycznie (przy odpowiedniej konfiguracji) i zapewniają uaktualnienie wszystkich narzędzi, a nie tylko pojedynczych.

5. Aplikacje użytkowe - nieodłączną częścią zmiany systemu operacyjnego jest zmiana narzędzi. Jak już wcześniej wspominałem, wiele aplikacji z których korzystamy na codzień może nie być wspierane na nowej platformie. Wymusza to poszukiwanie nowych rozwiązań, które może i nie jest przyjemne, ale pozwala poszerzyć nasze horyzonty i dać nam wcześniej niedostępne możliwości.

3.2 Zalety zmiany środowiska

Różnorodność. To pierwsze słowo jakie przychodzi mi do głowy gdy pomyślę o Linuxie. Każdy pomysł związany z używaniem go wiązał się z odkrywaniem nowych rozwiązań, narzędzi lub technologii. Domyślne aplikacje potrafią przyćmić nam obraz rynku jakie stanowią, przez co odebrać nam możliwość prubowania nowych i ciekawych rzeczy.

Rozwój. Na każdym kroku stawiane są przed tobą małe lub większe wyzwania. To one wymuszają na tobie potrzebę zrozumienia tematu/zagadnienia. Za każdym wyzwaniem natomiast idą nowe zestawy drzwi, które możemy otworzyć, poznając nowy i zupełnie odmienny świat. Przy odpowiedniej motywacji i cierpliwości nie będzie rzeczy niemożliwych, a nawet z tych najbardziej męczących będą wypływać najlepiej uczące lekcje.

Bezpieczeństwo. Za większą otwartością idzie więcej rozwiązań dostarczających zabezpieczenia. Często wymagają one jednej lub dwóch komend w terminalu by zapewnić ochronę danych na dysku, prywatne przeglądanie internetu czy bezpieczne połączenie z drugą maszyną. Przewagą na tym polu Linuxa nie polega na tym, że technologie są lepsze, bo na windows znajdziemy równie skuteczne rozwiązania. Chodzi tutaj ściśle o dostępność i konwencjonalność dla przeciętnego użytkownika, z czym ten system radzi sobie bardzo dobrze.

3.3 Przed migracją

Przed zmianą środowiska pracy warto zadać sobie parę pytań sprawiających, że przenosiny będą prostsze i bardziej satysfakcjonujące. Przyczynią się też do lepszych wrażeń podczas pierwszej poważnej styczności z Pingwinem. Czynności te można wykonywać w różnej kolejności choć zaproponowana jest rekomendowana.

Czynności przed migracją:

- Czy masz motywację? - Zadać sobie to ważne pytanie, ponieważ od niego zależy czy uda ci się przetrwać okres nauki i rozwiązywania wyzwań. To przede wszystkim motywacja stanowi kluczowy element przy większości niepowodzeń i zrażeń się do systemów Linux.
- Czy chcesz pozbyć się Windowsa? - Zdefiniuje to którą metodę instalacji i korzystania z systemu wybierzesz. Często osoby nie mogą wyzbyć się produktu Microsoftu z względu na pracę lub szkołę. Wtedy przychodzi nam rozwiązanie podwójnego rozruchu o którym mowa jest w późniejszym rozdziale.

- Gdzie chcesz trzymać Pingwina? - Często pomijany temat, lecz ważny dla tych którzy nie czują się pewnie przy instalowaniu systemów i boją się o swoje dane. Warto pomyśleć czy mamy stary laptop lub PC, który może posłużyć jako urządzenie próbne lub dodatkowa maszyna do nauki.
- Do czego wykorzystujesz system? - Nawiązanie do tego pytanie znajduje się w następnym rozdziale. Ta odpowiedź zawęży wybór dystrybucji do bardziej przyjaznych ilości. Razem z dystrybucją będzie szło GUI jakie nam najbardziej będzie sprzyjać.

Gdy już odpowiesz sobie na te cztery ważne pytania, będziesz o wiele bliżej celu. Zapamiętaj odpowiedzi, bo będą ci potrzebne w następnych rozdziałach. W tym momencie podróży nie warto się śpieszyć tylko dokładnie przeanalizować swoje potrzeby, bo nie warto psuć sobie pierwszych wrażeń przez nieodpowiednie GUI lub format niewłaściwego dysku.

Rozdział 4: Rodzaje dystrybucji

4.1 Modele licencjonowania

Pomyśl o programach, których używasz na co dzień. Za ile z nich płacisz miesięczne opłaty? Ile było kupionych jednorazowo i masz do nich dostęp do dziś. A jakie są darmowe i pozwalają ci swobodnie malipulować sobą bez konsekwencji? To wszystko określają modele licencjonowania.

Gdy spojrzymy dziś na rynek laptopów zauważamy pewną zależność, większość z nich ma przypisany system Windows 10 do siebie. Dla producenta kupno licencji hurtowo jest opłacalne i nie musi dokładać z własnej kieszeni, my natomiast nie martwimy się o dodatkowy wydatek na system operacyjny lub jego instalację.

Teraz przez głowę może przechodzić pytanie, skoro to jest takie wygodne i opłacalne to czemu powiedziałem, że 'większość' laptopów ma przypisaną licencję zamiast 'wszystkie'?

Odpowiedź jest bardzo prosta, a wynika ona z udziału rynkowego.

Producenci wiedzą jak wygląda zapotrzebowania na systemy operacyjne, czyli w znacznej większości jest to produkt Microsoftu, jednak od czasu do czasu w sklepie może pojawić się kupiec proszący o czystą maszynę bez żadnego systemu na niej. Klient taki swoją potrzebą może zmniejszyć koszt produktu i uczynić go bardziej opłacalnym dla kolejnych kupujących.

Takie sytuacje mogą zachodzić z względu na różnorodność modeli licencjonowania. Każdy producent sam wybiera, który mu i jego odbiorcom pasuje, bowiem nie każdy program potrzebuje być płatny jeśli może tworzyć go społeczność zapalonych twórców, a także wyspecjalizowane narzędzia muszą znaleźć swój segment cenowy. Przedstawię teraz podstawowe modele i omówię ich najważniejsze cechy

- Licencje płatne
 - Płatne własnościowe - podczas dokonania transakcji dostajemy w własne ręce kopię programu, do którego mamy dożywotni dostęp. Zawierają konkretną wersję oprogramowania i rzadko wspierają ulepszenia. Można pośród nich wyróżnić te, które pozwalają na modyfikację samą kopią (modyfikacją plików lub tworzeniem własnych dodatków) oraz te które pozwalają jedynie z kopii korzystać.
 - Płatne użytkowe - coraz częściej spotykane rozwiązanie polegające na wykupieniu czasowego dostępu do usługi. Pozwala to zmniejszyć wstępny koszt oraz pozwolić użytkownikowi zrezygnować w dowolnym momencie zgodnie z umową. Takie licencje skłaniają wydawców do częstego naprawiania, ulepszania i poszerzania produktu, co pomaga trzymać klientów jak najdłużej.
- Licencje darmowe
 - Użytkowe - bardzo popularna opcja dla rozwijających się produktów. Pozwala udostępnić funkcjonalności i oczekiwać od części użytkowników komunikatów zwrotnych o problemach i promozycjach rozbudowy. Jesteśmy przy tym upoważnieni tylko do użytkowania aplikacji chyba, że umowa określa inne warunki. Ważnymi rodzajami tego modelu jest Freemium oraz Donationware.

Pierwszy jest połączeniem słów Free i Premium. Takie podejście zapewnia produkt premium z drobnymi ograniczeniami dla każdego użytkownika jak i możliwość zakupu ulepszonej, pełnej wersji. Dobrymi przykładami są to muzyczne serwisy streamingowe oraz Duolingo.

Drugim jest fundowanie rozwoju poprzez dotacje. Dostraczone usługi są w swojej pełnej formie i ma do nich dostęp każdy klient. Taki użytkownik po dobrych doświadczeniach może zdecydować się na wsparcie twórców wysyłając im niewielką sumę pieniędzy (czasem sumy te nie są tak małe). Najpopularniejszą usługą korzystającą z tego modelu jest Wikipedia.

- Open-sorce - jest to dodatkowa cecha, która mówi o dostępności kodu źródłowego dla użytkownika. Znaczy to, że jest on w stanie na własną rękę zarządzać, manipulować lub ulepszać własną kopię produktu. Daje możliwość wykazania się społeczności swoimi umiejętnościami i kreatywnością. Wspomaga to także wyszukiwanie błędów i niepoprawości, a za tem także rozwiązywanie takowych. Produkt omawiany w owej dokumentacji, Linux, jest jednym z największych przykładów Open-Sorce. Jego dokładna licencja to GNU GPL.

4.2 Dystrybucje Linuxa

Wymienione systemy są tylko najpopularniejszymi opcjami, nie będę tutaj opisywał ich dokładnie, lecz dostarczałem linki do materiałów referencyjnych do dalszego wglądu. Zachęcam do własnego rozeznania po zapoznaniu się z działem. Pamiętaj, żeby wybierać najnowsze, rekomendowane przez developerów wersje oprogramowania.

Najważniejsze dystrybucje:

. MX Linux

Dosyć młody produkt, który cały czas się rozwija i coraz bardziej zadziwia. Dostarcza on proste i przejrzyste środowisko graficzne. Jest dystrybucją lekką, co oznacza, że będzie świetnie działał nawet na starszych jednostkach. Jest także skonfigurowany tak aby użytkownik mógł korzystać z niego zaraz po zainstalowaniu.

Strona internetowa: <https://mxlinux.org>

Dodatkowe informacje: <https://distrowatch.com/mxlinux>

. Linux Mint

Jeden z najpopularniejszych Linuxów. Usytuował się przez lata swojej obecności na rynku jako najlepsza opcja dla wszystkich, którzy chcą prosty system bez potrzeby konfiguracji, działający na każdej maszynie jaką można mieć pod ręką. Jest bardzo stabilny oraz posiada szeroką społeczność użytkowników wspierających rozwój. Pozwoli ci to rozwiązać, każdy problem na jaki możesz natrafić. Daje trzy różne wersje jak i możliwość dalszego dostosowania systemu pod siebie.

Strona internetowa: <https://linuxmint.com/>

Dodatkowe informacje: <https://distrowatch.com/mint>

. Ubuntu

Najpopularniejsza dystrybucja jaka kiedykolwiek powstała. Zaskarbiła sobie serca użytkowników prostotą użytkowania oraz możliwościami konfiguracji interfejsu wedle własnych zachcianek. Posiada ogromną bibliotekę wspieranych aplikacji oraz sterowników, a także podobnie jak w przypadku Minta, fora społeczności czy grupy dyskusyjne pozwalają zdobyć pomoc przy każdym problemie. Na jej bazie powstało wiele oddzielnych wersji takich jak Lubuntu czy Xubuntu (zazwyczaj podstawowymi różnicami jest poziom optymalizacji pod konkretne parametry systemowe).

Strona internetowa: <https://ubuntu.com/>

Dodatkowe informacje: <https://distrowatch.com/ubuntu>

Alternatywne wersje (Xubuntu): <https://xubuntu.org/>

Alternatywne wersje (Lubuntu): <https://lubuntu.net/>

. Debian

Ojciec wielu systemów. To na jego bazie powstała znaczna większość dzisiejszych dystrybucji linuxa. Jednak sam Debian także ma wiele do zaoferowania. Nie jest to jednak system przystosowany dla początkującego użytkownika, ze względu na swoją surową formę. Po instalacji otrzymujemy dostęp tylko do powłoki w postaci terminalu. Dopiero po jej uruchomieniu możemy sami zainstalować własne komponenty tworzące system. Dostarcza to stabilności oraz optymalizacji zasobów. Z tego względu Debian stanowi najważniejszy filar w branży informatycznej, jako baza dla rozwiązań od serwerów WWW przez serwery obliczeniowe do chmury. Jeżeli jednak zamierzasz tworzyć serwery gier multimedialnych dla znajomych, Debian to pewniak.

Strona internetowa: <https://www.debian.org/>

Dodatkowe informacje: <https://distrowatch.com/debian>

. Arch Linux

Arch dostarcza zupełną dowolność w wyborze pakietów poszerzających możliwości systemu. Jest to nie tylko wartościowe z perspektywy graficznej, choć i tu Arch jest uznawany za króla, ale także umożliwia dokładną modyfikację zużycia baterii lub zwiększania wydajności. Jednak jak z każdym 'zrób to sam' przychodzi problem. Proces instalacji wymaga użycia komend oraz zrozumienia zasad działania linuxa, dlatego instalacja dokonywana przez świeżego użytkownika powinna być przy wsparciu poradnika, a wszystkie konieczne modyfikacje przedstawione przez społeczność lub inny poradnik. Często ten żmudny proces odstrasza, dlatego też odradza się wybieranie go jako pierwszy system. Jednakowo jeśli chcesz dać sobie szansę to jest duża szansa, że nie pożałujesz żadnej spędzonej nad nim minuty. Ważnym też jest, że Arch jest, tak jak Debian czy Ubuntu, budowany od podstaw. Wymaga to korzystania z oddzielnych repozytoriów przystosowanych pod niego jak i innych form zapisu komend lub programów.

Strona internetowa: <https://archlinux.org>

Dodatkowe informacje: <https://distrowatch.com/archlinux>

. Manjaro

Jeśli po zapoznaniu się z Archem, poczułeś zainteresowanie, ale boisz się skomplikowanego procesu Manjaro przychodzi z pomocą. Z względu na właśnie takie zapotrzebowanie rynku powstała ta dystrybucja. Skupia ona pozytywne cechy Archa, będąc budowanym na jego podstawie i składa je w pakunek przystępny dla nowego użytkownika, oferując doświadczenie Out-Of-The-Box, nie wymaga dużej wiedzy ani konieczności konfiguracji wszystkich jego części. Zaskarbił swoje imię stabilnością i łatwością użycia. Jego użytkowanie przygotowuje również do możliwości przejścia na Archa gdy pocujemy niedosyt. Jeśli do tej pory nie masz wybranej dystrybucji zalecam rozpocząć właśnie od Linux Manjaro.

Strona internetowa: <https://manjaro.org/>

Dodatkowe informacje: <https://distrowatch.com/manjaro>

. Kali Linux

Potężny i ciężki system, skierowany w stronę etycznych hakerów, testerów bezpieczeństwa, inżynierów systemów ale także wszystkim, którzy chcą takimi zostać lub skorzystać z ich narzędzi. Zawiera on wszystkie programy i biblioteki o jakie mogłby poprosić ktoś w wyżej wymienionych. Nie jest to jednak opcja rekomendowana do zwykłego użytku gdyż długo się ładuje, a jej optymalizacja nie współgra z użytkowaniem biurowym lub przeglądaniem internetu. Warto jednak posiadać ją jako drugi system lub wirtualną maszynę by nie musieć się martwić o dostępność narzędzi ani konieczność ich instalacji. Jeśli security to Kali. Jeśli natomiast potrzebujesz czegoś lżejszego ale wciąż posiadającego możliwości, spróbuj ParrotOS

Strona internetowa: <https://kali.org/>

Dodatkowe informacje: <https://distrowatch.com/kali>

ParrotOS: <https://parrotlinux.org>

Jeśli jednak potrzebujesz systemu oferującego inny zestaw funkcji nie ograniczaj się do wymienionych tutaj dystrybucji. Są to najbardziej popularne opcje stworzone pod potrzeby większości konsumentów. Inne, mniejsze produkty, mogą natomiast posiadać listę cech idealną dla Ciebie.

Świat linuxa polega na eksperymentowaniu i doświadczaniu mocy Open-Sorce na własnej skórze. Pamiętaj także, że nie musisz zostawać z jednym systemem na komputerze. Dzięki rozwiązaniu dual/triple/multi -boot możesz posiadać wszystko czego potrzebujesz. Ogranicza Cię tutaj tylko miejsce na dysku i zdrowy rozsądek. Warto próbować nowych połączeń i szukać idealnego dla swoich potrzeb.

Poniżej zamieszczam pomocne linki do stron skierowanych do osób szukających dystrybucji dla siebie.

Kwestionariusz wyboru dystrybucji: <https://distrochooser.de/en/>

Blog LinuxConfig.org: <https://linuxconfig.org/choose-distro>

Rozdział 5: Zmiana systemu

Uwaga! Proces instalowania wymaga operacji na dyskach twardych twojej maszyny. Może to wiązać się to z omyłkowym usunięciem danych lub ich zniszczeniem. Przez rozpoczęciem upewnij się, że masz zrobioną kopię zapasową wszystkich ważnych plików i/lub poprzedniego systemu operacyjnego.

5.1 Instalacja

Proces instalowania warto podzielić na 3 etapy:

1. Tworzenie nośnika ISO
2. Instalacja systemu na docelowej maszynie
3. Konfiguracja poinstalacyjna

Teraz kolejno przejdziemy przez wszystkie 3 na przykładzie Linux Manjaro XFCE 20.0.3.

5.1.1 Tworzenie nośnika

Aby móc zainstalować system operacyjny na komputerze potrzebujemy urządzenia będącego nośnikiem obrazu takiego systemu. Do tego potrzebujemy pendriva conajmniej 8GB oraz obrazu systemu w postaci ISO. Taki można pobrać na stronach twórców w zakładce Downloads. Gdy mamy już te elementy potrzebujemy narzędzia, które przygotuje pendriva jako nośnik. Polecam Rufusa dla użytkowników Windowsa oraz UByte Software na MacOS. Nie są to jednak jedyne opcje i warto poszukać czy aktualnie nie wyszły nowsze, lepsze narzędzia.

Windows

Rufus: <https://rufus.ie/>

YUMI: <https://www.pendrivelinux.com/yumi-multiboot-usb-creator/>

MacOS

UByte Software: <https://www.uubyte.com/download/uubyte-iso-editor.dmg>

Disk Utility - The Default ISO Buner (narzędzie wbudowane, polecane na starych urządzeniach.)

Teraz gdy masz wszystko co potrzebne, postępuj zgodnie z instrukcją na stronie lub po prostu dodaj ISO i wybierz domyślne ustawienia. Jeśli posiadasz już odpowiednio przygotowanego pendriva przejdźmy do następnego etapu.

5.1.2 Instalacja

Przedstawię tutaj przykładową instalację Linuxa Manjaro w najnowszej dla mnie wersji. Wiele elementów będzie niezmienna w przypadku większości dystrybucji. Jednak jeśli nie jesteś pewien swoich umiejętności adaptacji zachęcam do poszukanie w internecie poradników przystosowanych do twoich potrzeb.

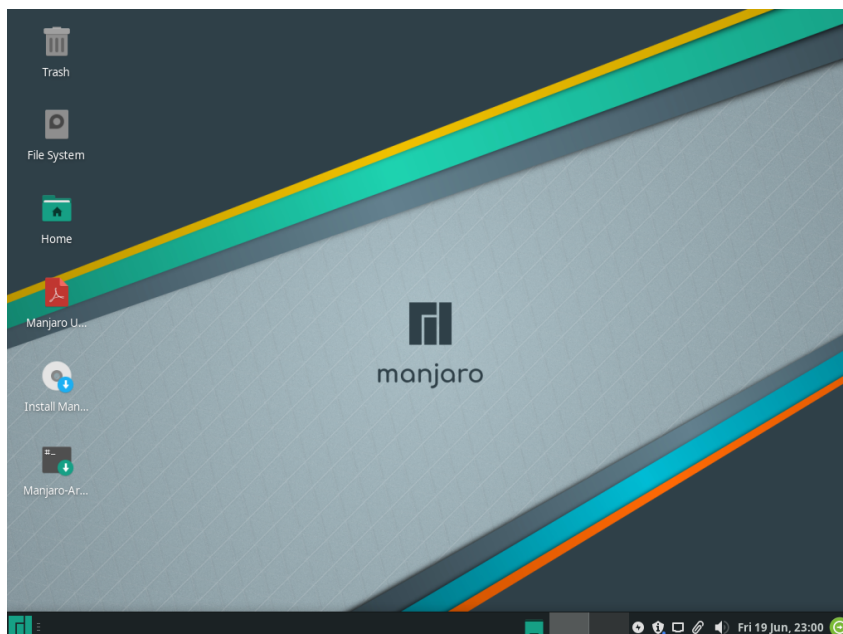
1. Wybór urządzenia do rozruchu.

Po włożeniu pendrive do komputera, aby go wybrać, trzeba tą maszynę uruchomić ponownie i podczas uruchamiania klikać przycisk F9, F10 lub F12 zależnie od producenta. Taką informację łatwo znaleźć szukając frazy "boot key [nazwa producenta lub model]". Gdy już uda nam się przejść do manu wyboru, wybieramy nasz pendrive i czekamy aż obraz się załaduje.



2. Live USB

W tym momencie jesteśmy w środku środowiska instalacyjnego. Możemy tu sprawdzić jak wizualnie będzie wyglądał system, jakie będziemy mieli opcje konfiguracji czy też jak nam się używa taki interfejs. Gdy już skończysz oględziny i uznamy, że chcemy kontynuować wybieramy instalator z pulpitu i przechodzimy dalej.



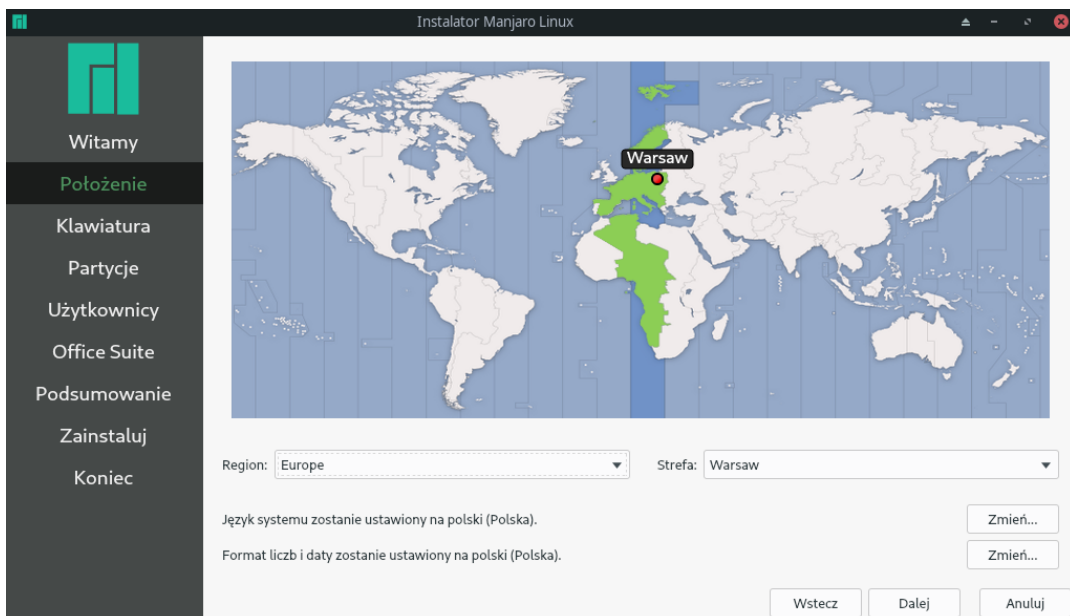
3. Instalator

Wiele dystrybucji posiada polskie instalatory co ułatwi proces osobą z mniejszym zrozumieniem języka angielskiego. Każdy też ma swoje specyficzne etapy, które charakteryzują różne dystrybucje.



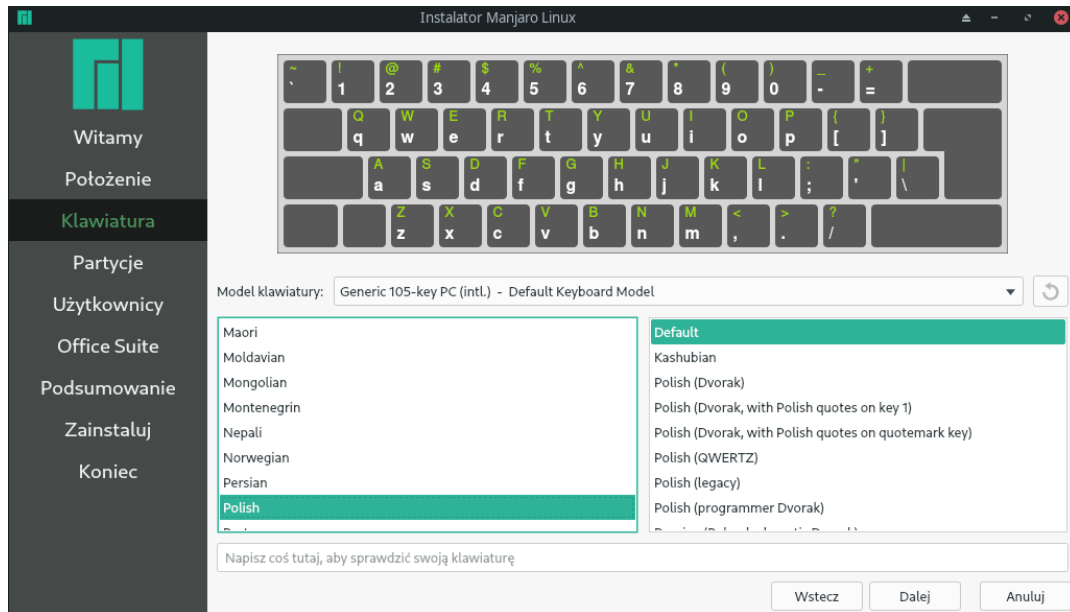
4. Wybór strefy czasowej

Aby upewnić się, że czas systemowy jest taki sam jak czas na twoim zegarku musimy ustawić strefę czasową dla servera NTP.



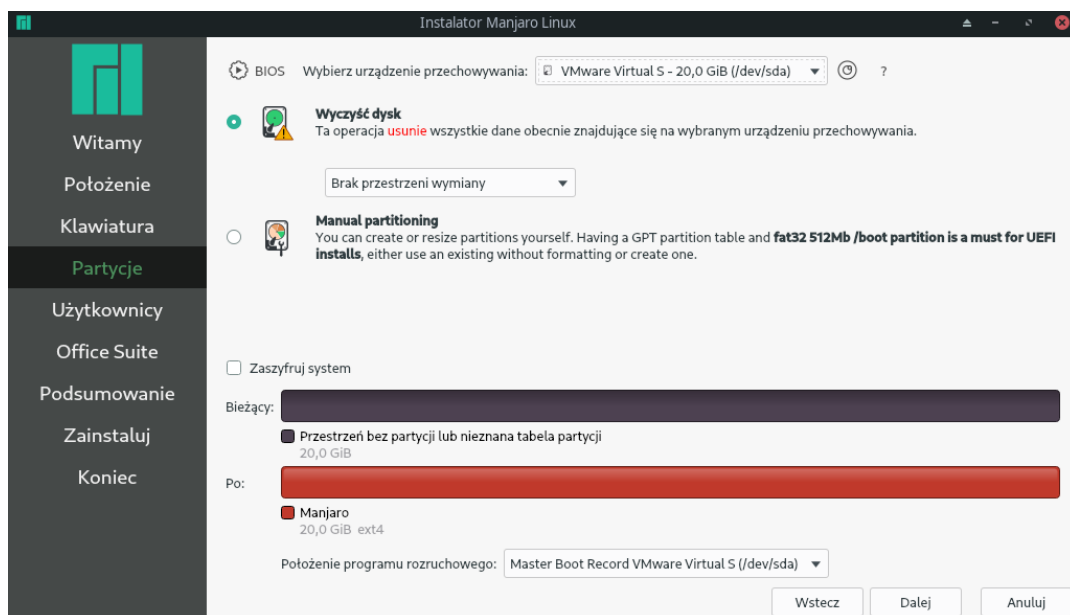
5. Klawiatura

Jeśli masz w imieniu lub po prostu zamierzasz użyć polskich znaków lub innych niedostępnych na standardowej klawiaturze angielskiej, wybierz układ odpowiedni do twoich potrzeb.



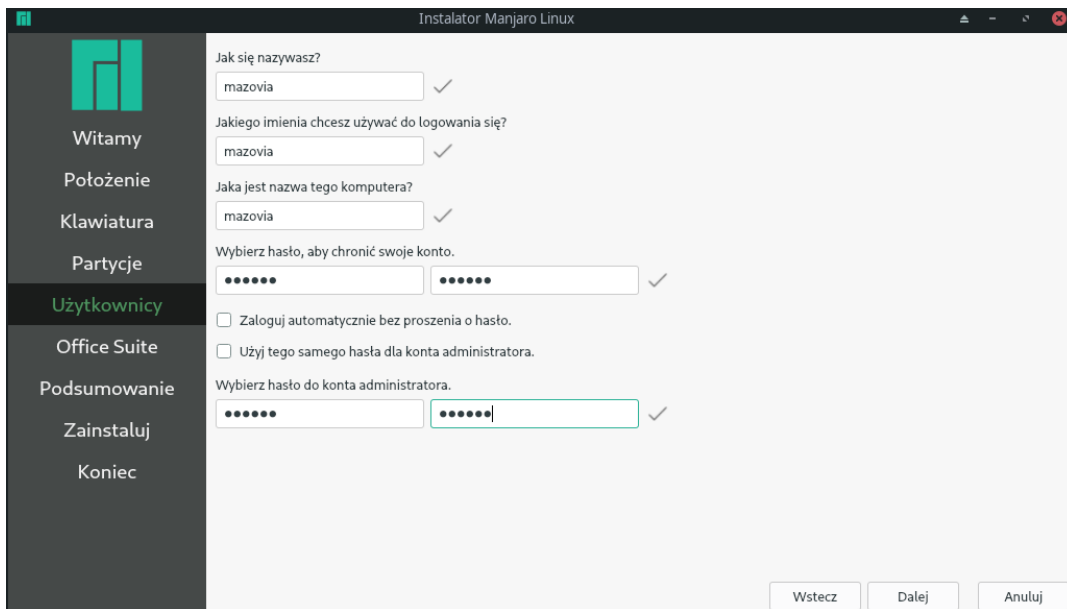
6. Partycjonowanie

Jest to najważniejszy etap instalacji. Tutaj czyścisz dysk aby zrobić miejsce na nowy system lub tworzysz odpowiednie środowisko z dostępnego miejsca. Dużo instalatorów posiada opcje domyślne, niewymagające własnego wkładu w partycjonowanie. Jeśli jednak nie masz takiej opcji poczytaj dokładnie poradniki do instalacji twojej dystrybucji.



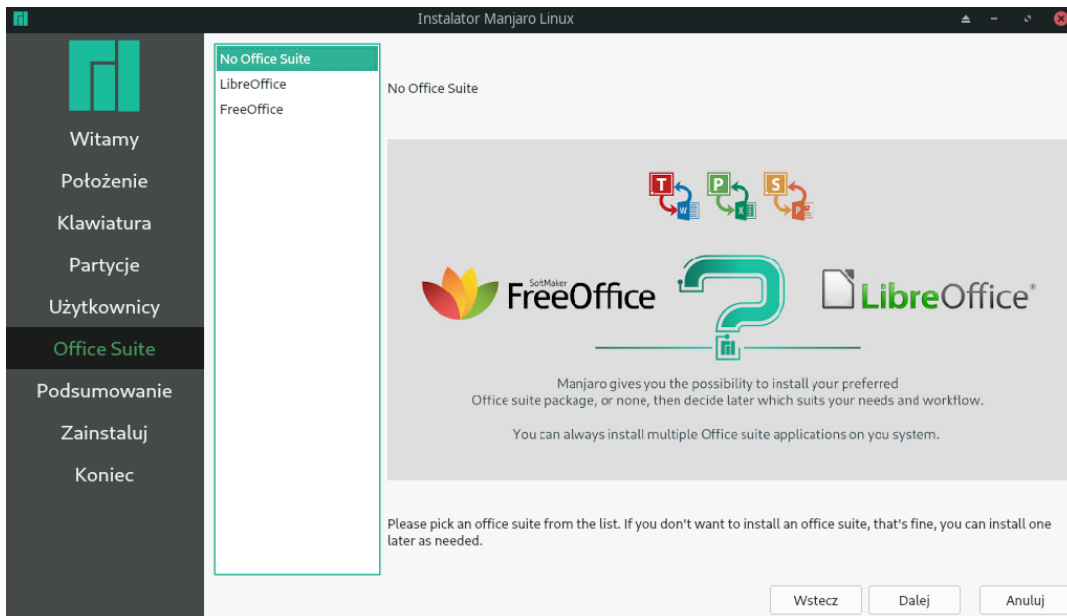
7. Tworzenie kont administratora i użytkownika

Aby skorzystać z systemu potrzebujemy konta umożliwiającego dostęp do niego. Podczas instalacji zawsze tworzymy konto administratora "root", które posiada wszelkie uprawnienia na maszynie. Stosunkowa większość dystrybucji wymaga utworzenia także konta upoważnionego, mającego dostęp do wykonywania operacji administratora ale posiadające inne hasło niż sam administrator. Pomaga to zwiększyć bezpieczeństwo.



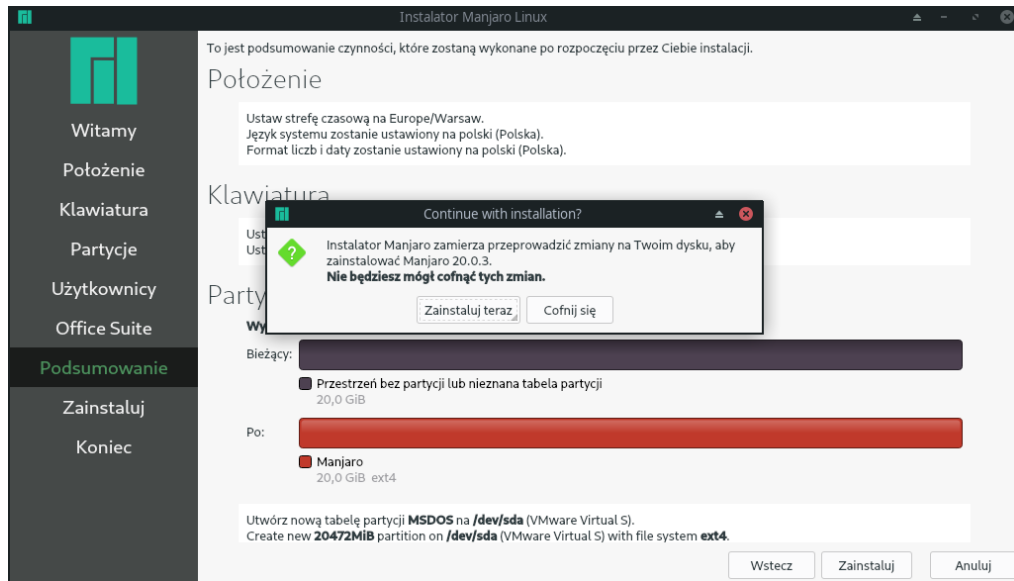
8. Dodatkowe opcje

Różne dystrybucje posiadają różne dodatkowe opcje poprawiające wrażenia zaraz po instalacji. Głównie dotyczą wyboru pakietów aplikacji oraz konfiguracji serwisów.



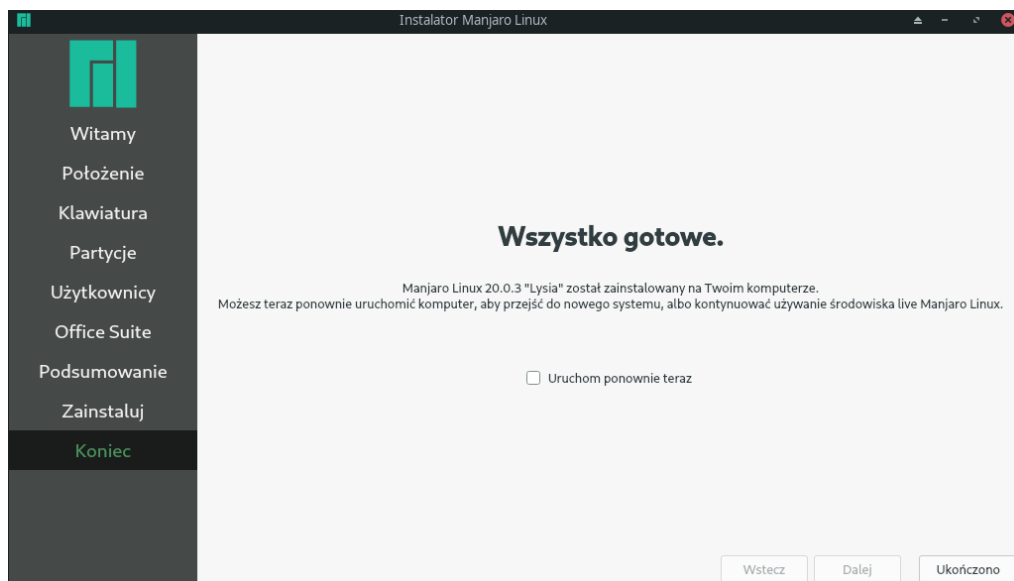
9. Podsumowanie

Tutaj upewnij się, że wszystkie elementy instalacji są takie same jak wybrany wcześniej. Po uruchomieniu instalatora nie będzie można dokonać żadnych zmian. Wiąże się to też z utratą danych na dysku więc upewnij się, że masz kopię zapasową plików ważnych dla ciebie. Jeśli jednak po instalacji okaże się, że wybrana opcja była inna niż zamierzana, a dotyczy ona samego systemu operacyjnego to po prostu uruchom instalator ponownie, tym razem wybierając odpowiednio.



10. Zakończenie

Gdy już udało ci się poprawnie zainstalować Linuxa musisz uruchomić komputer ponownie, żeby móc korzystać z nowego systemu. Pamiętaj, że przed uruchomieniem się komputera warto wyjąć pendriva. Ułatwia to maszynie rozpoznanie dysku zawierającego system.



5.1.3 Konfiguracja poinstalacyjna

Nie każda dystrybucja jest gotowa do użytku zaraz po zainstalowaniu, a nawet te które są wciąż posiadają swoje mankamenty i małe rzeczy poprawiające wrażenia. Aby znaleźć listę takich rzeczy po prostu wyszukaj "[twoja dystrybucja] post install". Jednak jeśli nie chcesz marnować czasu na szukanie informacji przedstawiam tutaj parę najważniejszych ustawień wartych uwagi.

1. Zaktualizuj system

Aby mieć pewność, że system jest w swojej najnowszej wersji warto zaraz po instalacji wykonać sprawdzenie aktualizacji przy pomocy wbudowanego programu. Możesz także wpisać komendy wykonujące wszystko za ciebie

- **Rodzina Debiana i Ubuntu** (menager pakietów APT) - `sudo apt update && sudo apt upgrade -y`
- **Arch oraz Manjaro** (menager pakietów Pacman) - `sudo pacman -Syu`

2. Zainstaluj sterowniki

Upewnij się, że wszystkie sterowniki są zainstalowane w swojej najnowszej wersji. Ich brak może wiązać się z dużymi problemami z konkretnym urządzeniem. Zapewnia to też najlepszą wydajność i stabilność.

3. Zainstaluj kodeki multimedialne

Aby móc odtwarzać najróżniejsze formaty wideo i audio warto zainstalować kodeki odpowiadające za ich przetwarzanie. Można tego dokonać tą komendą choć na twoim systemie może się różnić:

- `sudo apt-get install ubuntu-restricted-extras`

4. Zarządzanie baterią

Najważniejsza opcja dla użytkowników laptopów to zestaw narzędzi do zarządzania energią. Pakiet najbardziej rozbudowany i najczęściej używany to **TLP**. Pozwala znacząco poprawić żywotność laptopa oraz monitorować pobór mocy. Można go zainstalować komendą:

- `sudo apt install tlp tlp-rdw && sudo tlp start`

Aby zapoznać się z wszystkimi aspektami tego usprawnienia warto przeczytać dokumentację twórców:

<https://linrunner.de/tlp/introduction.html>

5. Zmniejszenie preferencji użycia pliku wymiany

Jeśli twoja maszyna ma powyżej 4 GB ramu to może się okazać, że system nie jest tak szybki jak mógłby być. Związane jest to z względnie agresywnym ustawieniem użycia pliku wymiany zamiast pamięci RAM. Aby zapewnić szybsze działanie wystarczy zmienić wartość tej preferencji w plikach konfiguracyjnych takimi komendami:

- `sudo sysctl vm.swappiness=10`

5.2 Nauka obsługi systemu oraz dobre nawyki

Przejście na inny system operacyjny może być wymagające nawet dla osób średnio zaawansowanych, dlatego ważnym jest, żeby umieć wyrobić sobie dobre nawyki podczas nauki oraz możliwie ułatwiać sobie tą naukę. Tu przedstawiam najprostrze techniki przyzwyczajania się do systemu:

1. Samemu szukaj rozwiązań

Jeśli natrafiasz na problem lub nie możesz czegoś znaleźć, spróbuj samemu rozwiązać problem. Rozpatrz jak mogło by to wyglądać na Windowsie i jakie było by tam rozwiązanie. Jeśli problem dotyczy obsługi terminalu możesz zagłębić się w opis używanego programu dzięki tym komendą:

- `man [twoja_komenda]` - uruchamia to instrukcję obsługi danej komendy razem z przykładami użycia
- `[twoja_komenda] -h` lub `[twoja_komenda] --help` - powoduje to wyświetlenie się instrukcji poprawnego zapisu oraz wyjaśnienia parametrów
- `[twoja_komenda] -d` lub `[twoja_komenda] --debug` - powoduje wyświetlenie wszystkich procesów oraz błędów podczas pracy programu. Umożliwia to szybsze znajdowanie problemów i sprawniejsze ich rozwiązywanie.

2. Korzystaj z społeczności

Nawet najlepsi potrzebują wsparcia innych. Jest to uwarunkowane obszernością zagadnień informatycznych, a co za tym idzie, brakiem zrozumienia wielu z nich. Rozwiązaniem do tego jest szeroka społeczność w , której każdy robi to co lubi najbardziej dzieląc się swoją wiedzą z innymi, samemu czerpiąc jeszcze więcej. Dlatego nie bój się zadawać pytań lub rozpoczynać dyskusji na portalach społeczności. Najpopularniejsze z nich to:

- <https://reddit.com/> - ważnymi pokojami dla początkujących i szukających pomocy mogą być:
 - <https://reddit.com/r/linux4noobs/>
 - <https://reddit.com/r/linux/>
 - <https://reddit.com/r/learnlinux/>
- <https://stackoverflow.com/> - największe forum technologiczno-programistyczne na świecie
- <https://stackexchange.com/>
- <https://quora.com/>

3. Pracuj systematycznie

Kluczem do nauki jest systematyczna praca. Bez niej praktycznie niemożliwym jest nabranie nawyków i przyzwyczajenie się do nowego narzędzia pracy.

Jeśli nie jest to twój jedyny komputer możesz mieć skłonności do unikania go z względu na wymagany większy wkład własny. Niestety tak często opóźniamy sobie postępy i utrudniamy cały proces.

Warto aby codziennie spędzić choć pół godziny przeglądając internet lub pisząc dokumenty właśnie na linuxie. Im częściej mamy z nim kontakt tym bardziej naturalny się wydaje, a wszystko staje się prostsze.

Rozdział 6: Aplikacje i środowiska alternatywne

6.1 Multimedia

- **Odtwarzacz wideo - VLC i SMPlayer**

Najpopularniejszymi opcjami są VLC oraz SMPlayer.

<https://www.videolan.org/vlc/#download>

<https://www.smplayer.info/pl/info>

- **Edytor graficzny - GIMP**

Jedyną z najbardziej rozbudowanych opcji do obróbki graficznej na linuxie pozostaje GIMP.

<https://www.gimp.org/>

- **Stylizacja zdjęć - RawTherapee**

Aby móc obrabiać i procesować surowe zdjęcia z aparatów fotograficznych zazwyczaj używa się Adobe Lightroom, jednak na Linuxie jak i innych platformach można wykorzystać jego alternatywę RawTherapee.

<https://rawtherapee.com/>

6.2 Twórczość

- **Program do rysowania - Krita**

Pomimo iż GIMP jest bardzo rozbudowanym i kompletnym rozwiązaniem to skupia się jednak na manipulacji gotowymi zdjęciami, a nie dostarcza uproszczeń dla osób tworzących cyfrowe rysunki. Wtedy warto zainteresować się Kritą, która rozwiązuje większość problemów GIMPa.

<https://krita.org/en/>

- **Tworzenie wektorów - Inkspace**

Do tworzenia grafik wektorowych, przystosowanych do druku i dużych formatów potrzebujemy narzędzia, które umożliwi nam taką twórczość. Tu pomaga nam Inkspace jak jeden z niewielu takich edytorów dostępnych za darmo.

<https://inkscape.org/>

- **Obróbka i nagrywanie wideo - OBS i DaVinci Resolve 16**

Zarówno na Windowsie jak i Linuxie OBS pozostaje jednym z najlepszych wyborów do nagrywania ekranu komputera oraz streamowania. Jednak nowością jest tu Resolve. Jest on o wiele lepiej zoptymalizowany i pozwala na płynniejszą obróbkę, niż konkurencja szczególnie, że zarówno Adobe jak i Sony wciąż są daleko w tyle z aplikacjami na Linuxa.

<https://obsproject.com/>

<https://www.blackmagicdesign.com/products/davinciresolve/>

6.3 Programowanie

- **Edytory kodu w terminalu - Vim/Nano**

Programy te umożliwiają szybką zmianę zawartości pliku poprzez terminal. Warto jest umieć korzystać chociaż z jednego z nich bo przyspiesza to rozwiązywanie problemów i konfigurację aplikacji.

- **Prosty edytor kodu - Atom**

Idealny wybór jeśli nie jesteś wymagający i piszesz małe programy, głównie skrypty systemowe lub proste programy.

<https://atom.io/>

- **Edytor kodu - Sublime Text**

Jedną z najlepszych alternatyw dla XCode na MacOS i VisualStudio na Windows jest Sublime Text. Dostarcza przejrzysty interfejs i wiele porządkanych funkcji jak podział ekranu na wiele zakładek czy tryb skupienia.

<https://www.sublimetext.com/>

- **Środowisko tworzenia aplikacji - Android Studio**

Jedyny faktyczny wybór gdy chodzi o tworzenie aplikacji mobilnych. Udostępnia emulator czystego androida oraz szeroko rozbudowane narzędzia do wyszukiwania błędów. W połączeniu z nową technologią Flutter pozwala tworzyć aplikacje kompatybilne zarówno z Androidem jak i IOSem.

<https://developer.android.com/studio>

- **Platforma konteneryzacji - Docker**

Rozpoczynając przygodę z tworzeniem produktów warto jest spróbować swoich sił w konteneryzacji. Gdy dobrze opanowana pozwala zmniejszyć potrzebny wkład pracy i zwiększyć wydajność pisanych przez siebie programów. Umożliwia także pełną automatyzację serwisów dzięki technologii klastrów.

<https://www.docker.com/>

Rozdział 7: Automatyzacja zadań

7.1 Skrypty

Najważniejszą gałąź automatyzacji dostępną dla zwykłego użytkownika to skrypty. Wraz z rozwojem interfejsów programowania aplikacji poszerzyły się zastosowania takich skryptów. Pierwotnie ograniczały się one do wykonywania żmudnych operacji systemowych w ułamkach sekund. Teraz można łączyć się z stronami internetowymi oraz wielkimi bazami danych aby osiągnąć duże rezultaty przy małym wkładzie pracy.

Pozwalają wizualizować aktualne dane firmowe z twoimi lokalnymi bazami lub konfigurować programy podczas ich instalowania. Umożliwiają wykonywanie powtarzalnych czynności przy pomocy jednej komendy.

7.1.1 Tworzenie skryptów

Skrypty w swojej idei są listą kolejnych komend wprowadzanych do terminalu w wydzielonym środowisku zwanym powłoką. Są one wykonywane po sobie, czasami prosząc użytkownika o potwierdzenie działania lub wybranie opcji z możliwych.

Dlatego też skrypty mają formę plików tekstowych uruchamianych w powłoce bash. Jest to osiągnięte dwoma metodami:

- Dodawanie do pliku rozszerzenia **.sh**, aby domyślnym programem wykonującym była powłoka. To samo odnosi się do wszystkich innych języków programowania jak Python czy PHP. Podobnymi rozszerzeniami są stosowane na windowsie **.bat** oraz **.ps1**.
- Dodawanie na początku skryptu tzw. 'shebang' określającego program interpretujący skrypt. Ma formę **# !/bin/bash**, gdzie znaki hasz i wykrzyknik są specyficznie rozpoznawane przez system na bazie Unixa. Podobnie jak w poprzednim przykładzie zmiana ścieżki na **/bin/bash/python** pozwoli interpretować skrypt przez pythona nawet bez odpowiedniego rozszerzenia.

Dobłą praktyką jest zawsze używać obu metod z racji tego, że się nie wykluczają i dokóci nie wejdiesz na bardziej zaawansowany poziom, oszczędzi ci to wiele problemów.

Zatem tworzenia skryptu zaczynamy od 3 najważniejszych kroków:

1. Ustawiamy lub tworzymy katalog roboczy - **mkdir skrypty && cd skrypty**
2. Tworzymy plik - **touch hello.sh**
3. Dodajemy możliwość uruchamiania pliku - **chmod +x hello.sh**
4. Uzupełniamy 'shebang' w pliku - **echo "# !/bin/bash" >> hello.sh**
5. Uruchamiamy plik w edytorze tekstu np. vim - **vi hello.sh**
6. Po ukończeniu pliku uruchamiamy go - **./hello.sh**

Przykładowe wykorzystanie zostanie pokazane na końcu działu.

7.2 Kontenery

Młoda gałąź technologii wirtualizacji pozwalająca tworzyć środowiska wraz z własnymi zasobami wewnątrz systemu operacyjnego. Umożliwia to prostą i szybką separację aplikacji oraz optymalizację ich działania.

Konteneryzacja zaczęła być popularna dopiero w 2013 roku gdy na rynku pojawił się docker lecz dopiero w ostatnich latach widzimy większe zainteresowanie tą technologią, co zawdzięczają rozwojowi systemów zarządzania nimi np. Kubernetes.

Wraz z klastrami umożliwiają tworzenie wydajnych i stabilnych infrastruktur, a także dostarczają automatyczną naprawę błędów dzięki klastrą fail-over.

Ciekawe zastosowanie, które udało mi się stworzyć łącząc prace paru twórców społeczności ułatwia proces tworzenia serwerów multimedialnych bazujących na **SteamCMD**. Pozwala on na uruchamianie takich serwerów w oddzielnych kontenerach przy minimalnym wkładzie użytkownika. Oto co trzeba zrobić:

1. Rozpocznij od stworzenia maszyny wirtualnej na swoim komputerze lub skorzystaj z darmowych środków na platformach

- <https://cloud.google.com/>
- <https://portal.azure.com/>
- <https://www.digitalocean.com/>.

2. Utwórz swój własny obraz Dockera. Pozwoli ci to dobrać wszystkie potrzebne parametry jak i przygotować jego środowisko. Na samym początku polecam odwiedzić bibliotekę społeczności docker.com gdzie znajdziesz wszelkie możliwe obrazy do każdego zastosowania. Samemu skorzystałem z obrazu <https://hub.docker.com/r/cm2network/steamcmd>, w którym zmieniłem parę opcji.

Obraz taki ma taką formę:

```
1 #####
2 # Dockerfile that contains SteamCMD
3 #####
4 FROM debian:buster-slim
5
6 LABEL maintainer="valentinlamonos@gmail.com"
7 ARG PUID=1000
8
9 ENV STEAMCMD /home/steam/steamcmd
10
11 # Install, update & upgrade packages
12 # Create user for the server
13 # This also creates the home directory we later need
14 # Clean TMP, apt-get cache and other stuff to make the image smaller
15 # Create Directory for SteamCMD
16 # Download SteamCMD
17 # Extract and delete archive
18 RUN set -x \
19     && apt-get update \
20     && apt-get install -y --no-install-recommends --no-install-suggests \
21         lib32stdc++6=8.3.0-6 \
22         lib32gcc1=1:8.3.0-6 \
23         wget=1.20.1-1.1 \
24         ca-certificates=20190110 \
25     && apt-get install -y wget \
26     && apt-get install -y curl \
27     && useradd -u $PUID -m steam \
28     && su steam -c \
29         "mkdir -p ${STEAMCMD} \
30         && cd ${STEAMCMD} \
31         && wget -qO- 'https://steamcdn-a.akamaihd.net/client/installer/steamcmd_linux.tar.gz' | tar xzf -" \
32     && apt-get remove --purge -y \
33         wget \
34     && apt-get clean autoclean \
35     && apt-get autoremove -y \
36     && rm -rf /var/lib/apt/lists/*
37
38 # Switch to user steam
39 USER steam
40
41 WORKDIR $STEAMCMD
42
43 VOLUME $STEAMCMD
```


Polecam zajrzenie do opisu twórcy obrazu, którym się wzorowałem gdyż dostarcza on wysoki poziom zrozumienia oraz wyjaśnia jak korzystać z samego dockera.

Aby uruchomić serwer w dockerze, w tym przypadku Do Not Starve Together, wystarczy wykonać te kolejne komendy po zainstalowaniu dockera na swojej maszynie:

```
docker pull yzere/steamcmd:latest
docker run -it -d --name=steamcmd yzere/steamcmd bash
docker exec -it steamcmd bash
/bin/bash ./steamcmd.sh +login anonymous +app_update 343050 +quit
curl -o ~/run_dedicated_server.sh https://github.com/yzere/steamcmd/raw/master/run_dedicated_server.sh
chmod u+x ~/run_dedicated_server.sh
mkdir ~/.klei/
mkdir ~/.klei/DoNotStarveTogether/
```

późniejsza konfiguracja wymaga już własnego wkładu w sposób ustawiania serwerów multimedialnych. Ważnym jest to, że wystarczy zmienić numer silnika instalowanego w obrazie Dockera, aby zmienić rodzaj instalowanej gry. Pełną listę można znaleźć na stronie https://developer.valvesoftware.com/wiki/Dedicated_Servers_List

Rozdział 8: Podsumowanie

Linux jest systemem uniwersalnym i sprawdza się w prawie każdym środowisku jeśli tylko mamy chęć i czas z niego skorzystać. Oferuje stabilność i zaufanie pośród wszystkich zaangażowanych w technologię. Jako najbrzydszy z trójki braci może nie być najpopularniejszy lub najpiękniejszy. Nie świadczy to o jego możliwościach i potędze. To właśnie dzięki standardyzacji systemów informatycznych Linux stał się najlepszą opcją do zarządzania serwerami, bazami danych, systemami bezpieczeństwa czy chmurą. Ułatwia życie deweloperom, udostępniając szerokie rozwiązania społeczności i upraszczając do nich dostęp.

W epoce informacji komputer stał się najpotężniejszym narzędziem, a system operacyjny pośrednikiem. Dobór tych dwóch części pozwala nam komunikować się z światem i tworzyć dla świata w sposób dla nas najwygodniejszy. Bo właśnie wygoda jest dziś najważniejsza. Wiąże się ona z oszczędnością czasu oraz wysiłku. Określa ona dostępność zasobów oraz stopień komplikacji ich uzyskiwania. Jest wyznacznikiem przejrzystości i zadowolenia z używanej maszyny. Więc odpowiedz sobie na pytanie: Co jest wygodne dla Ciebie?

Często nawet nie zdajemy sobie sprawy jak męczą nas pewne aspekty życia. Przyzwyczajamy się z nimi żyć, przyjmując ich wieczność. Lecz czasem trzeba się zatrzymać i zastanowić czy to co robię, jak to robię i czym to robię jest najlepszą opcją z wielu.

Wygoda płynąca z przyzwyczajenia do Windowsa, automatycznej aktualizacji systemu, bezproblemowego dostępu do gier, interfejsu z którym się 'urodziliśmy' wprowadza nas do strefu komfortu. Jednak czasem trzeba ją opuścić by znaleźć nowy kierunek.

Tym właśnie jest Linux. Nowym kierunkiem dla tych potrzebujących zmiany. Zmiany najpewniej na lepsze. Zmiany będącej początkiem pięknego związku obfitującego produktywnością i determinacją.

Dziękuję za lekturę dokumentacji i zapraszam do zapoznania się z programem warsztatów dołączonym do niej pod linkiem:

<https://github.com/yzere/linuxmigrationgithub/raw/master/warsztaty/warsztaty.pdf>

Rozdział 9: Źródła

Poniżej zawieram linki do moich materiałów jak i dodatkowych materiałów merytorycznych z których czerpałem wiadomości.

Moje materiały:

- Dokumentacja: <https://github.com/yzere/linuxmigrationgithub/raw/master/documentation.pdf>
- Repozytorium serwera SteamCMD: <https://github.com/yzere/steamcmd>
- Repozytorium dokumentacji w formacie .tex: <https://github.com/yzere/linuxmigrationgithub>
- Profil github: <https://github.com/yzere>
- Prezentacja projektu w formie powerpoint: https://tm1edupl-my.sharepoint.com/:p:/g/personal/maciej_tracz_tm1_edu_pl/EQ6SGFr0i6tBjp_otvNkXncBRsYdWYDT-TY8BZ5NnpZtNg?e=pUWvCA
- Prezentacja projektu w formie strony internetowej aplikacji Sway: <https://sway.office.com/EK7qb1KV4ngwyVFe?ref=Link>

Materiały zewnętrzne wykorzystane w dokumentacji lub służące do poszerzenia zrozumienia tematu:

- Ściągi do pisania skryptów: <https://devhints.io/bash>
- Obraz docker użyty w repozytorium SteamCMD: <https://hub.docker.com/r/cm2network/steamcmd>
- Informacje zawarte w otwartej encyklopedii internetowej Wikipedia: <https://en.wikipedia.org/wiki/Linux> oraz wszystkie pochodne tego artykułu.
- Pozostałe źródła merytoryczne obejmujące blogi, artykuły społeczności, dokumentacja oraz prace naukowe: <https://docs.google.com/document/d/1EWPshBZEQyMFVMd14gyNCKK1Qdd0BhUnswrH6BAws68/edit?usp=sharing>