Západočeská univerzita v Plzni

Fakulta aplikovaných věd

Katedra informatiky a výpočetní techniky

Diplomová práce

Automatická síťová instalace operačních systémů v prostředí KIV

Plzeň, 2019 Petr Černohouz

Místo této stránky bude vloženo zadání

Prohlašuji, že jsem diplomovou práci vypracoval samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne 27.6.2019 Petr Černohouz

Automatic network installation of operating systems at KIV

The work focuses on finding a suitable solution for automated installation of computers. The analytical part is devoted to the research of available solutions for the installation of various operating systems with focus on MS Windows and GNU/Linux. Both individual methods and ready-made solutions offering the possibility of mass installations were examined.

The implementation part of the thesis focuses on creating a solution for the complete life cycle of workstation installation. An important part is the creation of a web interface for managing and monitoring the installation process, which helps to operate the entire solution. In the conclusion is a description and evaluation of detailed testing in real environment of laboratories of the Department of Informatics and Computer Science.

Automatická síťová instalace operačních systémů v prostředí KIV

Práce se zaměřuje na nalezení vhodného řešení automatizované instalace počítačů. Analytická část se věnuje průzkumu dostupných řešení pro instalaci různých operačních systémů s důrazem kladeným na systémy MS Windows a GNU/Linux. Zkoumány byly jak jednotlivé metody, tak hotová řešení nabízející možnost hromadných instalací.

Realizační část práce se zaměřuje na tvorbu řešení obsluhujícího kompletní životní cyklus instalace pracovních stanic. Důležitou součástí je tvorba webového rozhraní pro správu a monitoring instalačního procesu, které pomáhá obsluhovat celé řešení. V závěru je popis a zhodnocení detailního testování v reálném prostředí laboratoří katedry informatiky a výpočetní techniky.

Obsah

1 Úvod

V organizacích s velkým počtem počítačů vzniká potřeba systémového řešení jejich životního cyklu. Nedílnou součástí životního cyklu je instalace počítačů. Instalace je významným prvkem zvláště u veřejně přístupných počítačů, jaké se vyskytují například v knihovnách, školících centrech nebo na univerzitách. Důvodem je častější potřeba reinstalace těchto počítačů ať už na nový akademický rok nebo nový kurz.

Proces instalace většího množství počítačů je však časově náročnou operací a je komplikován také potřebou relativně častého opakování instalací pro nasazení aktualizací nebo pro opravy chyb. Chyby mohou vznikat i při samotném procesu instalace v případě, kdy je prováděn kompletně ručně, nebo obsahuje manuální kroky.

Ideálním řešením výše zmíněných problémů by byl proces automatické instalace. Ten by pomohl předcházet chybám vznikajících při manuální instalaci. S automatickou instalací lze také opravovat stroje se softwarovými problémy v podstatě okamžitě po nahlášení problému.

Vlastní automatickou instalaci je vhodné doplnit o systém pro správu a provádění instalací. Administrátoři tak získají okamžitý přehled o stavu počítačů, průběhu prováděných operací a budou schopni instalace vzdálené spouštět.

Téma práce vzniklo na základě situace na katedře informatiky a výpočetní techniky (KIV), která zde panuje v souvislosti se správou počítačových laboratoří. V laboratořích je dohromady několik desítek strojů k dispozici studentům pro výuku a vypracovávání semestrálních prací, kterým musí být věnována pozornost. Dle využití laboratoří se liší požadavky na dostupný software, konfiguraci systémů a mohou se také lišit oprávnění dostupná uživatelům.

V práci bude zkoumána použitelnost již existujících nástrojů i kompletních řešení. Vybraná řešení nebo nástroje budou nasazeny a přizpůsobeny pro prostředí KIV. Případné chybějící části budou v rámci této práce vyvinuty. Výsledkem by měl být systém řešící situaci na KIV a usnadňující práci administrátorům.

2 Instalace počítačů

V následujících kapitolách budou prozkoumány a zhodnoceny existující nástroje a možnosti pro instalaci počítačů s přihlédnutím ke specifickým potřebám hromadné instalace.

2.1 Způsoby instalace počítačů

Pro přípravu počítačů máme k dispozici manuální a automatickou instalaci. Metody se liší mírou nutných příprav, úrovní automatizace i možnostmi přizpůsobení dané instalace. Zvláštním druhem přípravy počítače je klonování připraveného obrazu disku. Metody přípravy detailně popisují následující kapitoly

2.1.1 Manuální instalace

Jednoduchý způsob instalace, při kterém se typicky používá instalátor operačního systému. Instalátor může být dle druhu operačního systému textový, textově/grafický nebo grafický. Administrátor v tomto případě prochází jednotlivé kroky instalace od nastavení sítě přes rozdělení disku a výběr instalovaných balíčků až po tvorbu uživatelů a konfigurace jména počítače.

Během instalace může administrátor vybírat volby přesně na míru instalovanému počítači a instalovat různorodou skupinu počítačů ze stejného instalačního zdroje. Tato volnost však může být jedním z rizikových faktorů a zdrojů chyb během instalace.

Manuální instalace je časově náročný proces. V počáteční a koncové fázi vyžaduje přítomnost administrátora a velmi špatně se škáluje. Pro urychlení je možné instalovat více počítačů současně, ale v tomto případě je zde vyšší riziko možných chyb způsobených přecházením mezi počítači. Ve výsledku může snaha o urychlení instalace vést k delšímu výslednému času, než při sekvenčním průběhu z důvodu nutnosti opravy chyb v instalaci. Možnosti urychlení jsou závislé na poměru času, ve kterém je nutná přítomnost administrátora oproti času, kdy instalace běží bez nutnosti interakce, například při kopírování souborů se administrátor může věnovat jinému počítači.

Během instalace může být nutné pro některé počítače přidat ovladače atypického hardware jako jsou řadiče disku nebo síťové karty. Po instalaci samotného operačního systému se pokračuje instalací potřebného softwarového vybavení. V prostředí

organizací se počítače následně připojují do domény Active Directory [1], aplikují se skupinové politiky a následují další akce, aby byly splněny interní IT standardy. Dodatečné kroky mohou být dalším zdrojem chyb a komplikací během instalace a hrozí zde také riziko zapomenutí některého kroku nebo nedodržení požadovaného pořadí kroků.

Manuální instalace nepotřebuje žádné speciální prerekvizity, stačí mít k dispozici pouze instalační soubory. Je však kvůli potřebě manuálních zásahů časově náročná a je zde velké riziko lidské chyby.

2.1.2 Klonování obrazu disku

Časovou náročnost a náchylnost na chyby manuální instalace může vyřešit instalace strojů nasazení hotových obrazů disku. Obrazy by měly být připravené ve formě umožňující následné přizpůsobení cílovému počítači, abychom předešli stavu, ve kterém budeme mít po nasazení větší množství počítačů se shodným jménem a/nebo IP adresou.

Tvorba obrazu probíhá na referenčním počítači, kde se provede manuální instalace včetně všech nastavení systému a instalace požadovaného software jak bylo popsáno v kapitole 2.1.1 Následně je pomocí vhodného nástroje, jako je například program Clonezila [2] nebo Acronis True Image[3], vytvořen obraz disku pro další distribuci.

Při více typech počítačů nebo požadavku na různé konfigurace je možné vytvořit několik různých obrazů pro různé kombinace, nebo provést přizpůsobení již nasazených obrazů. U přizpůsobení se může jednat o manuální úpravy nebo spuštění připravených skriptů pro úpravu.

Doba klonování je závislá na velikosti klonovaného obrazu, rychlosti disku a při přenosu po síti i rychlosti sítě. Dobu může prodloužit zvolený způsob a množství úloh pro přizpůsobení obrazu pro konkrétní počítač nebo použití. Při větším množství variant také rostou nároky na kapacitu úložiště obrazů. I když se provede optimalizace a obraz zabírá pouze velikost odpovídající zaplnění disku, jedná o několik jednotek nebo desítek gigabytů na obraz. U obrazů komplexních systému, nebo bez patřičné optimalizace může velikost překročit hranici terabytu.

Dnes běžné síťové instalace pracují s rychlostí 1 Gb/s, což znamená maximální teoretickou rychlost přenosu 125 MB/s. To je nižší rychlost, než se kterou pracují běžně používané pevné disky, proto budu uvažovat síť jako úzké hrdlo celého procesu. V tomto případě by přenos obrazu o velikosti 1 TB trval v teoretickém ideálním stavu 2 hodiny a 20 minut. U obrazu velikosti 50 GB se pak dá dostat na teoretických 7 minut. Je však potřeba brát v úvahu, že se jedná jen o teoretická maxima, která jsou v reálném prostředí velmi těžko dosažitelná, a potřebné časy tak budou pravděpodobně delší.

Při potřebě aktualizací obrazu dochází k jeho spuštění na referenčním počítači, provedení kroků nutných k aktualizaci a opětovné vytvoření upraveného obrazu. Tato operace je stejně jako příprava původního obrazů manuální činností prováděnou administrátory.

2.1.3 Automatická instalace

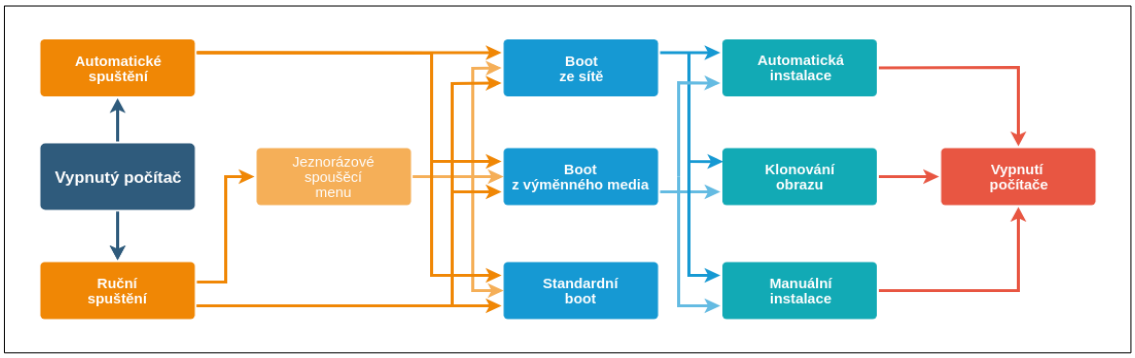
Automatická instalace umožňuje odstranit manuální kroky a zároveň nemá tak vysoké nároky na úložnou kapacitu jako klonování disku. Automatická instalace spočívá ve spuštění instalačního programu, který není obsluhován administrátorem, ale řídí se profilem, podle kterého prochází jednotlivé kroky. Proces může následně pokračovat v konfiguraci cílového počítače a instalaci dalšího softwarového vybavení.

Takto řízený proces může vyžadovat například jen start počítače a výběr vhodného profilu instalace, nebo může být i zcela automatizovaný a o výsledné podobě instalace se rozhoduje na základě předchozího nastavení nebo vlastností cílového počítače, jako jsou hardwarová adresa síťové karty, jméno stoje, model apod.

V průběhu automatické instalace dochází pokaždé k nasazení konkrétní verze systému včetně bezpečnostních oprav a nasazení požadovaných verzí softwarového vybavení. Příprava na tento typ instalace spočívá ve vytvoření instalačního profilu, kterým se následně řídí příslušné instalátory. Pokud pro instalaci doplňkového softwarového vybavení nejsou použity systémové balíčky, je třeba ještě dodat odkaz na umístění instalátorů. Výsledné profily následně obsahují jen posloupnost úkolů a jejich parametrů a jsou velmi malé. Není tak problémem mít k dispozici desítky různých instalačních profilů. Z uvedených metod se jako nejvhodnější pro další postup jeví metoda automatická instalace, která řeší nedostatky manuální instalace i klonování obrazu disku.

2.2 Kroky instalačního procesu

Instalace začíná spuštěním počítače, na kterém budeme provádět instalaci. Po spuštění počítače se načte BIOS, který následně hledá zavaděč operačního systému na zařízeních dle jejich pořadí ve spouštěcí posloupnosti. Při nalezení prvního zařízení se zavaděčem se mu předá řízení a postupuje se dle jeho konfigurace. Zavaděč může zobrazit možnosti spouštění počítače nebo spustit instalaci. Následně je provedena samotná instalace podle jednoho z postupů v kapitole 2.1. Součástí instalace může být i spuštění nástrojů pro přizpůsobení příslušného počítače a instalaci potřebného softwarového vybavení, pokud je to potřeba. Schéma životního cyklu instalace je zobrazeno na Obr 2.1.



Obr 2.1: Životní cyklus instalace

Po provedení instalace bude již počítač nabíhat do nově nainstalovaného systému a nesmí docházet k opětovnému spuštění instalátoru. Cesty k dosažení tohoto cíle jsou závislé na nastavení BIOSu a budou blíže popsány v kapitole 2.4

2.3 Zapnutí počítače

Vyjma obyčejného zapnutí počítače stiskem napájecího tlačítka, které běžně vyžaduje fyzickou přítomnost můžeme u většiny dnes dostupných počítačů využít i vzdálený start. Možnosti a omezení těchto metod se liší dle použitého protokolu a budou teď rozebrány.

2.3.1 Probuzení po síti

Pro probuzení po síti se využívá technologie Wake-on-Lan – WoL která vznikla na základě white paperu s názvem Magic packet technology od AMD [4] a umožňuje zapnutí počítače zasláním speciálně sestaveného Magic paketu obsahujícím

ethernetovou adresu spouštěného počítače. Použití ethernetové adresy limituje WoL na jednu broadcastovou doménu sítě.

Při potřebě spouštět počítače ve více podsítích je jednou z možností řešení omezení protokolu připojení počítače provádějícího spouštění do všech požadovaných sítí. Druhou možností nasadit agenty na počítače běžící v daných sítích. Mělo by se jednat o servisní počítač (může být i virtuální), který bude v požadované síti a bude neustále zapnutý, aby mohl spouštět ostatní počítače. Pro servisní počítač na platformě MS Windows existuje nástroj WOL Agent [5] pro GNU/Linux se mi nepodařilo najít hotové řešení, ale principiálně nejde o složitý software a mohl by být v případě potřeby vytvořen.

Magic paket začíná šesti byty obsahující `FFh` následovaných šestnáctkrát opakovanou hardwarovou adresou cílového počítače. Na konci paketu může být ještě 6 bytů pro uložení hexadecimálního hesla. Celková délka paketu může být až 108 bytů v závislosti na použití a délce hesla. Struktura paketu je zobrazena na Obr 2.2 Funkce je k nalezení v některých BIOSech pod názvem SecureOn, není však běžně podporována a vzhledem k nízkému riziku pramenícím ze zneužití WoL funkcionality by mohlo použití této funkce zbytečně omezit využitelnost této technologie.

Obr 2.2: Magic paket

Pro generování paketů v GNU/Linuxu existují nástroje `etherwake` [6], který vyžaduje superuživatelská práva nebo `wakeonlan` [7]. Oba nástroje jsou součástí všech běžných linuxových distribucí. Pro MS Windows existují obdobné nástroje, například WakeMeOnLan [8] nebo WakoOnLanX [9]

2.3.2 Intelligent Platform Management Interface

Intelligent Platform Management Interface – IPMI je software na ovládání počítačů, hlavně serverové infrastruktury, představen společnosti intel v roce 1998 [10]. Postupně začaly tuto funkcionalitu podporovat i další výrobci a dnes je dostupná

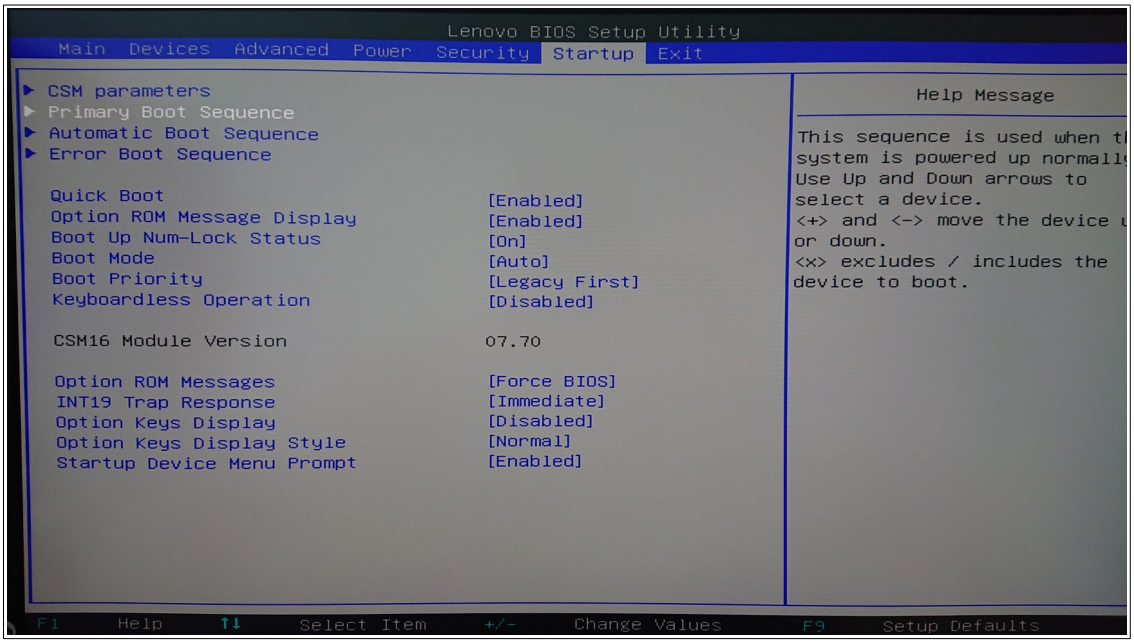
u všech významných značek. IPMI umožňuje přistupovat k počítači i ve stavu, kdy neběží operační systém a také v případě vypnutí počítače, pokud je stále připojen ke zdroji napájení a do počítačové sítě.

IPMI nabízí široké možnosti správy a monitoringu počítače, včetně vzdáleného přístupu na konzoli systému. IPMI lze zpřístupnit přes dedikovaný síťový port, nebo může sdílet běžný síťový port počítače. V případě sdílení je na síťovém portu krom IP adresy počítače i IP adresa IPMI rozhraní. U sdíleného portu nemusí být některé pokročilé funkce dostupné.

V současné době je IPMI k dispozici převážně v serverech, ale je možné se s ním setkat i ve výkonných pracovních stanicích. Potřebuje však vlastní IP adresu, a proto není IPMI v běžných instalacích příliš využívané. Při použití je potřeba věnovat důraz zabezpečení přístupu na IPMI. Možnosti zabezpečení protokolu vylepšuje specifikace IPMI v2.0 [11]

2.4 Výběr zdroje zavaděče operačního systému

V BIOSu počítače je možné nastavit, v jakém pořadí se prohledávají zařízení, na kterých se hledá zavaděč systému. Kromě běžného pořadí spouštění, které se provádí při každém startu při spuštění počítače zapínacím tlačítkem, můžeme zvolit jiné pořadí pro případy automatického zapnutí, například pomocí WoL jak je zobrazeno na Obr 2.3. Pro servisní zásahy bývá k dispozici i jednorázové zobrazení menu pro výběr zařízení se zavaděčem dostupné stiskem klávesy.



Obr 2.3: Výběr zdroje zavaděče v BIOSu

U běžného pořadí spouštění je vhodné dávat na první místo pevný disk počítače, abychom měli jistotu, že se nám vždy spustí požadovaný systém. Při umístění výměnných medií nebo sítě před pevný disk může docházet k nechtěným startům do jiné konfigurace, například při ponechaném připojeném USB disku se zavaděčem.

Pro využitý výměnných médií jako zdroje pro získání zavaděče je výhodnější použít vyvolání jednorázového boot menu během startu systému. Slouží k tomu typicky jedna F kláves, často F10 nebo F12. Toto jednorázové menu může být v biosu zcela vypnuto, nebo může být chráněno heslem zabraňujícím neoprávněné manipulaci s počítačem.

Pro případy automatického startu, například po probuzení ze sítě, se dá nadefinovat odlišné pořadí prohledávání zařízení. Zde můžeme zvolit jako výchozí metodu boot ze sítě a zpřístupnit tak instalátor nebo servisní menu zavaděče pouze při automatickém spuštění.

Vzhledem k provádění úprav přímo v biosu jsou možnosti vzdálené změny velmi limitované a jsou možné pouze v případě, že počítač disponuje management rozhraní jako je IPMI popsané v 2.3.2 nebo například iDRAC [12].

2.5 Zavaděč operačního systému

Na základě nastaveného pořadí spouštění popsaného v předchozí kapitole 2.4 dojde při nalezení zavaděče operačního systému k jeho spuštění. Zavaděč je odpovědný za nahrání jádra operačního systému, případně jádra instalačního programu. Zavaděč může obsahovat menu s výběrem možností spouštění. Toto menu však nemusí být zobrazováno, například u počítačů s jedním operačním systémem. Případné servisní volby se mohou zobrazit po stisknutí předdefinované klávesy během startu počítače. Jednotlivé druhy zavaděčů s konkrétními příklady implementací popisují následující kapitoly.

2.5.1 Lokální zavaděč

Lokální zavaděč se nachází na začátku pevného disku počítače a slouží k zavedení nainstalovaných operačních systémů. V našem scénáři se uplatní v případě standardního bootu již nainstalovaného sytému. Ačkoliv se jedná o lokální zavaděč, může být samotný systém stahován ze sítě. Možnosti záleží na konkrétním zavaděči. Dva hlavní zavaděče si nyní představíme.

2.5.1.1 GNU GRUB

GNU GRUB začal vznikat v roce 1995 jako nástroj na spouštění systému GNU Hurd [13] a širším cílem bylo vytvořit modulární zavaděč. Na základě této práce vznikl následně v roce 2002 GRUB 2 řešící problémy s modularitou a rozšiřitelností původního zavaděče.

GRUB je multiboot zavaděč využíván většinou linuxových distribucí, který podporuje i zavádění MS Windows a systému na bázi BSD. Primárně slouží jako zavaděč uložený na lokálním disku počítače, ale poskytuje i základní podporu pro využití jako síťový zavaděč. V roli síťového zavaděče však je k dispozici minimum možností nastavení, tudíž ho budeme využívat jen jako lokální zavaděč. Je však možné spustit zavaděč lokálně a následně stáhnout příslušné jádro a další potřebné soubory po síti. Tato možnost je zajímavá pro situace, kdy můžeme mít na síťovém úložišti připravené servisní obrazy, které nebudou zabírat místo na lokálním disku.

Konfigurace zavaděče může být pevná s vybranými volbami, které jsou na výběr, nebo může být po instalaci vygenerována dle konfigurace daného počítače, rozdělení

disků, nebo například přítomnosti konkrétních souborů případně zařízení. Jednotlivé položky nebo celém menu je možné chránit pomocí hesla a zabránit tak neautorizovanému spuštění servisních voleb. Pro ladění nastavení je také k dispozici minimalistická příkazová řádka. Ta však může být i bezpečnostním rizikem, protože je možné skrz ni spustit GNU/Linux v jednouživatelském režimu a získat tak plný přístup do systému. Přístup k ní by měl být tedy na běžných počítačích zakázán nebo chráněn heslem.

2.5.1.2 Windows boot manager

Windows boot mnager slouží k zavedení operačních systémů. Je standardní součástí instalace MS Windows. Není však omezen pouze na tento systém, ale umožňuje zavádění i dalších systémů, například GNU/Linuxu. Konfigurace se nachází v úložišti Boot Configuration Data – BCD [14]. Pro úpravy konfigurace slouží nástroj BCDEdit [14].

Samotný zavaděč je umístěn na systémovém disku s MS Windows a při běžných instalacích s ním uživatel nepřijde do styku, protože jako výchozí volba je běžný start systému a je potlačeno zobrazování obrazovky zavaděče. Toto chování je možno změnit editací parametrů v BCD. Obdobně je možné přidat zde další operační systémy do nabídky.

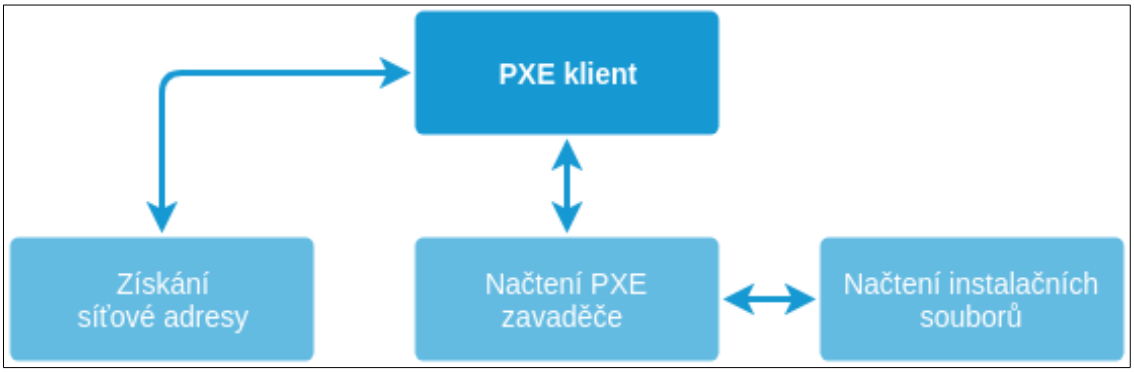
Celkově nabízí Windows boot manager menší možnosti konfigurace než GNU GRUB a proto je vhodné oba zavaděče kombinovat. Jako primární zavaděč je spouštěn GNU GRUB, který při požadavku na spuštění MS Windows předá řízení zavaděči Windows boot manager. Při výchozím nastavení se skrýváním nabídky tento krok běžně uživatel ani nepostřehne a není tudíž rušivý.

2.5.2 Zavaděč pro přenosná média

Pro systémy na přenosným mediích, jako jsou instalační nebo záchranná CD/DVD nebo také přenosné USB disky je také potřeba zavaděč. Tento zavaděč má svá specifika a je umístěn na daném mediu. Jako příklad může být zmíněn zavaděč ISOLINUX [15] ze sady nástrojů Syslinux [16], který umožňuje přípravu zavaděče pro ISO obrazy i USB disky. Výměnná media jsou však mimo rozsah této práce, a proto nebude dále rozebírán.

2.5.3 Síťový zavaděč

Síťový zavaděč může být zaveden ze síťového úložiště a umožňuje nám spouštět systémy nebo instalátory po síti. Standardem v oblasti síťového bootu je technologie Preboot eXecution Environment – PXE [17]. Tato technologie do společnosti Intel nahrazuje a sjednocuje dříve používaná řešení . Technologii PXE musí podporovat počítač. Proces načtení zavaděče je znázorněn na Obr 2.4



Obr 2.4: Schéma načítání PXE zavaděče a následně instalátoru

Aby bylo možné načíst vlastní zavaděč, je potřeba nejprve získat síťovou adresu, následně dojde k načtení samotného zavaděče a podle jeho konfigurace k načtení instalačních souborů nebo navázaných zavaděčů. Potřebné nástroje a používané protokoly budou detailně rozebrány v kapitole 2.5.4. Nyní se podíváme na přípravu samotného zavaděče a jeho konfigurace. Jako síťový zavaděč je možné použít například GNU GRUB popsaný v kapitole 2.5.1.1. Ten ovšem nabízí jen minimální možnosti konfigurace a její přizpůsobení jednotlivým koncovým počítačům. Další možností je nástroj PXELINUX [18] ze sady nástrojů Syslinux, který si popíšeme podrobněji.

2.5.3.1 PXELINUX

Jedná se kompletní zavaděč s podporou textového i grafického rozhraní. Je k dispozici ve dvou verzích, základní PXE podporuje pouze jednoduchý přenosový protokol pro získání instalátorů, pokročilejší gPXE/iPXE zavaděč podporuje i další protokoly popsané v kapitole 2.5.4, ovšem nemusí být podporován na všech počítačích.

Vlastní zavaděč a konfigurační soubory jsou umístěny na síťovém serveru. Adresu tohoto serveru počítač získá společně s vlastní síťovou adresou v prvním kroku.

Možnosti získání adresy jsou popsány v kapitolách 2.5.4.1 a 2.5.4.2. Jako síťový server může sloužit například i router, který podporuje potřebné protokoly a možnosti nastavení.

Po načtení zavaděče probíhá hledání konfiguračního souboru na stejném síťovém serveru, ze kterého byl získán zavaděč. Samotné hledání vhodného konfiguračního souboru probíhá v několika krocích a platí, že po nalezení shody se dále v hledání nepokračuje. V prvním kroku se vyhledává soubor, jehož název je shodný s UUID počítače zapsaného v šestnáctkové soustavě malými písmeny. Následuje hardwarová adresa daného počítače zapsaná v šestnáctkové soustavě malými písmeny a jako oddělovač bytů slouží pomlčka. Další je série kroků, kdy dojde k převodu IP adresy na šestnáctkovou soustavu, tentokrát však s velkými písmeny a hledá se nejdelší shoda, tzn. postupně se ukrajuje znak po znaku. Jako poslední možnost je soubor s názvem `default`. Konfigurační soubor může obsahovat vícepoložkové boot menu s podporou zanořování nebo zde může být jen jedna volba, která ani nezobrazí menu a systém s ní okamžitě naběhne.

Obecný konfigurační soubor by mohl jako výchozí volbu obsahovat předání řízení lokálnímu zavaděči pro případy, kdy došlo k neúmyslnému načtení síťového zavaděče. Další mohou být servisní volby, použitelné pro diagnostiku a opravu počítačů. Pro samotnou instalaci je možné využít speciální konfigurační soubory, které spustí rovnou požadovanou instalaci. Dá se využít pojmenování souborů dle hardwarové adresy. Nemusí se jednat jen o fyzické soubory, ale jsou podporovány i symbolické odkazy. Zde je však podmínka, že musí být relativní vzhledem k síťové cestě k zavaděči.

2.5.4 Nástroje a protokoly pro načtení síťového zavaděče

Síťový zavaděč ke své funkci potřebuje podporu na straně počítače a zároveň také podporu dalších síťových protokolů. Popis relevantních protokolů pro získání síťové adresy obsahují první dvě podkapitoly 2.5.4.1 a 2.5.4.2. Protokoly pro přenos souborů zavaděče a protokoly pro načtení instalačních souborů jsou popsány v podkapitolách 2.5.4.3, 2.5.4.4 a 2.5.4.5.

2.5.4.1 Bootstrap protocol

Pro možnost konfigurace síťových parametrů byl v roce 1985 uveden bootstrap protocol - BOOTP. Protokol je definován v RFC 951 [19]. Umožňuje tvorbu profilu pro počítače, kdy součástí profilu je IP adresa počítače, maska, adresa brány a adresa DNS serverů. Krom těchto údajů je možné poslat i adresu serveru se zavaděčem pro podporu startu ze sítě.

Možnost startu ze sítě byla důležitá hlavně pro bezdiskové stanice, které byly schopny v jedné zprávě od BOOTP serveru získat veškeré informace k tomu, aby mohly ze síťového úložiště stáhnout obraz disku a spustit systém.

Nevýhodou protokolu je nutnost tvořit profily pro jednotlivé počítače. Každý počítač tak měl pevně přidělenou IP adresu ve svém profilu a tato adresa nemohla být použita jiným počítačem ani v případě, kdy nebyla využívána. Při změnách v síti nebo přidávání počítačů je tak nutné vždy zasáhnout do statické konfigurace uložené na BOOTP serveru.

2.5.4.2 Dynamic Host Configuration Protocol

Dynamic Host Configuration Protocol - DHCP je nástupcem metody BOOTP a slouží pro automatické přidělené IP adresy koncovému počítači. Je definován v RFC 2131. [20]. Na rozdíl od BOOTP podporuje dynamickou alokaci adres a není nutná manuální tvorba profilů pro každý jeden počítač na síti. Společně s IP adresou počítače se předává adresa výchozí brány a adresy DNS serverů. Mezi další časté parametry patří DNS doména pro vyhledávání a adresa serverů pro synchronizaci času pomocí Network Time Protokolu – NTP [21]. Díky připravenosti protokolu na předávání doplňujících parametrů došlo k rozšíření i o volby sloužící k předání důležitých parametrů pro síťový boot [22]. Jedná se hlavně o volby obsahující adresu instalačního serveru a název souboru se zavaděčem. Adresa souboru se uvádí relativní ke kořenovému adresáři instalačního serveru. Pomocí tohoto mechanismu můžeme načíst zavaděč celého systému pro bezdiskové stanice nebo pouze modul se síťovým zavaděčem pro spouštění instalace a dalších servisních úloh.

2.5.4.3 File Transfer Protocol

Kapitola popisuje FTP ve verzi RFC 959 z roku 1985 [23], nikoliv starší verze definované nad protokolem ARPANET. Protokol si kladl za cíl umožnit sdílení dat a programů a odstínit uživatele od různých nesourodých protokolů pro sdílení. Toto se povedlo a dodnes je tento protokol využíván například pro distribuci balíčků distribucí GNU/Linuxu.

FTP využívá ke komunikaci Transmission control protocol – TCP [24] a konkrétně používá porty 20 a 21. Port 21 slouží k přenosů řídících příkazů a je přes něj ovládáno celé spojení. Vlastní data jsou pak přenášena na portu 20. Pro přenos souborů je k dispozici aktivní režim, kdy přenos dat inicializuje server (po dojednání přes řídící port). Toto nefunguje v případě, kdy server nemůže inicializovat přímé spojení s klientem, z důvodů vlastností nebo nastavení sítě Pro tyto případy je v protokolu k dispozici pasivní režim. V pasivním režimu probíhá inicializace datového spojení od klienta. Pokud je však klient za mechanismem překladu adres, je potřeba, aby překládající počítač měl znalosti FTP protokolu a musí do tohoto protokolu aktivně zasahovat.

2.5.4.4 Trivial File Transfer Protocol

TFTP je zjednodušená verze FTP. Podporuje pouze možností čtení respektive zápisu souboru na vzdálený server. Není zde podpora vypsání obsahu adresářů, ani jiné pokročilé funkce. Pro maximální zjednodušení byla také původně definovaná pevná velikost bloku 512 bytů [25]. Možnost přenášet větší bloky byla následně přidána [26], ale ostatní vlastnosti protokolu zůstaly zachovány a veškeré změny jsou zpětně kompatibilní.

Jedna ze zásadních vlastností je použití přenosové protokolu User Datagram Protocol – UDP [27] . Tento protokol patří mezi nespolehlivé, tudíž veškeré řízení musí řešit protokol vyšší vrstvy nebo případně aplikace. TFTP přenáší vždy jen jeden paket a poté čeká na potvrzení. Při ztrátě se přenos opakuje. Řešení má nevýhodu na pomalých linkách, kdy tento přístup snižuje celkovou propustnost, není však nutné implementovat na klientech cache, kam by se případně ukládaly další segmenty při průběžném potvrzování.

2.5.4.5 HyperText Transfer Protocol

HTTP [28] byl prvně definován v roce 1991 na půdě organizace World Wide Web Consortium - W3C a v popsané verzi 0.9 uměl pouze metodu GET s parametrem názvu souboru a tento soubor vrátil. Následující rok byl popsán návrh verze HTTP 1.0 [29]. Vývoj se postupně přenesl na půdu Internet Engineering Task Force - IETF [30] kde bylo HTTP 1.0 standardizováno v roce 1996 jako RFC 1945 [31]. V současné době je aktuální verze HTTP/2 [32], ale tato verze protokolu je využívám primárně na webu a není relevantní pro naše použití.

Proti FTP je u HTTP výhoda v použití pouze jednoho portu a to konkrétně TCP portu 80 přes který probíhá veškerá komunikace. Oproti TFTP nabízí HTTP především na lokálních sítích vyšší rychlost přenosu. HTTP a FTP ovšem vyžadují speciální zavaděč s podporou těchto protokolů, čímž může dojít k omezení kompatibility se staršími počítači. Pro podporu TFTP stačí základní zavaděč a vzhledem k celkem malým souborům přenášených ve fázi zavádění systému není toto omezení u TFTP natolik zásadní, abychom ho museli řešit.

2.6 Automatická instalace operačního systému

Pro naplnění cílů práce se z možností popsaných v kapitole 2.1 nejvíce hodí metoda automatická instalace, která nabízí vysokou variabilitu a míru autonomie celého procesu. Pro automatickou instalaci operačních systémů máme k dispozici větší množství nástrojů, které se liší škálou nabízených možností instalace, podporou různých operačních systému, možností konfigurace a nástroji pro řízení a sledování instalačního procesu.

Vlastnosti relevantních projektů, jejich silné stránky a omezení budou rozebrány v následujících kapitolách. Protože některé nástroje mají své specifické prerekvizity, jako je závislost na konkrétních síťových protokolech, budou u konkrétních nástrojů popsané i tyto prerekvizity.

2.6.1 Preseed

Preseed slouží pro vytvoření automatických instalací distribucí GNU/Linuxu založených na systému Debian [33]. Instalaci je možno přizpůsobit a automatizovat pomocí vytvoření preseed souboru. V tomto souboru jsou předpřipravené odpovědi,

které běžně vyžaduje debian installer pro konfigurační možnosti nebo při nastavování balíčků.

Preseed soubory je možné použít pro částečné nebo úplné přednastavení instalace. Při částečném přednastavení se jedná například o nastavení vhodného zrcadla pro stahování balíčků a nastavení odesílání mailů. Pro úplnou instalaci se dají přednastavit veškeré volby instalačního procesu. Pro tvorbu základní instalace jsou k dispozici ukázkové pressed soubory [34].

Preseed nabízí několik možností distribuce konfiguračního souboru. Pro síťovou instalaci, uvažovanou v této práci, jsou k dispozici dvě možnosti. Zabalení konfigurace do initrd obrazu nebo distribuce po síti. Zabalení do initrd umožňuje načtení už od počátku instalace včetně nastavení sítě. Je však nutné s každou změnou preseed souboru vygenerovat nový initrd obraz, což je časově náročné a snižuje to flexibilitu celého řešení. U načtení ze sítě je nutné projít manuální kroky až k přípravě sítě nebo spustit instalaci v automatickém režimu. Manuální kroky se snažíme v celé práci eliminovat a nejsou tak alternativou. Automatický režim se pokusí získat nastavení sítě a následně si stáhnout konfiguraci instalace. Veškeré případné dotazy, bez kterých se dá docílit nastavení sítě, jsou pozdrženy, dokud nedojde ke stažení preseed souboru. V této konfiguraci se může předat název souboru jako parametr jádra pro spuštění, což jsme schopni ovlivnit v PXE menu nebo jako volba v DHCP zprávě použité pro nastavení adresy, jak je popsáno v dokumentaci [35]. Jsme tak schopni spustit vlastní instalaci přímo ze síťového zavaděče.

Omezením této metody je nutnost využít pro instalaci volné místo na disku nebo nové rozdělení celého disku. Není možno využít už existující oddíly vytvořené například při předchozí instalaci. Z důvodů tohoto omezení by nebylo možné přeinstalovat pouze GNU/Linux samostatně, ale vždy by se musel přeinstalovávat celý počítač.

Metoda má podporu pro instalaci dodatečných balíčků, zde však záleží, jaké všechny otázky může debian installer [36] klást, protože otázky mohou být kontextové a při výchozím průchodu nemusí být zobrazeny. Aby se tomuto předešlo, je doporučeno nainstalovat poprvé veškeré požadované balíčky ručně a následně uložit použité volby do pressed konfiguračního souboru.

Po provedení instalace je možné spouštět i nástroje pro přizpůsobení systému, například stažení nebalíčkovaného software nebo úprava nastavení, které není možné změnit přes debian installer. Pomocí poinstalačních kroků by bylo možné nasadit na instalovaný stroj i MS Windows, ovšem pouze metodou klonování obrazu disku popsanou v kapitole 2.1.2. Vzhledem k provázání na instalační proces by bylo však komplikované nasadit pouze MS Windows bez předchozí instalace GNU/Linuxu, což by negativně ovlivnilo variabilitu celkového řešení.

2.6.2 Fully Automatic Installation

Fully Automatic Installation – FAI [37] je neinteraktivní systém pro instalaci, úpravy a správu systémů GNU/Linux a správu programů a konfigurací na těchto systémech instalovaných. Umožňuje nasazení na fyzických počítačích, virtuálních počítačích nebo prostředí chrootu a je použitelný pro malé instalace i rozlehlé clustery.

Instalace s FAI potřebuje podpůrné funkce na instalačním serveru. Krom PXE a TFTP, které obsahuje vlastní, ale umožňuje použití již existujících prostředí, je to hlavně Network File System server, přes který přistupuje ke konfiguračním souborům, na základě kterých se následně provede instalace. Veškeré tyto prerekvizity je však možné nainstalovat na většině systémů pomocí balíčku `fai-quickstart`, který má v závislostech uvedené všechny ostatní potřebné balíčky, které se spolu s ním nainstalují. Následně zavoláním programu `fai-setup` připravit celé potřebné prostředí.

Základním konceptem FAI jsou třídy. Jedná se o stavební bloky, které umožňují definovat jednotlivé aspekty instalace jako je rozdělení disku, nastavení sítě, nainstalované balíčky nebo použité konfigurace. Tyto samostatné celky se dají téměř libovolně kombinovat a umožňují vytvoření systému s vysokou variabilitou při zachování znovupoužitelnosti. Pro vytvoření kompletního postupu instalace se třídy spojují do profilů, které určují finální stav. Můžeme tak vytvořit různé profily pro různé skupiny hostů, umožňující instalaci specifických ovladačů pro jednotlivé typy počítačů nebo i pro jednotlivé koncové počítače. Výběr cílového profilu je možné provádět dynamicky během instalace na základě zjištěných vlastností cílového počítače nebo třeba na základě jména daného počítače.

Během instalace nejsme omezování jen jednotlivými úkoly, které podporuje přímo FAI, ale můžeme připojit vlastní skripty a libovolně upravit průběh instalace. Tyto skripty nám mohou například reportovat dokončení instalace zpět do systému pro správu. Pomocí těchto skriptů je také možné nasadit do připraveného oddílu disku MS Windows pomocí metody klonování obrazu disku popsaného v kapitole 2.1.2.

2.6.2.1 Network File System Protocol

Pro zpřístupnění souborů potřebný pro běh FAI a konfigurací se využívá Network File System Protocol – NFS. Jedná se o produkt firmy Sun Microsystems, který byl uveden v roce 1984 ještě pod názvem Sun Network Filesystem v roce 1985 [38]. Následně byl standardizován na půdě IETF v podobě RFC 1094 [39] pod názvem Network File System Protocol - NFS. Jedná se o síťový souborový systém. Již v návrhu protokolu byla snaha vytvořit celé řešení maximálně bezstavové, což zvyšuje odolnost celého řešení proti výpadkům.

Souborový systém nabízí tradiční stromovou strukturu uložených souborů včetně uživatelských oprávnění známých z UNIXových systémů. Pokročilé přístupová práva ve formě access control lists - ACL byla přidána v poslední verzi NFSv4 definovaném RFC 5661 [40].

NFS je primárně rozšířeno v UNIXových systémech. Existují i možnosti jak připojit NFS i do systému MS Windows [41], ale není to tak běžné a ve Windows sítích, případně hybridních systémech se častěji používají jiné protokoly, například Server Message Block – SMB [42] popsaný v jedné z následujících kapitol.

2.6.3 Kickstart

Jedná se obdobné řešení jako preseed vyvinuté primárně pro distribuce založené na balíčkovacím systému RPM [34] jako jsou RedHat Enterprise Linux [44] nebo Fedora [45]. Cílem je umožnit bezzásahovou instalaci velkého množství počítačů pomocí připraveného konfiguračního souboru. Systém je zaměřen primárně na síťové instalace a umístění souboru je možné na webovém serveru, FTP nebo NFS úložišti. V případě potřeby je možné tento soubor uložit i na jiné medium, třeba instalační USB disk.

Soubor s konfigurací instalace je textovým souborem, který je možné tvořit a upravovat v libovolném textovém editoru. Vzhledem k možné rozsáhlosti se však doporučuje jeho tvorba z již provedené manuální instalace podle postupu dle kapitoly 2.1.1 [46]. Na referenčním počítači, nebo ve virtuálním PC provedeme instalaci systému včetně všech potřebných nastavení a úprav a následně vygenerujeme konfigurační soubor, který reflektuje stav počítače. Při nutnosti odchýlení se od nastavených úprav můžeme tento soubor ručně editovat, nebo na referenčním stroji upravit stav a znovu soubor vygenerovat. Je k dispozici i webový konfigurační nástroj od společnosti RedHat, přes který je možné se soubor vytvořit [47], ale ten podporuje pouze RedHat Enterprise Linux [44] a pro jeho použití je nutný účet na RedHat portálu. Předání konfiguračního souboru je možné pomocí parametru pří spuštění instalace pomocí PXE kdy položka v menu může spustit síťovou instalaci.

V případě systému založených na Debianu je možné kickstart metodu použít obdobně jako v případě preseed souboru. Projevují se tu však různá drobná omezení v některých parametrech, které není možné touto metodou nastavit. Zde by mohla být omezující absence nastavení firewallu na cílovém počítači, přidávání modulů do jádra a nastavení kerbera [48], který se na univerzitě používá pro jednotné přihlašování. Tyto chybějící možnosti by mohlo být možné nahradit úpravami spouštěnými po instalaci nebo kombinací s preseed souborem. Obě možná řešení by však zvyšovala komplexitu celého systému. Instalace MS Windows by se obdobně jako u preseed metody 2.6.1 dala řešit klonováním obrazu disku dle kapitoly 2.1.2

2.6.4 Microsoft Deployment Toolkit

Microsoft Deployment Toolkit – MDT [49] slouží k automatickému nasazování systémů v organizacích. Jsou k dispozici tři úrovně autonomie instalačního procesu.

- User-driven installation – Předkonfigurovaná síťová instalace, kde uživatel nebo administrátor během procesu doplňuje volby specifické pro daný instalovaný počítač.
- Lite-touch installation – Obdobné vlastnosti jako User-Driven installation, přidává podporu instalace z přenosného média a nemá přímou závislost na Windows Serveru

- Zero-touch installation – Téměř plně autonomní instalace, vyžadující Microsoft System Center Configuration Manager [50] a další serverové komponenty. Vyžaduje ruční potvrzení zahájení instalace.

Všechny tři metody jsou do značné míry autonomní a liší se hlavně mechanismem spouštění a průběhem.

Během tohoto scénáře se nejprve tvoří referenční obrazy instalace, které se pak nasazují na cílovém počítači. Podle zvolené metody se používají příslušné komponenty MS Windows serveru, které se následně využívají i pro samotnou instalaci.

Všechny tři metody potřebují ke svému přípravě MS Windows Server. Metody User-driven a Zero-touch installation potřebují MS Windows server i pro fázi instalace, protože řídí celý její průběh a jsou na něm uloženy konfigurace, instalační obrazy a databáze instalovaných počítačů. Pro spuštění instalace se používají minimalistická instalační média s Windows Preinstallation Environment – Windows PE [51] doplněném dle zvoleného scénáře nástrojem Task Sequence Wizard případně Windows Deployment Wizard.

Start instalace ze sítě je možné zajistit pomocí nástroje Windows Deployment Services - WDS poskytující služby PXE serveru. Tato funkcionalita se dá přidat na MS Windows server jako jeho role. Server pak slouží jako transportní komponenta a sám o sobě může jenom poskytnout možnost startu ze sítě a nabídku obrazů k instalaci. Celou autonomní instalaci ovšem nezajistí. Pomocí této metody také není možné nasadit na stejný stroj také GNU/Linux

2.6.5 Windows Assessment and Development Kit

Wondows Assessment and Development Kit - Windows ADK [52] je sada nástrojů pro přípravu instalací počítačů ve velkém měřítku. Sada obsahuje několik samostatných komponent, které umožňují připravit vše potřebné pro různé instalační scénáře, jako je nasazování hotových obrazů disků, tak online instalace s požadovanými parametry. V následujících kapitolách si představíme tři důležité komponenty tohoto řešení. Konkrétně si ukážeme nástroj pro vytvoření minimalistického běhového prostředí, ze kterého je možné připojit síťovou složku a spustit instalaci. Dále nástroj pro úpravu instalačních obrazů, pokud chceme mít předpřipravený obraz se společnými

nástroji a na závěr nástroj pro tvorbu konfiguračních souborů pro bezzásahovou automatickou instalaci.

2.6.5.1 Windows Preinstallation Environment

Windows PE je minimalistický operační systém určený pro spouštěných podpůrných úloh spojených s instalací nebo opravami systému. Umožňuje například připravit disky, instalovat systém z lokálního nebo síťového úložiště, zachytit a nasadit obraz disku, provést úkoly pro opravu poškozeného systému nebo zachránit data z neběžícího systému.

Obraz Windows PE se dá vytvořit pomocí rozšíření nástroje Windows ADK, kde můžeme zvolit, zda se bude prostředí spouštět v interaktivním módu, nebo bude provádět automatizované úkoly. Je možné vytvořit Windows PE prostředí i na systému GNU/Linux. V takovém případě se použije instalační DVD operačního systému a nástroj `mkwinpeimg` [53].

Prostředí Windows PE je možné spustit i přenosného média, jako je USB, nebo pomocí startu ze sítě, kde nejprve nastartuje PXE prostředí, které následně načte obraz Windows PE a předá mu řízení.

2.6.5.2 Deployment Image Servicing and Management

Sada nástrojů pro příkazovou řádku Deployment Image Servicing and Management - DISM [54] slouží pro správu obrazů virtuálních disků a instalačních obrazů. Pomocí této sady je možné obrazy připojovat, spravovat, přidávat a odebírat komponenty a měnit vnitřní uspořádání jednotlivých obrazů. Primárně je určen pro práci s neaktivními obrazy, ale v omezené míře je možné pracovat i s obrazy běžícího systému.

Postup práce pro vytvoření obrazu instalace, který je určen k následnému nasazení sestává z instalace referenčního počítače. Po nastavení všeho potřebného se provede generalizace instalace pomocí nástroje sysprep, který je součástí sady DISM, aby se dal výsledný obraz nasazovat na další počítače a nedocházelo ke konfliktům ve jménech počítačů a dalších unikátních parametrech.

V případě potřeby aktualizovat část obrazu, změnit nastavení nebo přidat ovladače, stačí provést příslušné operace pomocí nástroje DISM a není nutné znova

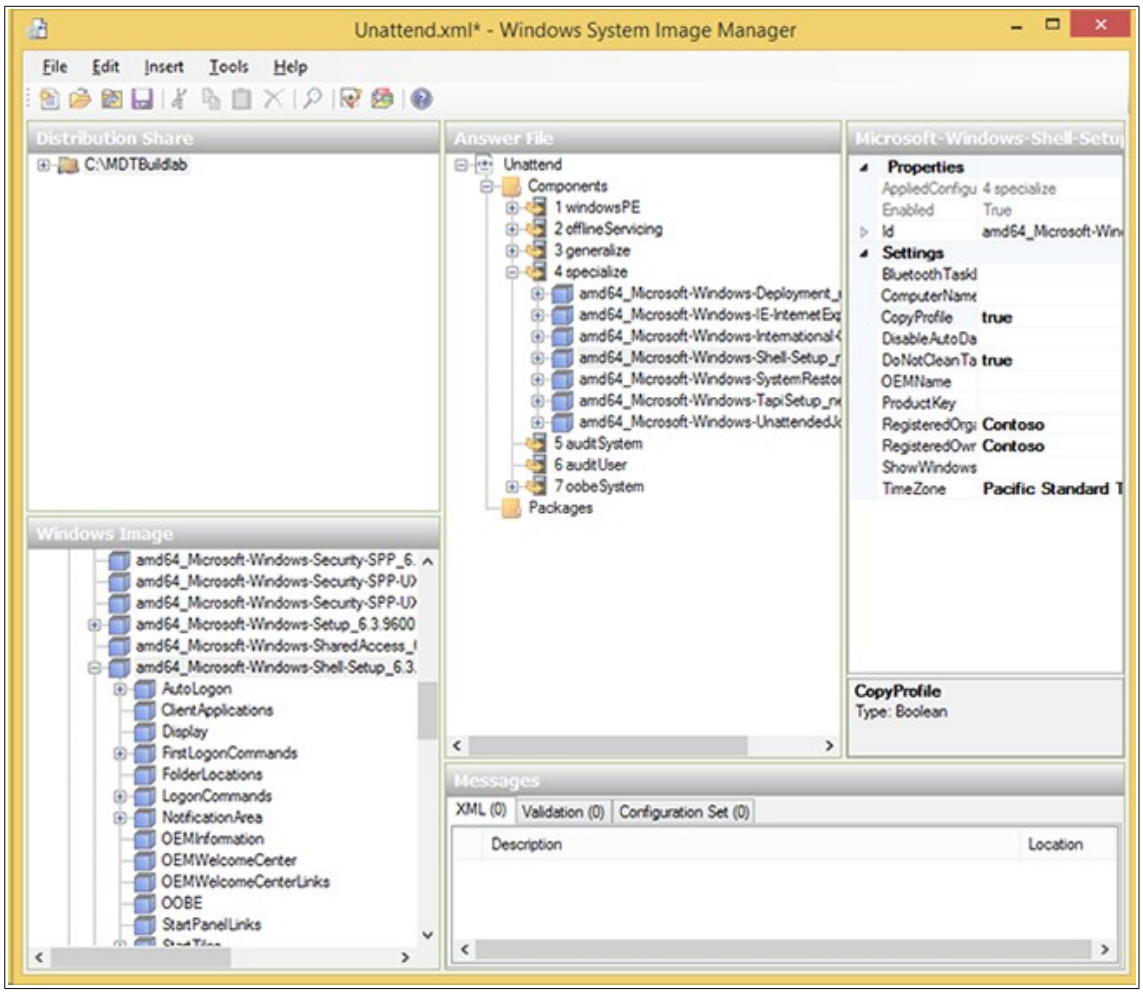
reinstalovat celé prostředí. Výsledkem je však vždy obraz disku, který se nasazuje na cílový počítač.

Díky generalizaci by bylo toto řešení použitelné v případě počítačů s jedním operačním systémem, ale v samotném nástroji je problematické snadno specifikovat pouze část disku, kam by se měl systém nasadit a je to dáno i potřebou pro systémový oddíl, proto primárně slouží tento nástroj pro nasazování na nové počítače, kde jediným systémem je MS Windows.

2.6.5.3 Windows System Image Manager

Pro vytvoření souboru pro řízení bezobslužné instalace je určen nástroj Windows System Image Manager - Windows SIM [52]. Tento nástroj na základě souborů z instalačního média nebo katalogu umožňuje vytvořit soubor s konfigurací, která se během instalace aplikuje na daný počítač.

Samotná instalace se skládá z několika fází a pro všechny tyto fáze můžeme definovat odpovídající akce. Hned v první fázi, Windows PE si můžeme zvolit třeba jazyk rozhraní, ve kterém bude instalace probíhat a nebo rozdělení disků. Pro instalaci na počítači, která obsahuje i další instalace můžeme přeskočit krok rozdělení disku a rovnou naformátovat vyhrazené oddíly připravené v předchozí instalaci. Ukázku rozhraní vidíme na Obr 2.5



Obr 2.5: Rozhraní nástroje Windows System Image Manager Zdroj: [52]

Výsledný XML soubor se dá použít pro síťovou instalaci i pro distribuci společně s instalátorem na přenosném disku. Pomoc výše popsaného nástroje DISM je možné ho přidat do instalačního obrazu systému. V našem případě se bude jednat o soubor na síťovém disku, kde budou také umístěné instalační soubory systému.

2.6.5.4 Server Message Block

Pro zpřístupnění instalačních souborů do prostředí Windows PE, které je pak následně použije je možné využít síťové úložiště zpřístupněné pomocí Server Message Block protokolu. Původně proprietární protokol jehož vývoj začal v IBM v roce 1983 a jeho cílem bylo rozšířit schopnosti operačního systému DOS o možnost přistupovat k souborům na síti [55]. Postupně se do vývoje zapojila i společnost Microsoft. Vzhledem k výhradnímu postavení toho protokolu pro síťovou komunikaci v prostředí MS Windows a absenci veřejně dostupné specifikace vznikaly další implementace,

jednalo se primárně o serverové části, pomocí reverzního inženýrství. Asi nejznámější je implementace jménem Samba[42]. Pro podporu interoperability byly specifikace nových verzí protokolu již otevřené [56].

Hlavní funkcionalitou protokolu je sdílení souborů a tiskáren v lokální síti. V minulosti se jednalo ještě o sdílení sériových portů, ale to již v dnešní době není využíváno. V novějších verzích přibylo hlavně silnější šifrování přenášených dat a kontrola jejich integrity.

2.6.6 AIO boot

Jedná se o all-in-one (odtud název AIO [57]) nástroj pro tvorbu spouštěcích USB/HDD s podporou mnoha systémů. Projekt je aktivně vyvíjen, poslední verze je v době psaní práce cca 10 měsíců stará [58]. Ačkoliv je systém primárně zaměřen na tvorbu spouštěcích disků, podporuje také start po síti a spuštění instalace MS Windows a GNU/Linuxu.

Instalace systému jsou však v základní podobě, to znamená že z prostředí vytvořeného pomoci nástroje AIO boot jsme schopni spustit tradiční instalační dialog, jako bychom instalovali počítače lokálně a je nutné se těmito dialogy proklikat a zadat potřebné údaje v příslušných krocích. Tento postup neodpovídá požadavkům na automatizaci.

Pro instalaci systému MS Windows je zde použito minimalistické prostředí Windows PE, které vypadá jako vhodný nástroj pro tento úkol, jen by musela být doplněna automatizace této vrstvy. Lehce nedůvěryhodně působí odkaz na stažení Windows PE prostředí směřující na anonymní úložiště, když je možno toto prostředí snadno vytvořit z instalačního média MS Windows a to nejen přímo na systému MS Windows, ale i v GNU/Linuxu. Možnosti instalace linuxových systémů jsou obdobné, tzn. spuštění standardního instalátoru daného systému a jeho manuální průchod.

Celé řešení je distribuováno jako jeden exe soubor, který podle nadefinovaných voleb vytvoří spustitelné medium. Ačkoliv je volně k dispozici, nejedná se o open-source řešení, a tudíž by bylo nutné provádět změny až ve výsledném vytvořeném médiu a tyto změny by musely být uchovávány odděleně, aby mohly být aplikovány při tvorbě aktualizovaného média. Stejně tak doplnění nutných částí pro automatickou

instalaci se touto uzavřeností stává netriviálním problémem a bylo by potřeba kontinuální péče, aby se podařilo udržet řešení funkční požadovaném stavu. Některé komponenty, jako například výše zmíněné WinPE by ovšem při vhodném doplnění mohli být cestou, jak spustit automatickou instalaci MS Windows.

2.6.7 Projekt FOG

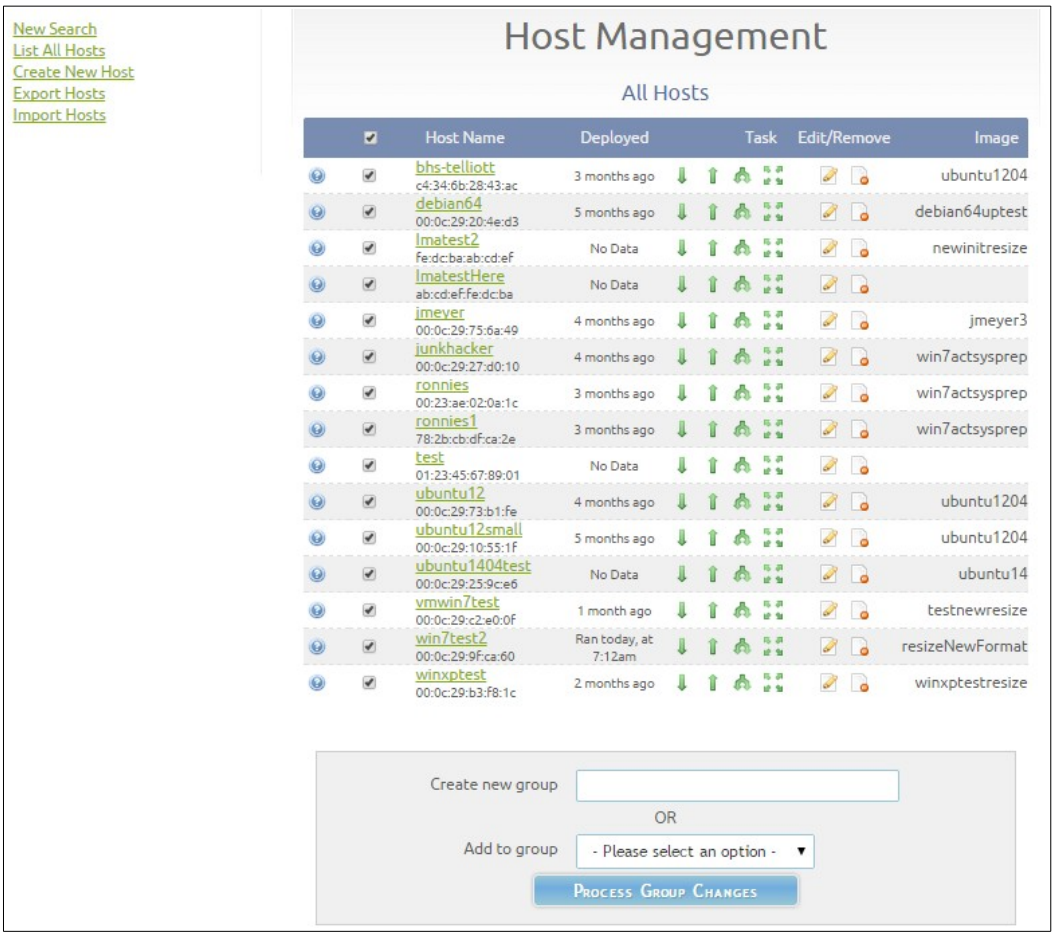
Projekt FOG vznikl ve vzdělávací instituci z podobných pohnutek, jako řeší tato práce [59]. Protože je postaven na opensource technologiích, rozhodli se ho autoři poskytovat také jako opensource a kompletní zdrojové kódy jsou dostupné na githubu [60]. Ačkoliv vývoj stále probíhá a aktuálně je k dispozici řada 1.5, dokumentace dosti zaostává za vývojem a nejsou výjimečné odkazy na verzi 0.3. Dokumentace končící Windows 7, i když systém podporuje i aktuální Windows 10 nebo popisy k verzi 1.3.0 z roku 2016 s poznámkou - “stále ve vývoji”.

Projekt se zaměřuje na přípravu počítačů pomocí přenesení obrazu disku, tyto obrazy mohou obsahovat více systému. Je také podporována možnost přípravy počítačů s více instalovanými pevnými disky. Obrazy disků jsou vytvářeny na referenčním počítači, který je nainstalován dle požadavků dle postupu v kapitole 2.1.1 a následně je uložen jeho obraz. Přizpůsobování obrazů je podporováno jen pro MS Windows pomocí snapin modulů, což jsou upravené instalátory doplněné o metadata. Možnost instalace těchto modulů je podmíněna během FOG klientského programu na cílovém počítači.

Správa obrazů a jejich nasazení je řízeno pomocí webového rozhraní zobrazeného na 2.6.7. Jedná se však o jiné rozhraní, než je zobrazeno na domovské stránce projektu s poznámkou “Zobrazené rozhraní bude dostupno v budoucím vydání” [61]. Členění tohoto rozhraní vypadá velmi logicky a mohlo by splňovat požadavky na možnosti správy.

Jedinou avšak zásadní slabinou tohoto řešení je úzká vazba na klonování hotových obrazů disků. To jde přímo proti požadavku snadné správy a umožnění změn v instalačním procesu. Doplnění dalších instalačních metod by znamenalo od základu změnit logiku aplikace. Protože se jedná o zásadní změnu aplikace, hrozí zde riziko, že by tyto změny nebyly přijaty do hlavního stromu projektu a v tom případě by vznikl fork projektu, který by musel být pravidelné udržován synchronizovaný s hlavním

projektem. Tyto aspekty představují vysoké riziko pro udržitelnost projektu a tudíž nebude tento projekt použit.



Obr 2.6: Webové rozhranní nástroje fog zdroj [62]

2.7 Rozhraní pro řízení a správu instalací

Administrátoři systému potřebují mít možnost sledovat stav instalací a zadávat povely k instalacím novým. Pro tyto operace je třeba mít k dispozici vhodné rozhraní. Dostupné možnosti rozebírají následující kapitoly.

2.7.1 Nativní aplikace

Aplikace psané přímo pro konkrétní operační systém nabízejí výhody lepší integrace do systému, nabízejí také uživatelům známe rozhraní, které zapadá do vzhledu systému. Mezi jednu z výhod patří také možnost fungovat lokálně bez síťového připojení, což ovšem v tomto případě není relevantní, protože potřebujeme obsluhovat

síťové instalace. Výhodou nativní aplikace je také možnost snadné komunikace s hardwarem daného zařízení. Je zde tedy možnost přímo z aplikace generovat Magic pakety pro WoL. Zde musí být splněna podmínka přítomnosti počítači na stejné síti jako spouštěné počítače, případně použít některý z nástrojů popsaných v posledním odstavci kapitoly 2.3.1.

Mezi nevýhody naopak patří nutnost vytvořit aplikaci kompatibilní s konkrétním systémem. Pro podporu co nejširšího spektra by to znamenalo aplikaci s podporou MS Windows, MacOS X, GNU/Linuxu a systémy pro mobilní zařízení. Zde je možné použít pro psaní jazyk JAVA, který nám umožní udělat multiplatformní aplikace.

2.7.2 Mobilní aplikace

Speciální verze nativní aplikace optimalizovaná pro použití na malých displejích mobilních zařízení je mobilní aplikace. U mobilních aplikací máme na výběr, zda budeme vyvíjet na konkrétní platformu nebo využít multiplatformní framework.

Úlohu mobilní aplikace může také zastoupit webová aplikace s responzivním rozhraním, proto se jako nejvhodnější volba pro celé řešení jeví právě vývoj webové aplikace, kterou jsme schopni pokrýt většinu potřeb

2.7.3 Webové rozhraní

Další možností je ovládání a sledování instalačního procesu pomocí webové aplikace. Webová aplikace má výhodu v možnosti využití z různých nezávislých míst pouze otevřením internetového prohlížeče. Jedná se tak o platformě nezávislé řešení, nabízející velmi širokou dostupnost celého systému bez ohledu na systém nebo zařízení používané administrátorem.

Webové rozhraní nenabízí tak široké možnosti komunikovat s hardwarem na nízké úrovni, ovšem ke své funkci potřebuje centrální uzel, kde bude nasazena databáze. Na tento uzel můžeme nasadit worker komponentu, která poběží jako služba a bude provádět nízkoúrovňové příkazy.

2.8 Požadavky na nový systém

Hlavním cílem nového systému je automatizace celého instalačního procesu. Administrátor by měl být schopen ze svého pracoviště plánovat, spouštět a monitorovat instalaci nejen na úrovni laboratoří, ale i jednotlivých počítačů.

Pro start instalace je nutné, aby bylo možné počítače vzdáleně spustit a načíst zavaděč ze sítě. V případě použití boot menu u síťového zavaděče by mělo být menu neinvazivní. Většina startů bude pro běžné použití a tudíž by menu síťového zavaděče nemělo uživatele obtěžovat.

Při jednotlivých krocích se počítače spouští s různými startovacími parametry dle průběhu instalace. Boot menu by mělo také podporovat možnosti spouštět jednotlivé kroky instalace i ručně případně umožnit boot z externích médií. Tyto volby jsou určeny jen pro administrátora, a proto by k nim měl být omezen přístup například pomocí nutnosti zadat heslo.

Instalované systémy musí umožnit nasazování včetně předinstalovaných aplikací a přizpůsobení pro konkrétní počítač. Potřebná je také možnost rozdělení disku. Instalace by měla proběhnout včetně aktualizací a s instalací potřebného softwarového vybavení. V systému by měly být dostupné veškeré potřebné konfigurace, kořenové certifikáty univerzitní certifikační autority a přístupy ke sdílenému úložišti AFS.

Obsluha celého procesu by měla probíhat z aplikace dostupné z počítačů administrátorů. Na základě poznatků získaných v kapitole 2.7 by byla nejvhodnější webová aplikace, aby nebyla obsluha instalačního procesu vázaná na konkrétní počítače. V této aplikaci je požadováno spuštění instalace pro skupinu počítačů, například laboratoř, pro jednotlivé počítače, zapnutí počítače a volba definice instalačního procesu.

2.8.1 Zhodnocení existujících řešeních

Na základě požadavků definovaných v kapitole 2.8 byla porovnána řešení popsána v kapitole 2.6 a výsledek srovnání klíčových parametrů je zobrazen v tabulce 2.1.

Nástroje AIO popsaný v kapitole 2.6.6 a projekt FOG (2.6.7) představují zástupce komplexních systémů, které umožňují řešit velkou část instalačního procesu. Bohužel

jsou tyto nástroje orientovány na nasazení klonováním pomocí obrazů disků, což sebou přináší nevýhody popsané v kapitole 2.1.2.

U ostatních nástrojů je vidět jejich zaměření na systémy, pro které byly vytvořeny a chybí zde podpora druhé požadované platformy. Omezení je možné obejít klonováním, což ale znamená, že plný potenciál bychom využili jen u systému, jehož nástroj jsme použili a druhý systém bude jen naklonován a bude pouze popelkou. Je však možné využít síly jednotlivých nástrojů a používat jeden pro instalaci GNU/Linuxu a druhý pro MS Windows.

Protože ani jedno řešení nesplňuje požadavky a případné změny na doplnění funkcionality jsou netriviální bylo přistoupeno k tvorbě nového systému, který se ale v maximální možné míře pokusí využít již existující komponenty.

Požadovaná vlastnost	**AIO**	**FOG Project**	**Preseed**	**FAI**	**Kickstart**	**MDT**
Instalace po síti	Po přizpůsobení	**Ano**	**Ano**	**Ano**	**Ano**	**Ano**
Probuzení instalovaných počítačů	Ne	**Ano**	Ne	Ne	Ne	Ne
Instalace MS Windows	**Ano**	Klonování	Klonování	Klonování	Klonování	**Ano**
Instalace GNU/Linuxu	**Ano**	Omezeně	**Ano**	**Ano**	**Ano**	Ne
Podpora automatizované instalace	Ne	Ne	Omezená	**Ano**	**Ano**	Omezená
Rozhraní pro správu instalací	Ne	**Ano**	Ne	Ne	Ne	Částečná

Tab 2.1: Zhodnocení schopností existujících řešení

2.8.1.1 Vhodné řešení pro instalaci GNU/Linuxu

Po zhodnocení výhod a nevýhod popsaných řešení jsem z podrobnějšího zkoumání vyřadil metodu kickstart, která má nemalé množství omezení, která nejsou kompenzována výraznými výhodami oproti dalším posuzovaným řešením.

Instalace pomocí nástroje preseed je sice omezena na systémy z rodiny Debian, ale jedná se o primárně používaný systém a v laboratořích, které jsou hlavním cílem práce. Drobnou slabinou je nemožnost snadno ošetřit první kroky instalace před konfigurací sítě. Problémem by však mohlo být vytváření preseed souborů, které jsou závislé na otázkách debian installeru, které se mohou u nových verzích balíčků měnit. I doporučená tvorba základů těchto souborů cestou provedení instalace není zcela uživatelsky přívětivá.

Vybraným řešením pro implementaci v rámci práce se tak stal systém FAI. Rozhodujícím byla snadná konfigurovatelnost a rozšiřitelnost celého systému a možnost nastavit kompletní cyklus instalace od prvopočátků. Podpůrným argumentem bylo také aktuální používání systému FAI v rámci KIV a CIV, které poskytuje možnost sdílení znalostí v rámci jednotlivých složek univerzity a zkracuje dobu nutnou na přípravu nasazení a případné zaškolení odpovědných pracovníků, které by bylo nutné s novými technologiemi seznámit.

2.8.1.2 Vhodné řešení pro instalaci Windows

Omezení MDT v rovině plně automatické instalace není možné obejít ani za použití nástroje WDS. Navíc pro celé řešení je potřeba MS Windows server, který by zbytečně zvyšoval nároky na nasazení a následnou údržbu celého řešení zvláště v situaci, kdy by tento server nemohl zastávat zároveň roli instalačního serveru pro instalace GNU/Linuxu. Vzrostla by tím také komplexita celého řešení a vyvstal by problém se synchronizací jednotlivých kroků, kdy by DHCP server musel střídavě odkazovat na jeden nebo druhý instalační server.

Pro tuto práci s jeví jako vhodné nástroje ze sady Windows ADK a to konkrétně Windows PE jako prostředí, ze kterého bude spuštěna instalace. Pro tvorbu automatizačních XML souborů pro bezzásahovou instalaci bude použit Windows SIM. Zároveň můžeme vytvořit i jednu instanci Windows PE prostředí, která se spustí v interaktivním módu a umožní administrátorům základní servisní úlohy nad již existující instalací MS Windows.

Windows PE prostředí i instalační soubory a XML soubory pro bezzásahovou instalaci budou umístěny na stejném serveru, jako instalační soubory pro GNU/Linux a i samotné PXE prostředí. Spuštění Windows PE prostředí, instalačního i servisního, bude realizováno přidáním voleb do spouštěcího PXE menu pro dané scénáře.

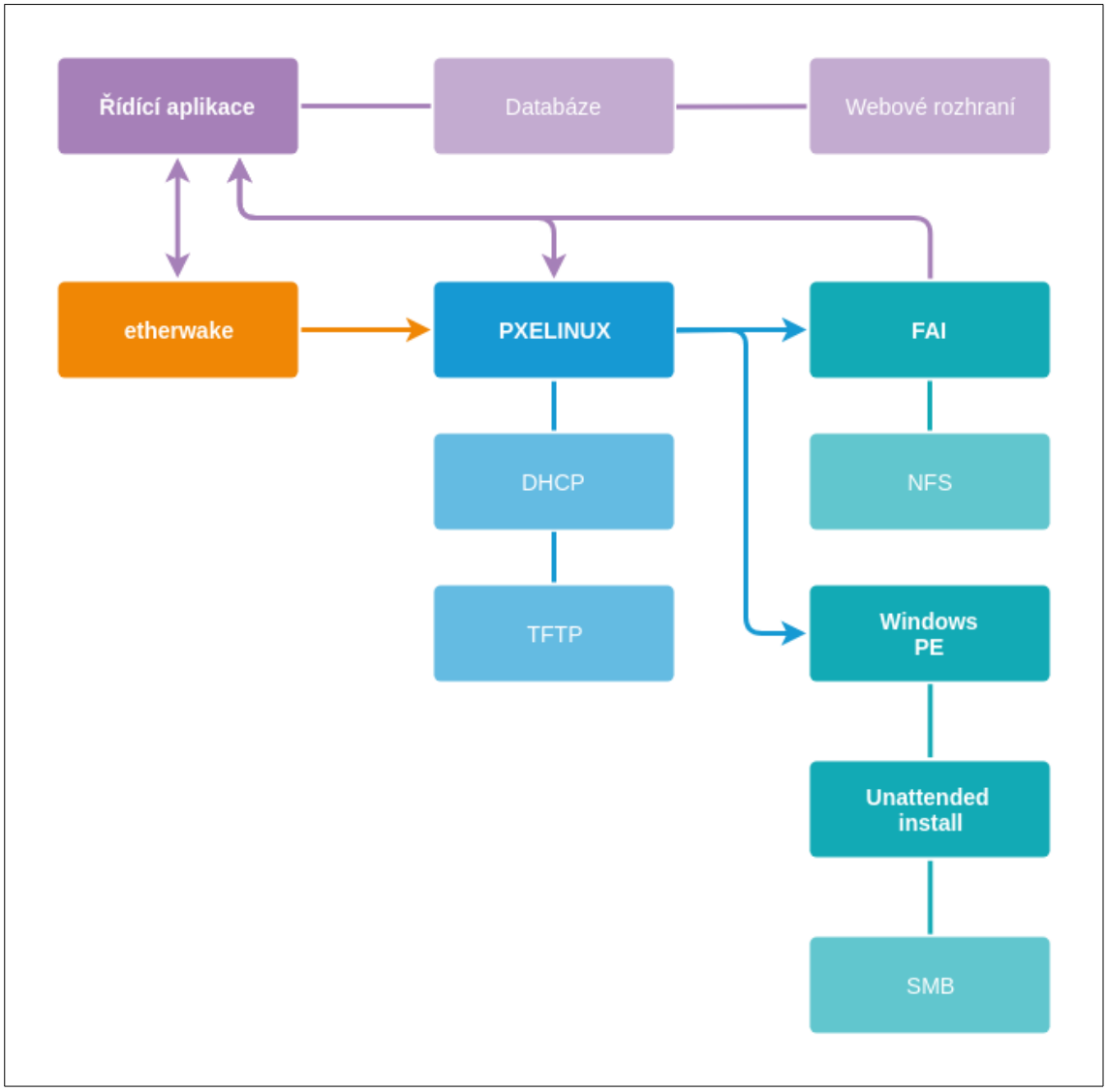
3 Nový systém automatické síťové instalace

Na základě poznatků získaných v kapitole 2 a z toho vycházejících požadavků sepsaných v kapitole 2.8, bylo přistoupeno k tvorbě nového systému, protože žádný ze zkoumaných systémů neodpovídá požadavkům na nový systém a nedají se ani dostatečně snadno upravit pro zamýšlené potřeby.

3.1 Architektura systému

Z jednotlivých nástrojů, popisovaných napříč kapitolou 2 byla vytvořena architektura zobrazena na Obr 3.1. Z důvodu kompatibility vybraných nástrojů byl jako operační systém, na kterém bude systém stavěn vybrán GNU/Linux. Instalační proces začíná probuzením počítače, kde k automatickému zapnutí počítače budeme využívat nástroj `etherwake`, který umožňuje krom specifikace cílové adresy vybrat také síťový port, přes který bude magic packet odeslán. To systému umožní probouzet počítače na různých segmentech sítě. Podmínkou je, aby měl řídící server připojena síťová rozhraní do všech požadovaných sítí.

Po spuštění počítače dojde k načtení síťové adresy, k tomu je třeba mít nainstalovaný a správně nastavený DHCP server. Tento server může být součástí řídícího serveru, nebo se může na sítí již nacházet. V takovém případě požádáme jeho administrátora o změnu parametrů. Pro přípravu zavaděče a jeho konfiguračních souborů bude využit nástroj PXELINUX. Konfigurace pro konkrétní počítač se bude odvíjet od konkrétní činnosti. Pro požadované činností vzniknou nutné konfigurační soubory, které budou následně linkovány pro jednotlivé počítače. Vše následně umístíme do složky, kterou zpřístupníme pomocí TFTP serveru.



Obr 3.1: Architektura nového systému pro správu instalací.

Instalaci GNU/Linuxu bude obstarávat nástroj FAI popsaný v kapitole 2.6.2. Tento nástroj a jeho prerekvizity nainstalujeme na řídící server. Technicky ovšem může být FAI server umístěn i na jiném server, který je dostupný PXE klientům. V takovém případě stačí mít na TFTP serveru jádro systému, kterému přidáme parametry pro načtení samotného FAI ze vzdáleného serveru.

Pro přípravu souborů využívaných během automatické instalace MS Windows bude potřeba počítač s tímto operačním systémem. Veškeré nástroje mohou běžet na desktopové verzi operačního systému a není potřeba jeho serverová varianta. Může se jednat i o virtuální počítač. Na tomto počítači bude nainstalován a spouštěn nástroj Windows ADK pro přípravu zavaděče a konfigurace instalací. Instalační soubory

i konfigurace pro instalaci se následně umístí na SMB server. SMB server může být umístěn na řídícím serveru.

Celý proces bude ovládán řídící aplikací která bude vyvinuta v rámci práce. Tato aplikace bude připravovat jednotlivé konfigurace pro PXE, vzdáleně zapínat počítače a sledovat stavy běžících instalací. Stavy a záznamy o úkolech budou ukládány do relační databáze. K této databázi se bude zároveň připojovat vytvořené webové rozhraní, které nabídne administrátorům přehled o aktuální situaci a možnosti spouštění jednotlivých úloh.

3.2 Příprava řídícího serveru

Výchozím stavem pro přípravu je server s nainstalovaným Debian GNU/Linuxem. Postup instalace je k dispozici v instalační příručce [63]. Minimální požadavky na hardware jsou uvedeny v kapitole 3.4 instalační příručky a sami o sobě jsou velmi nízké. Jako minimum pro základní instalaci Debianu je uvedené CPU Pentium 4 na frekvenci 1 GHz, 128 MB RAM (doporučeno 512 MB) a 2 GB místa na pevném disku. Pro naše použití je vhodné použít více jader CPU kokrétně 2 – 4 jádra. Při použití jednoho řídícího serveru na všechny role by měly být dostatečné 4 GB RAM. Úložiště je závislé na množství instalovaných variant, ale pro zamýšlené použití by mělo být dostatečná velikost pevného disku 20 GB. Tato hodnota se může změnit při využití většího množství instalovaných variant nebo při použití obrazů pro klonování. Množství síťových karet je závislé na množství obsluhovaných podsítí z důvodů omezení možností probuzení po síti popsaných v kapitole 2.3.1. Síťové karty by měly podporovat rychlost 1 Gb/s a být připojeny do sítě podporující tuto rychlost, aby se zkrátily časy nutné pro přenos dat mezi serverem a instalovaným počítačem.

Na server je nutné postupně nainstalovat potřebné softwarové vybavení pro poskytnutí instalačních služeb jednotlivým strojům a upravit jejich konfiguraci. V jednotlivých kapitolách jsou popsány zajímavé fragmenty konfigurace. Celé konfigurace jsou následně obsaženy v detailní instalační příručce Příloha D.

3.2.1 Nástroj na probuzení instalované počítače

Z nástrojů pro probuzení instalované stanice popsaných v kapitole 2.3.1 jsem vybral nástroj `etherwake`, protože umožňuje krom cílové stanice specifikovat

i odchozí rozhraní Magic paketu a je tak využitelný i na strojích s více síťovými kartami.

```
apt-get install etherwake
```

Funkčnost ověříme zavolání příkazu pro výpis verze, který by měl vypsat aktuální verzi nainstalovaného nástroje.

```
etherwake -v
```

Očekávaný výstup:

```
etherwake.c: v1.09 11/12/2003 Donald Becker, http://www.scyld.com/
Sp
```

3.2.2 Přidělení síťové adresy a instalačních parametrů

Pro předání informace o umístění instalačního serveru a názvu souboru se zavaděčem dle popisu v kapitole 2.5.4 bude použit protokol DHCP, neboť protokol BOOTP je již zastaralý. Na řídícím serveru nainstalujeme DHCP server a připravíme konfiguraci, aby server do sítě předával potřebné parametry. Pokud již v síti existuje DHCP server, který budeme používat, postačí do příslušné sekce přidat potřebné volby.

3.2.2.1 Instalace nového DHCP serveru

Jako DHCP server použijeme ISC DHCP server vyvíjený sdružením Internet Systems Consortium, neboť se jedná v podstatě o referenční implementaci protokolu DHCP.

```
apt-get install isc-dhcp-server
```

Po instalaci musíme ještě upravit instalační soubor `/etc/dhcp/dhcpd.conf` konkrétně sekci `subnet`. V této sekci si nastavíme krom běžných parametrů také příslušná nastavení důležitá pro načtení zavaděče ze sítě (vyznačené tučně)

`/etc/dhcp/dhcpd.conf`

```
subnet 147.228.63.0 netmask 255.255.255.0 {
     authoritative;
     range 148.228.63.100 147.228.63.200;
     option routers 147.228.63.1;
     option domain-name-servers 147.228.63.1;
     next-server 147.228.63.9;
     filename "pxelinux.0";
}
```

Dále nastavíme rozhraní, na kterém bude DHCP server naslouchat

`/etc/default/isc-dhcp-server`

```
INTERFACES="ens18
```

Po změně konfigurace provedeme restart DHCP serveru a následnou kontrolu, že služba naběhla v pořádku.

```
systemctl restart isc-dhcp-server
systemctl status isc-dhcp-server
```

Očekávaný výstup:

```
● isc-dhcp-server.service - ISC DHCP IPv4 server
   Loaded: loaded (/lib/systemd/system/isc-dhcp-server.service;
   enabled; vendor
   Active: active (running) since Mon 2019-06-24 17:33:36 CEST; 24h
ago
```

3.2.2.2 Úprava již existujícího DHCP serveru

V případě existence DHCP server, který chceme používat i nadále stačí do konfiguračního souboru přidat konfigurační volby. V případě ISC DHCP serveru se jedná o zmíněné dva řádky vyznačené tučně v předchozí kapitole a následný restart a ověření funkčnosti dle stejného postupu.

3.2.3 Distribuce síťového zavaděče

K distribuci síťového zavaděče, jeho konfigurace a jádra instalátoru bude z protokolů popsaných v kapitole 2.5.4 využíván protokol TFTP, který nabízí nejširší kompatibilitu. Pro zpřístupnění souborů pomocí tohoto protokolu musíme nainstalovat TFTP server. Z dostupných implementací jsou k dispozici `atftpd`, `tftpd` a `tftpd-hpa` . `Tftpd` není doporučován pro použití v kombinaci PXE [64], zbylé dva servery

jsou ovšem parametry velmi podobné, a proto jsem na základě předchozích zkušeností zvolil implementaci `tftpd-hpa.` Server nainstalujeme následujícím příkazem.

```
apt-get install tftpd-hpa
```

Můžeme použít výchozí konfiguraci umístěnou v souboru `/etc/default/tftpd-hpa`, kdy domovským adresářem pro umísťování souborů je `/srv/tftp` do tohoto umístění budeme následně umisťovat komponenty síťového zavaděče a jeho konfigurační soubory.

Správně spuštění ověříme příkazem níže

```
systemctl status tftpd-hpa
```

Očekávaný výstup:

```
● tftpd-hpa.service - LSB: HPA's tftp server
   Loaded: loaded (/etc/init.d/tftpd-hpa; generated)
   Active: active (running) since Mon 2019-06-24 17:31:36 CEST; 1
day 1h ago
```

3.2.4 Prostředí síťového zavaděče

Soubory zavaděče získáme v balíčku `pxelinux`. Tento balíček obsahuje pouze samotný zavaděč. Pro podpůrné nástroje, díky kterým můžeme vytvářet boot menu potřebujeme ještě nainstalovat balíček `syslinux-common`.

```
apt-get install pxelinux syslinux-common
```

Po instalaci je třeba zkopírovat příslušné soubory do adresáře odkud je může nabízet TFTP server což je v našem případě dle nastavení v kapitole 3.2.3 adresář `/srv/tftp.` Do tohoto adresáře zkopírujeme jak zavaděč, tak i moduly pro tvorbu spouštěcího menu a další operace.

```
cp /usr/lib/PXELINUX/pxelinux.0 /srv/tftp/
cp /usr/lib/syslinux/modules/bios/*.c32 /srv/tftp/
```

3.2.4.1 Tvorba konfigurace síťového zavaděče

Konfigurace se nachází v adresáři `/srv/tftp/pxelinux.cfg` kde zavaděč hledá konfiguraci dle pravidel popsaných v kapitole 2.5.3.1. V položkách menu je možné nastavit širokou škálu možností. Kromě základního menu také grafické ztvárnění pozadí a v neposlední řade i ochranu menu heslem. Tato vlastnost je velmi důležitá kvůli zabránění náhodného spuštění instalace běžným uživatelem.

Ochranu heslem je možné použít jak pro celé menu, tak i pro jednotlivé položky a hesla můžou být i různá. Nejedná se však o zcela praktické řešení a hrozí zde zapomenutí nebo ztráta hesel. Ačkoliv je k dispozici možnost použít heslo v plaintextu, nejedná se o bezpečnou metodu a místo toho je doporučeno použít jednu z podporovaných hashovacích metod, nejlépe SHA-256 nebo vyšší [65].

Na výchozí konfiguraci si můžeme popsat jednotlivé sekce. V úvodní sekci je definován použitý modul pro tvorbu grafického menu, úprava barev pro dobrou viditelnost na nastaveném pozadí a výchozí volba použitá po uplynutí časového limitu. Následuje hlavička a heslo pro ovládání menu. Samotná definice hesla ještě nezaručuje jeho použití. Pro toto je nutné přidat do každé sekce záznam `MENU PASSWD` bez uvedení parametru. Následují pak jednotlivé sekce pro položky menu.

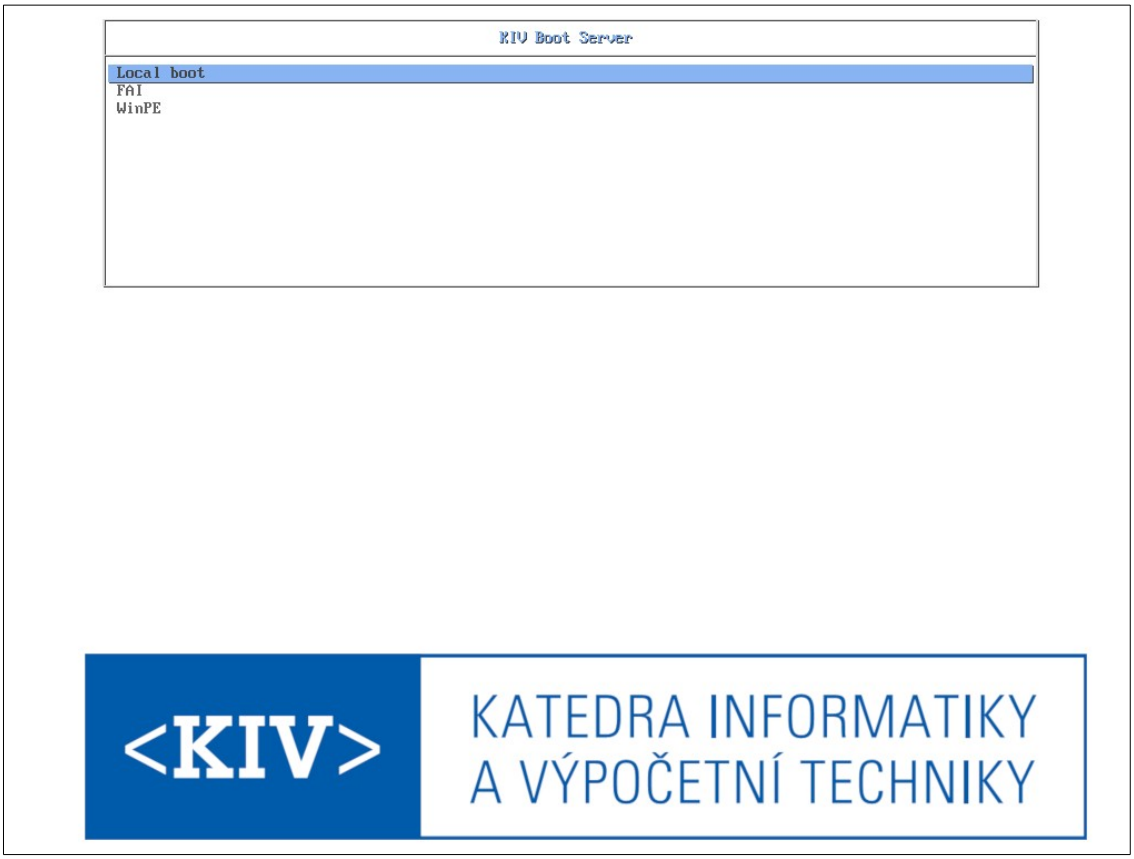
Položka `local` předává řízení lokálnímu zavaděči a je nastavena jako výchozí po uplynutí doby pro interakci. Další položky se již vztahují k jednotlivým instalačním metodám. Následuje zkrácený obsah souboru. Plné znění je k dispozici na CD ve složce `/tftp/pxelinux.cfg/` Výsledné menu zobrazuje Obr 3.2

`/srv/tftp/pxelinux.cfg/default - úryvek`

```
UI vesamenu.c32
PROMPT 1
TIMEOUT 50
ONTIMEOUT local
MENU RESOLUTION 1024 768
MENU BACKGROUND background.png
MENU COLOR sel   7;30;40    #f0000000 #c00090f0 all
MENU COLOR unsel  30;40    #e0000000 #00000000 none

MENU TITLE    KIV Boot Server
ALLOWOPTIONS 0
NOESCAPE 1
MENU MASTER PASSWD $5$KAL2oln1$5eOp...$
```

```
LABEL local
    MENU LABEL Local Boot
    MENU PASSWD
    MENU DEFAULT
    COM32 chain.c32
    APPEND hd0 0
```



Obr 3.2: Výsledné PXE menu

3.2.4.2 Variabilita konfigurace síťového zavaděče

Pro zajištění startu do požadovaného stavu je možné využít předdefinované konfigurace ve formátu `linux-instal[-profile]` a `windows-install[-profile].` Pro profil s názvem `uc326` bude výsledný název souboru pro instalaci GNU/Linuxu `linux-instal-uc326.` Tyto konfigurační soubory jsou k dispozici na přiloženém CD v adresáři `/tftp/pxelinux.cfg/` a při instalaci je přesuneme

na řídící server do adresáře `/srv/tftp/pxelinux.cfg/.` Tyto soubory mohou být zdrojem pro symbolické odkazy pro specifické stroje dle hardwarových adres, jak je popsáno v kapitole 2.5.3.1. V těchto souborech je vždy pouze jeden záznam spouštějící požadovaný krok automatické instalace.

Pro ověření správného nastavení můžeme nastartovat jeden z počítačů v dosahu správně nastaveného DHCP serveru a při startu pomocí vyvolání boot menu vybrat zavedení ze sítě. Mělo by zobrazit menu zavaděče, ovšem v tuto chvíli bude funkční pouze první možnost lokálního bootu.

3.3 Příprava automatické instalace GNU/Linuxu

Pro zprovoznění a přípravu prostředí umožňujícího využití nástroje FAI můžeme nainstalovat sadu potřebných balíků, jako je NFS server, konfigurátor FAI a další nebo použít připravený metabalíček `fai-quickstart.` V dokumentaci systému je doporučený použít repozitář projektu, aby byla zajištěna dostupnost nejnovější stabilní verze a z tohoto repozitáře nainstalovat potřebný balíček. Následující operace provádíme na řídícím serveru.

```
wget -O - https://fai-project.org/download/074BCDE4.asc | apt-key
add -
echo "deb http://fai-project.org/download stretch koeln" > /etc/apt/
sources.list.d/fai.list
apt-get update
apt-get install fai-quickstart
```

Před spuštěním inicializace je nutné ještě provést následující nastavení, konkrétně povolení zdroje balíčků pro inicializaci a nastavení uživatele pro logování průběhu instalace.

```
sed -i -e 's/^#deb/deb/' /etc/fai/apt/sources.list
sed -i -e 's/#LOGUSER/LOGUSER/' /etc/fai/fai.conf
```

Po těchto přípravách je již možné spustit inicializační program, který se postará o vytvoření NFS rootu, konfigurací exportů a bootstrap instalačního systému. Abychom mohli sledovat průběh, spustíme program v režimu se zvýšeným výpisem informací

```
fai-setup -v
```

Některé starší verze FAI nebyly připraveny na NFSv4 a konfigurace exportů byla nekompletní, což způsobovalo nemožnost připojit NFS oddíl a spustit instalaci. Ve správném stavu bychom měli mít ve výpisu inicializace a následně i v souboru `/etc/exports` následující obsah

```
/srv/nfs4 147.228.63.0/24(fsid=0,async,ro,no_subtree_check)
/srv/fai/config 147.228.63.0/24(async,ro,no_subtree_check) /srv/fai/
nfsroot 147.228.63.0/24(async,ro,no_subtree_check,no_root_squash)
```

Kontrolu běhu NFS serveru, který jsme v této fázi nainstalovali a nastavili dle potřeb provedeme pomocí `systemctl.`

```
systemctl status nfs-kernel-server
```

Očekávaný výstup:

```
● nfs-server.service - NFS server and services
   Loaded: loaded (/lib/systemd/system/nfs-server.service; enabled;
vendor preset: enabled)
   Active: active (exited) since Fri 2019-06-28 13:54:17 CEST; 3
weeks 6 days ago
```

Pokud nemáme vlastní konfiguraci, je možné jako základ našeho systému použít vzorové konfigurace dodávané s instalací. Tuto konfiguraci stačí přesunout do správného umístěni.

```
cp /srv/tftp/fai/initrd* /srv/tftp/
cp /srv/tftp/fai/vmlinuz* /srv/tftp/
cp -a /usr/share/doc/fai-doc/examples/simple/* /srv/fai/config/
```

Tímto jsme zprovoznili minimalistickou instanci nástroje FAI a můžeme spouštět základní instalace a využívat předdefinované třídy. Nyní potřebujeme přidat záznam do PXE menu, který nám zajistí spuštění instalačního procesu. Inicializace FAI nám umístí potřebné soubory do TFTP podadresáře FAI, z tohoto umístění je můžeme použít v záznamu pro PXE menu. Krom jádra a initrd obrazu musíme specifikovat root filesystém, což je v našem případě NFS úložiště s overlay systémem a umístění konfiguračních souborů.

```
LABEL FAI
     MENU LABEL FAI
     KERNEL fai/vmlinuz-4.9.0-9-amd64 APPEND initrd=fai/initrd.img-
4.9.0-9-amd64 ip=dhcp root=147.228.63.9:/srv/fai/nfsroot:vers=3
rootovl FAI_FLAGS=verbose,sshd,createvt,reset
FAI_CONFIG_SRC=nfs://147.228.63.9/srv/fai/config FAI_ACTION=install
```

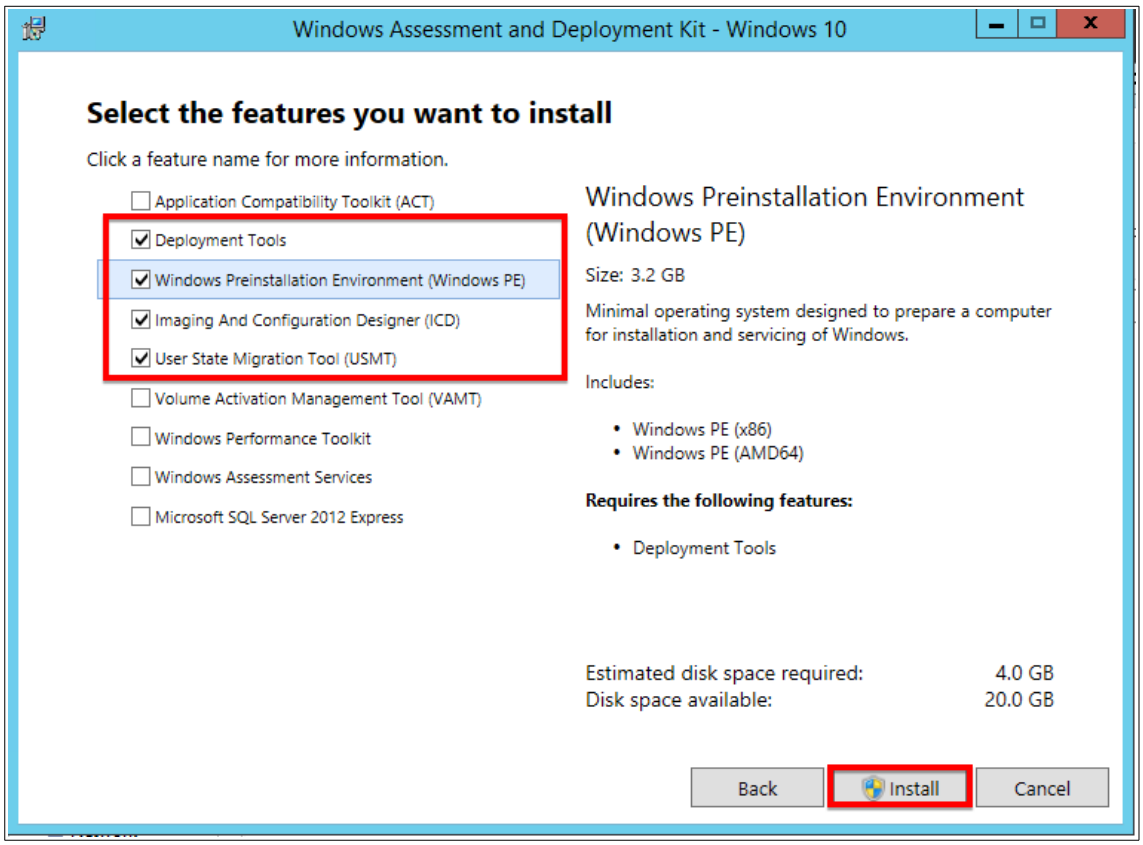
Takto připravenou sekci můžeme vložit do výchozí konfigurace PXE umístěné v souboru `/srv/tftp/pxelinux.cfg/default`. Pro otestování můžeme na testovacím stroji vybrat tuto volbu a spustit instalaci. **POZOR:** Ve výchozí stavu dojde ke smazání všech dat na cílovém stroji **!!!**

3.4 Příprava automatické instalace MS Windows

Pomocí nástrojů Windows ADK, konkrétně Windows SIM můžeme vytvořit soubor s kroky automatické instalace, který následně použijeme. Pomocí volitelného rozšíření Windows PE můžeme do Windows ADK přidat i možnost tvorby Windows PE prostředí pro start ze sítě. Pro kroky v této kapitole je nutné mít k dispozici stroj s operačním systémem MS Windows, na kterém budou operace probíhat. Předpokládá se desktopová verze systému MS Windows 10. Tvorbu Windows PE prostředí je možné provádět i z prostředí GNU/Linuxu ovšem pro zachování jednotnosti postupu budou veškeré kroky probíhat na počítači se systémem MS Windows. Tento počítač může být i virtuálním a je potřeba pouze pro přípravu potřebných souborů.

3.4.1 Instalace potřebných nástrojů

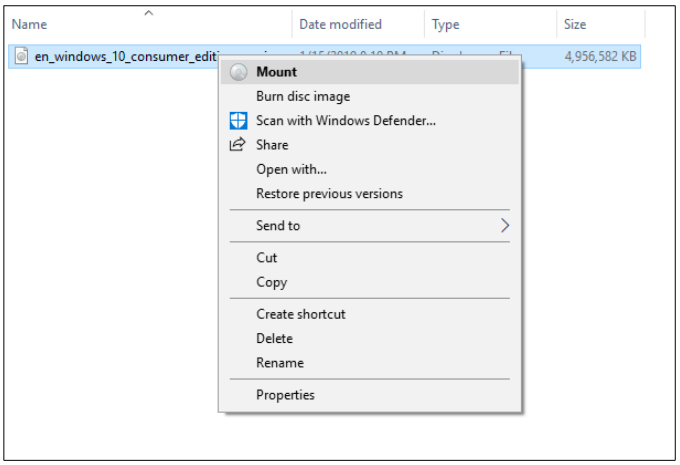
Instalátor sady nástrojů Windows ADK je z dispozici na oficiálních stránkách společnosti Microsoft [66]. Při instalaci musíme vybrat volbu Deployment tools zahrnující potřebné nástroje jak je zobrazeno na Obr 3.3 . Pomocí programů z tohoto balíčku následně budeme tvořit konfigurační soubory pro automatickou instalaci. Pro vytvoření konfigurací a přípravu síťové instalace musíme mít k dispozici instalační médium požadované verze systému. Dostatečná je forma obrazu disku, fyzické médium není podmínkou. Pokud nemáme médium nebo soubor k dispozici, můžeme si ho stáhnout na stránkách firmy Microsoft [67].



Obr 3.3: Výběr voleb pro instalaci Windows ADK

3.4.2 Tvorba konfigurace pro automatickou instalaci

Pro proces automatické instalace MS Windows potřebujeme konfigurační soubor, kterým je instalace řízena. Jedná se o předpis, podle kterého provede instalátor jednotlivé kroky a případně může volat definované skripty. Pro tvorbu souboru s kroky automatické instalace potřebujeme soubory z instalačního media MS Windows. Obraz instalačního disku, získaný například dle odkazu v kapitole 3.4.1, můžeme pomocí Průzkumníka připoj, pomocí položky v kontextové nabídce po stisku pravého tlačítka jak je zobrazeno na Obr 3.4 , jako jednotku a následně zkopírovat její obsah do vybrané lokální složky. Pro další postup budeme používat adresář **`C:\autoinstall\`**. Ve složce `C:\autoinstall` zkontrolujeme přítomnost klíčového souboru `install.wim`, který je nutný pro další postup. V případě použití vlastního obrazu disku může být na instalačním mediu přítomný soubor `image.esd`, který však není pro tento postup použitelný a je nutné stáhnout originální instalační medium z odkazu výše.



Obr 3.4: Připojení obrazu disku jako jednotky

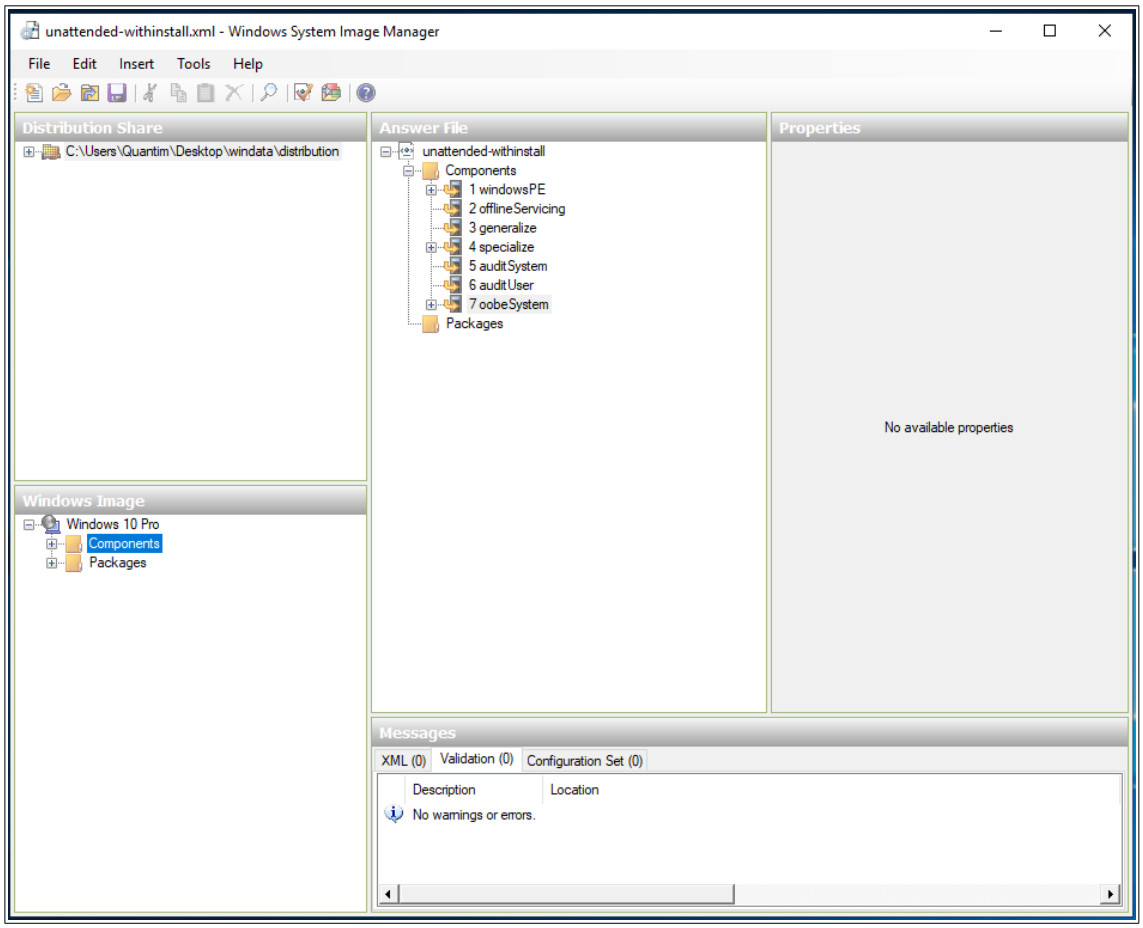
Konfigurační soubor pro automatickou instalaci bude vytvářen v nástroji Windows System Image Manager - Windows SIM, který je součástí instalace Windows ADK provedeném v předchozím kroku. Nástroj spustíme z nabídky start, kde je po instalace dle předchozí kapitoly umístěn zástupce.

Po spuštění programu musíme nejprve přes menu **File->Select Windows Image** vybrat umístění souboru `C:\autoinstall\install.wim` instalační médium může obsahovat více edicí systému. V tom případě vybereme, pro kterou edici připravujeme konfiguraci.

Pokud spouštíme nástroj prvně, nebo jsme vybrali novou edici, zobrazí se dotaz na vytvoření souboru s katalogem, který potvrdíme a bude nám vytvořen potřebný soubor s příponou `.clg` uložený ve stejném adresáři, jako instalační soubory, tudíž `C:\autoinstall\`.

Dalším krokem je tvorba pracovního adresáře s umístěním podpůrných souborů a nastavení využívaných při tvorbě konfigurace. Jedná se ve své podstatě o projektový pracovní prostor známý z vývojových prostředí. Složku vybereme pomocí z nabídky **File->Select Distribution Share** ,a zvolíme (v případě absence vytvoříme) `C:\autoinstall\distribution`

Posledním krokem je otevření prázdného konfiguračního souboru. K tomuto účelu slouží nabídka **File->New Answer File**. Tím máme připravené pracovní prostředí, které by mělo vypadat obdobně jako na Obr 3.5.



Obr 3.5: Windows SIM

V podokně Answer file vidíme 7 fází instalačního procesu, Pro náš účel jsou relevantní fáze 1 Windows PE, fáze 4 specialize a fáze 7 oobeSystem. Tyto fáze budou popsány v následujících kapitolách. Do jednotlivých fází přidáváme komponenty dostupné pro danou edici systému. Výběr komponent se provádí v sekci Windows Image v levém dolním rohu nástroje. Kompletní výpis komponent je k dispozici v dokumentaci [68]. Detailní postup instalace včetně snímků obrazovky pro jednotlivé kroky je vyobrazeny v příloze Příloha D

Veškeré cesty v nastavení automatické instalace jsou vzhledem k procesu instalace a nevztahují se k síťovému disku na počítači, kde se připravuje nastavení automatické instalace.

3.4.2.1 Nastavení sekce Windows PE

V této fázi dochází k nastavení jazyku instalace, operací s disky a vložení informace o licencích a sériovém čísle pro instalaci

Pro nastavení jazyka slouží komponenta **SetupUILanguage** nacházející se pod **amd64_Microsoft-Windows-International-Core-WinPE**. Provedeme zde nastavení dle preferencí. Vzhledem k bezobslužné instalaci není jazyk instalace natolik podstatný a hlavním účelem tohoto bodu je odstranění dialogu, který by se nám jinak zobrazil.

V další sekci je možné nastavit disky včetně jejich formátování a rozdělení. V našem scénáři očekáváme, že jsou disky již rozdělené a proto zde již nové rozdělení disku nebudeme provádět. V případě potřeby zde můžeme provést formát vybrané cílové oblasti, je však nutné dát si pozor, aby parametry komponenty disk **neobsahovaly Wipe: true**, což by vedlo ke smazání celého disku.

Pro určení cílové oblasti disku slouží komponenta **amd64_Microsoft-Windows-Setup→ImageInstall→OSImage→InstalTo** ve které určíme ID cílového disku. Disky jsou číslovány od nuly a nula odpovídá prvnímu disku detekovaným BIOSem.

Závěrečnou komponentou je **amd64_Microsoft-Windows-Setup → UserData → ProductKey** zde v sekci UserData vyjádříme souhlas s licenčním smlouvou nastavením volby **AcceptEula** na hodnotu true. V sekci ProductKey následně vložíme Klíč produktu, pokud nemáme vhodný klíč pro mnohonásobnou instalaci, můžeme v této fázi využít jeden z generických klíčů poskytovaný pro tyto účely společností Microsoft [69] .

3.4.2.2 Sekce přizpůsobení systému

Fáze specializace slouží k úpravě systému a případně instalaci požadovaných balíků. V tomto scénáři ji použijeme pro instalaci doplňujícího software. Ke spouštění úloh máme k dispozici synchronní a asynchronní úlohu, kdy u synchronní úlohy se čeká na její dokončení, zatímco po spuštění asynchronní se ihned po spuštění pokračuje další úlohou. Pro lepší kontrolu nad průběhem jsem zvolil synchronní úlohy z komponenty

amd64_Microsoft-Windows-Deployment_neutral → RunSynchronous → RunSynchronousCommand. Při použití více příkazů je potřeba určit jejich pořadí parametrem Order. Do cesty následně napíšeme potřebný skript.

V našem případě se jedná o powershell skript, který je stahován ze síťového úložiště. Toto úložiště je potřeba během instalace nejprve připojit jako síťový disk, aby bylo možno v této fázi skript volat. Toho docílíme volbou prvního **RunSynchronousCommand** příkau na

```
net use S: \\147.228.63.9\data
```

Díky umístění vlastního skriptu na sítí můžeme měnit jeho chování bez nutnosti zasahovat do vlastní instalace. Vzhledem k získávání skriptů ze sítě musíme přenastavit chování powershellu, který standardně akceptuje pouze podepsané skripty, aby spouštěl i námi vytvořený skript bez nutnosti podepisování. Výsledný příkaz, který přidáme do volby **Path** v konfiguraci pak vypadá následovně.

```
powershell.exe -ExecutionPolicy Bypass S:\scripts\default\
install.ps1
```

Samotný vzdáleně spouštěný skript prohledává síťový adresář a postupně spouští jednotlivé nalezené instalátory v tichém módu

`install.ps1`

```
$FILES = Get-ChildItem Z:\packages -File
$FILES = $FILES.name | Select-String -Pattern msi
foreach ($FILE in $FILES)
{
     Start-Process msiexec.exe -Wait -ArgumentList "/i
     Z:\packages\$FILE /quiet"
}
```

Využitím tohoto postupu můžeme vytvořit několik předpisů pro automatickou instalaci, které se budou lišit adresářem, ze kterého budou stahovat instalátory. Obsahem složek pak můžeme ovlivňovat výsledný stav připravovaného stroje. Instalace a zpřístupnění skriptů bude řešeno v kapitole 3.4.4

3.4.2.3 Operace před prvním spuštěním systému

V závěrečné sedmé fázi procesu se provádí už jen drobná nastavení, jako je volba jazykových předvoleb, skrytí obrazovek s licencí, nabídka registrace a další fáze, kterými se běžně musí procházet při prvním zapnutí systému. Můžeme zde také vytvořit lokálního uživatele v komponentě **amd64_Microsoft-Windows-Shell-Setup_neutral → UserAccounts → LocalAccounts** pro provádění správcovských úkolů a zadat mu heslo. Zde ačkoliv je možnost heslo skrýt, tak **heslo** ve výsledném souboru **NENÍ žádným způsobem šifrováno.** Jedná se pouze o převod do formátu BASE64 a tudíž je volně čitelné!

3.4.2.4 Uložení konfigurace

Po dokončení veškerých požadovaných nastaví uložíme výsledný soubor s konfigurací pomocí položky menu **File → Save As…** do adresáře, ve kterém se nachází instalační soubory rozbalené v kapitole 3.4.2. Jedná se o soubor ve formátu XML [70]. Pro další příklady je používáno jméno `unattended.xml`.

3.4.3 Tvorba Windows PE obrazu

Pro vytvoření obrazu Windows PE, která bude načten síťovým zavaděčem a ze které bude spuštěna vlastní automatická instalace. Obraz vytvoříme nástrojem **Deployment and Imaging Tools Environment** z nabídky start. Nástroj je součástí dříve nainstalovaného nástroje Windows ADK Deployment and Imaging Tools Environment je potřeba spustit jako Administrátor. Nástroj se spustí jako okno s příkazovou řádkou, do které zadáváme příkazy uvedené níže.

V otevřené příkazové řádce nástroje Deployment and Imaging Tools Environment vytvoříme adresář s pracovním prostředí pro následnou tvorbu obrazu

```
copype amd64 C:\WinPE_amd64
```

Po doběhnutí příkazu budeme mít v uvedeném adresáři vytvořenou potřebnou adresářovou strukturu a umístěny soubory pro tvorbu Windows PE prostředí.

3.4.3.1 Spouštěcí skript Windows PE

Jedním z parametrů při tvorbě obrazu se systémem Windows PE je skript, který se provede po nastartování Windows PE prostředí. Zde můžeme spustit například

příkazovou řádku `cmd.exe`, nebo jako v našem případě připravit prostředí pro automatickou instalaci a tu následně spustit. Příkazy prováděné při spuštění se definují v obrazu `boot.wim`, který je nutná připojit pomocí příkazu `Dism,` abychom mohli v tomto obrazu provádět změny.

```
Dism /Mount-Image /ImageFile:"C:\WinPE_amd64\media\sources\boot.wim"
/index:1 /MountDir:"C:\WinPE_amd64\mount"
```

Následně v adresáři s rozbaleným obrazem provedeme úpravu souboru `C:\WinPE_amd64\mount\Windows\System32\Startnet.cmd.` Tento soubor se faktický volá až při vlastní instalaci na cílovém počítači zde je přítomen pouze jako předpis, který se připojí do vytvářeného Windows PE prostředí a žádné z příkazů nejsou v tuto chvíli volány.

`Startnet.cmd`

```
wpeinit.exe
ping 127.0.0.1 -n 60 > null
net use s: \\147.228.63.9\data /user:foo pass:bar
s:
setup.exe /unattend:unattended.xml
```

Vzhledem k tomu že ne všechny příkazy jsou intuitivní, následuje jejich popis. `wpeinit.exe` provede inicializaci všech Plug & Play zařízení, což potřebujeme pro zpřístupnění sítě a načtení instalačních souborů se síťového úložiště.

Příkaz `ping 127.0.0.1 -n 61 > null` slouží jako náhrada neexistujícího `sleep` v příkazovém řádku. Využívá se zde výchozího intervalu mezi jednotlivými voláními, který je 1s. Tímto zpožděním dáme dostatek času síťovému adaptéru získat a nakonfigurovat adresu ze sítě. Řádek je důležitý především v rozsáhlých sítích, kde je nutná delší doba na získání IP adresy z důvodu komplexní topologie.

Připojení síťového složky jako síťového disku s písmenem `s:` se provede příkazem `net use s: \\147.228.63.9\data /user:foo pass:bar` Za povšimnutí zde stojí přihlašovací údaje. Ačkoliv jsme nastavili síťovou složku na povolení přístupu nepřihlášeným uživatelům, bez zadání uživatelského jména a hesla

tento příkaz zhavaruje a vypíše chybu. Je proto potřeba zadat uživatelské jméno a heslo, i když pak není při připojení kontrolováno a nezáleží na hodnotě těchto parametrů.

Zbylé dva řádky nás už přepnou do nově připojeného umístění a následně spustí instalátor, kterému jako parametr předáme soubor s předvolbami pro automatickou instalaci, který jsme vytvořili v předchozí kapitole 3.4.2.

Po uložení souboru je ještě nutné ukončit editování souboru `boot.wim`

```
Dism /Unmount-Image /MountDir:"C:\WinPE_amd64\mount" /commit
```

3.4.3.2 Vlastního tvorba obrazu

S upraveným spouštěcím skriptem a již můžeme přistoupit k vytvoření vlastního obrazu. Výsledkem je `iso` soubor.

```
MakeWinPEMedia /ISO C:\WinPE_amd64 C:\WinPE_amd64\winpe.iso
```

Vytvořený soubor přeneseme na řídící server a umísíme ho do adresáře s PXE zavaděčem. V našem případě do adresáře `/srv/tftp.`

3.4.4 Příprava síťového úložiště SMB

Následující operace jsou prováděny na řídícím serveru.

Samba server server se nachází ve stejnojmenném balíčku, ze kterého provedeme jeho instalaci

```
apt-get install samba
```

V konfiguraci povolíme přístup do námi zvolené složky s instalačními a konfiguračními soubory. Složku s těmito soubory umístíme jako podadresář `samba` do adresáře `/srv`. Samotný konfigurační soubor se jmenuje `/etc/samba/smb.conf` a doplněná sekce povolující sdílení složky vypadá následovně

`/etc/samba/smb.conf`

```
[data]
     path = /srv/samba
     read only = yes
     guest ok = yes
     create mask = 744
     directory mask = 755
```

Ke složce povolujeme přístup bez přihlášení, což v případě použití na lokální síti to nemusí být závažným problémem a umožní nám to následné použití tohoto úložiště pro dynamické načítání přizpůsobovacích skriptů. Složka je také nastavena do read-only módu.

Do složky `/srv/samba` přeneseme z počítače, na kterém jsme připravovali konfiguraci adresář `C:\autoinstall\` s rozbaleným obrazem instalačního média včetně souboru `unattended.xml` pro řízení bezzásahové instalace vytvořený v kapitole 3.4.2. Z přiloženého CD zkopírujeme adresář `/samba/scripts/` obsahující skripty používané pro přizpůsobení instalace jako `install.ps1` popsaný výše.

Následně zbývá restart samby, aby se načetla nová konfigurace a ověření správného spuštění příkazem `systemctl`, kde očekáváme status `active (running)`.

```
systemctl restart smbd
systemctl status smbd
```

3.4.5 Položka PXE menu

Abychom mohli spouštět Windows PE prostředí pomocí PXE zavaděče, potřebujeme do menu přidat další položku. Při jejím výběru dojde k načtení Windows PE `iso` obrazu, jeho připojení a startu, jako by se jednalo o lokálně připojené médium. Toto chování nám umožní minimalistický kernel `memdisk`, který je součástí balíčku `syslinux-common`. Jádru ještě předáme jako parametry příkazy `iso` a `raw`, které signalizují, aby se k obrazu choval jako k `iso` obrazu a neprováděl žádné optimalizace, které by mohly ovlivnit načítání.

```
LABEL winpe
     MENU LABEL Windows PE Installation
     KERNEL /memdisk
     INITRD winpe.iso
     APPEND iso raw
```

Správnost konfigurace můžeme ověřit ručním spuštěním nově přidané položky PXE menu, kdy nám naběhne prostředí Windows PE a po připojení síťového adresáře se spustí instalace.

3.4.6 Tvorba instalačních profilů

Pro vytvoření nového instalačního profilu musíme nejprve vytvořit novou konfiguraci v nástroji Windows SIM a výsledný XML soubor uložit pod novým jménem. V tomto souboru můžeme například zvolit jinou složku, ze které budeme instalovat požadovaný software. Musíme také vytvořit nový Windows PE soubor dle postupu v kapitole 3.4.3, kterému jako parametr instalátoru předáme nově vytvořený XML soubor.

Vytvořený Windows PE soubor i XML soubor s konfigurací instalace přesuneme na řídící server a pro jejich použití přidáme další položku do PXE menu spouštějící nový Windows PE obraz, případně vytvoříme separátní konfigurační soubor dle jména profilu.

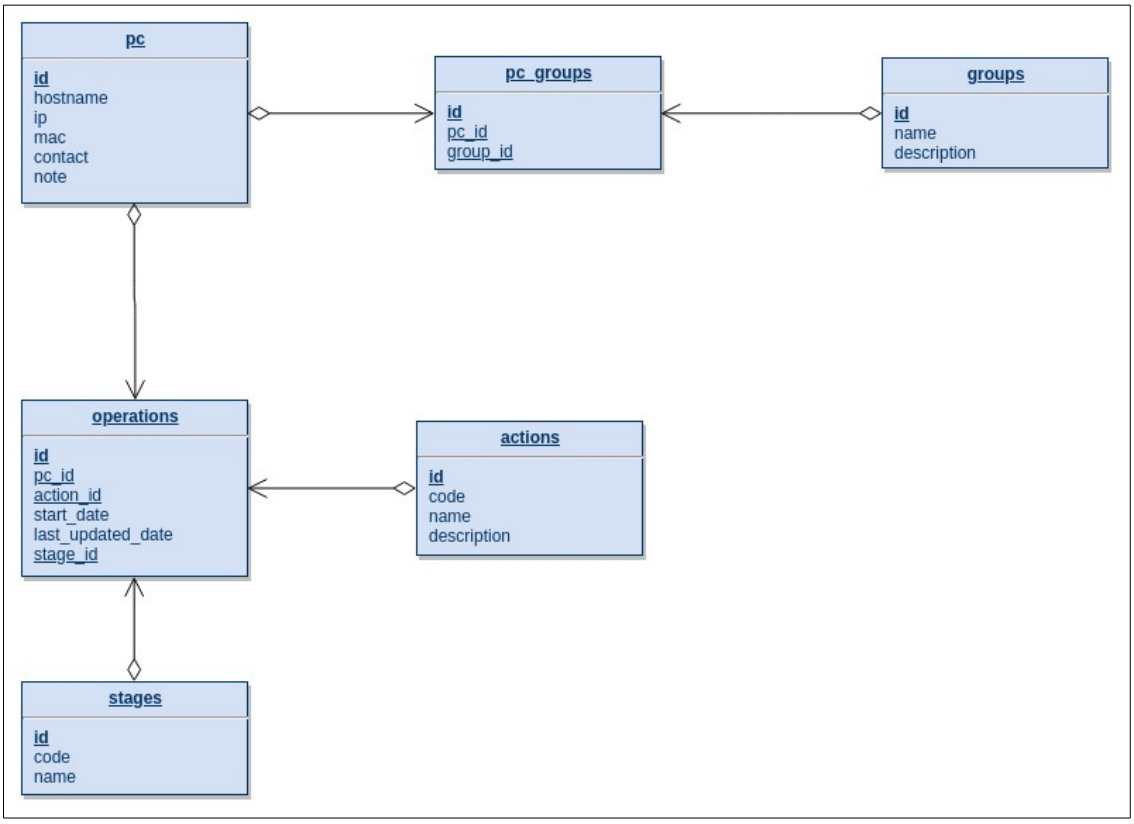
3.5 Systém pro správu instalací

Aby bylo možné spravovat a monitorovat instalace, bude vytvořen daemon pro řízení životního cyklu instalace. Dále bude vytvořena webová aplikace, se kterou bude daemon komunikovat přes frontu v relační databáze. V kapitolách 3.5.1, 3.5.2 a 3.5.3 jsou popsány jednotlivé komponenty, jejich nasazení na řídící server je pak popsáno v kapitole 3.5.4.

3.5.1 Datový model

Jako úložiště dat pro aplikaci jsem zvolil databázi MySQL, která poskytuje veškerou potřebnou funkcionalitu a dá se snadno napojit na webovou aplikaci i vytvářeného daemona. Zde byly na základě požadované funkčnosti navrženy tabulky

včetně vzájemných vazeb a podmínek pro zajištění konzistence ukládaných dat. Následuje seznam tabulek se základním popisem, detailní struktura je patrna z Obr 3.6.



Obr 3.6: Schéma databáze

- `pc` – informace o jednotlivých strojích, jejich jméno, IP a hardwarová adresa, model daného stroje a kontakt na osobu odpovědnou za tento stroj, typicky se jedná o osobu zodpovědnou za laboratoř
- `groups` – Skupina strojů s názvem a popisem dané skupiny.
- `pc_groups` – Vazební tabulka, počítač může být členem více skupin, například dle laboratoře a dle účelu.
- `actions` – Seznam akcí, které je možno s počítači a skupinami provádět, z těchto akci se generují funkční tlačítka ve webové aplikaci.
- `operations` – Log operací s informaci o prováděné operaci, vazbou na hosta a aktuálním stavu dané operace a času vytvoření a poslední změny stavu.

- `stages` – číselník stavů, používaný v tabulce operation pro sledování aktuálního stavu operace

3.5.2 Webová aplikace

Pro tvorbu webové aplikace jsem zvolil PHP [71] a framework Nette [72], který poskytuje veškerou požadovanou funkcionalitu a umožňuje rychlé vytvoření cílové aplikace. Pro vizuální elementy byla použita CSS [73] knihovna bootstrap [74] a jako zdroj ikon použitých v grafickém rozhraní slouží open-source kolekce ikon z projektu feather [75].

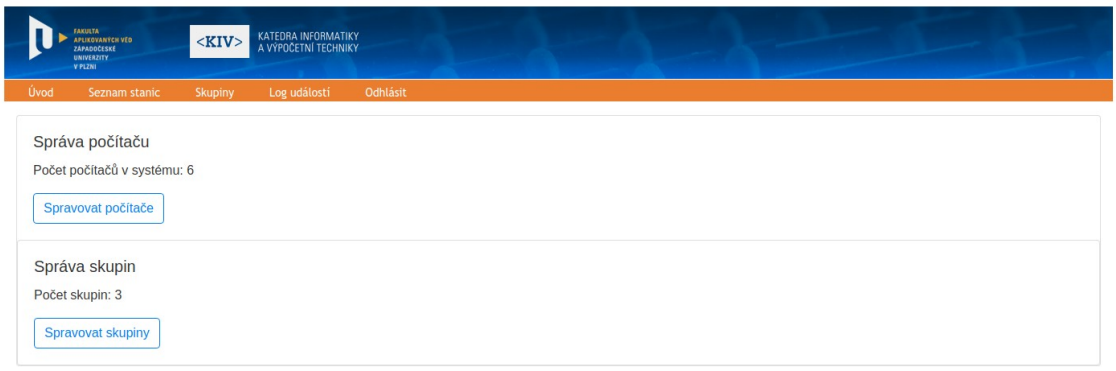
Nette framework má architekturu Model-View-Presenter, který je variantou architektury Model-View-Controller [76]. Toto rozdělení umožňuje oddělit interakci s databází do modelu, aplikační logiku do presenterů a zobrazená stránek do view vrstvy. V samotném frameworku jsou presentery reprezentovány PHP soubory v adresářové struktuře projektu. Soubory jsou uloženy v podadresáři `app/Presenters/` projektu. Na prezentery jsou navázány jednotlivé šablony, pomocí kterých se zobrazují získaná data.

Aplikace obsahuje několik hlavních presenterů, které se starají o práci z daty a následné plnění šablon zobrazovaných uživateli při jednotlivých akcích. Jména souborů jsou shodná s názvy presenterů doplněná o příponu `.php`. Ke každému prezenteru jsou vytvořeny šablony pro vytvoření stránky se zobrazením dat. Soubory se šablonami se nacházejí v podadresářích `app/Presenters/templates/<jmeno_presenteru>/`

- **HomepagePresenter** – Presenter pro domovskou stránku aplikace, obstarává získávání základních přehledových dat o systému, které používá výchozí šablona Domovská stránka pro přihlášení je zobrazena na Obr 3.7

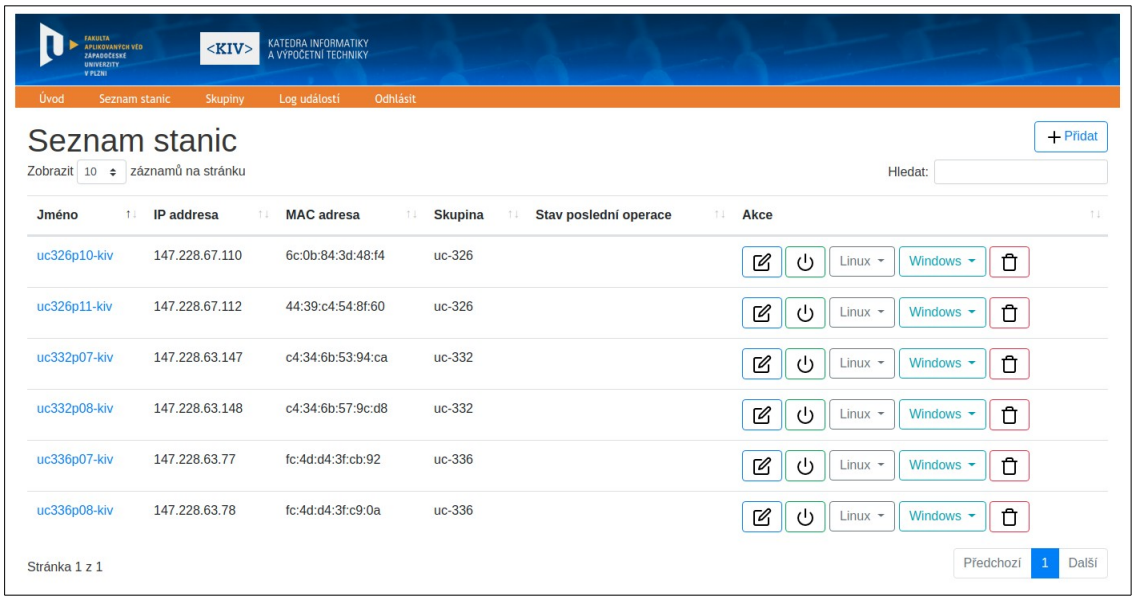
  Navázané šablony

  - default – Výchozí šablona zobrazující buď výzvu k přihlášení, nebo základní přehledová data o systému získaná z presenteru.



Obr 3.7: Domovská stránka webového rozhraní

- **PcPresenter** – Obstarává data o seznamu počítačů, tvorbu formuláře pro přidávání a úpravu počítačů, jeho zpracování a uložení do databáze a mazání. Je zde také funkce odchytávající stisk tlačítek s akcí, která zařídí vytvoření záznamu pro nový úkol v příslušné tabulce. Ukázka seznamu je na Obr 3.8



Obr 3.8: Seznam stanic

Navázané šablony

- show – šablona zobrazující seznam všech počítačů

- edit – šablona s formulářem, používaná pro tvorbu a úpravy záznamů o jednotlivých počítačích.

- **GoupPresenter** – Práce se informacemi o skupinách, počet počítačů spadajících pod skupiny, obsluha skupinových akcí jako hromadné zapnutí a hromadné instalace.

  Navázané šablony

  - show – výpis tabulky skupin s možností vyvolat akce nebo se prokliknout na členské počítače.

- **LogPresenter** – získávání informací o práve probíhajících operacích nad vybraným strojem nebo skupinou

  Navázané šablony

  - show – zobrazení seznamu operacích vztahujících se k vybranému prvku včetně informace o stavu a čase poslední změny.

- **SignPresenter** – obsluha přihlašování. Ověřování uživatelského jména a hesla.

  Navázané šablony

  - show – přihlašovací formulář

- **ApiPresenter** – obsluha externích událostí ze skriptů, jako například ohlášení konce instalace

  Navázané šablony

  - beacon – endpoint na ohlášení konce instalace, pro identifikaci jsou předávány dostupné hardwarové adresy na cílovém stroji

3.5.3 Obslužný daemon

Pro řízení instalací byl vytvořen obslužná daemon `backend.sh`. Tento daemon periodicky sleduje databázovou tabulku `operations` a vykonává naplánované akce. Pro probuzení zajistí zavolání nástroje `etherwake`, který pošle magic paket na síťovou adresu počítače. Skriptovací jazyk BASH byl použit pro snadné operace nad

soubory nutnými pro přípravu instalačních konfigurací a volání dalších příkazů systému.

Daemon obsahují hlavní smyčku, které čeká na novou práci. Pro úkoly slouží databázová tabulka `operations`. Při načtení nového úkolu ve stavu, který umožňuje provedení akce je tento řádek načten a dle parametrů je provedena požadovaná akce. V současné době jsou podporované akce `POWERON`, `INSTALL_LINUX` a `INSTALL_WINDOWS`, kdy instalační akce podporují ještě upřesnění profilu. Pokud profil není uveden, nebo je nastaven na hodnotu `default`, provádí se výchozí instalace.

Spuštění počítače spočívá v zavolání nástroje `etherwake` popsaného v kapitole 3.2.1 s hardwarovou adresou síťové karty získanou z databáze. Po zavolání tohoto příkazu je spuštěna asynchronní funkce, které kontroluje, zda se spuštěný počítač probudil. Asynchronní funkce je řešená zavolání BASH funkce na pozadí pomocí operátoru &. Pokud nedojde v nastaveném časové intervalu k probuzení stroje, pokusí se funkce tento pokus ještě dvakrát zopakovat a při opakovaném neúspěchu je dané operaci nastaven stav indikující, že skončila s chybou.

Při požadavku na instalaci zajišťuje daemon přípravu PXE konfigurací v definovaném tvaru pro konkrétní počítač. Operace spočívá ve vytvoření symbolického odkazu na konfiguraci dle zvoleného profilu instalace. S připravenou konfigurací počítač spustí, ověří se naběhnutí počítače dle popisu v předchozím odstavci. Pokud spuštění proběhne v pořádku, zavolá se další asynchronní funkce kontrolující, zda byla instalace dokončena. Po uplynutí časového limitu pro instalaci opět nastaví chybový stav. Zde pro tuto chvíli končí zodpovědnost tohoto daemona, samotné ukončení instalace je ohlášeno instalovaným počítačem pomocí zavolání API endpointu webové aplikace popsaného v kapitole 3.5.2. Po ohlášení ukončení je daemonem proveden úklid vytvořených konfigurací pro danou instalaci a označení celé instalace za hotovou.

Nastavení daemona se provádí v úvodní části souboru `backend.sh`, kde se nastavují potřebné údaje jako je přístup do databáze a jména síťových rozhraní.

```
#!/bin/bash

declare -A INTERFACE=()

INTERFACE[63]="ens18"
INTERFACE[67]="ens19"
PXE_CONFIG_DIR="/srv/tftp/pxelinux.cfg"
MYSQL_USER="installation"
MYSQL_PASSWORD="secret-pass"
MYSQL_DB_NAME="installations"
POWERON_RETRY=3
POWERON_TIMEOUT=60  # in seconds
INSTALL_TIMEOUT=900 # in seconds
```

Aby byl zajištěno automatické běh deamona, je využíván `systemd`, zodpovědný za spouštění a restart daemona v případě jeho pádu.

`installations-backend.service`

```
[Unit]
Description=Installations daemon
After=mysql.service


[Service]
ExecStart=/usr/local/bin/backend.sh
Restart=on-failure

[Install]
WantedBy=default.target
```

Výsledky operací, případně následné kroky v postupu instalace se zaznamenávají zpět do databázové tabulky `actions,` odkud jsou zobrazovány ve webovém rozhraní a je tak možno monitorovat průběh prováděných operací.

3.5.4 Nasazení

Pro nasazení potřebujeme na cílovém serveru, může se jednat i o řídící server, nainstalovat webový server, databázi a podporu PHP s potřebnými moduly, vše musí být také odpovídajícím způsobem nastaveno. Nejprve provedeme nasazení MySQL databáze jako úložiště dat, následně proběhne nasazení obslužného daemona a nakonec bude nasazen webový server s aplikací.

3.5.4.1 Instalace databázového serveru

Pro nainstalování databáze zvolíme metabalíček `mysql-server` zajišťující, že budeme mít nasazenou aktuální verzi databázového serveru

```
apt-get install mysql-server
```

Po instalaci se přihlásíme do databáze, a zde vytvoříme novou databázi, uživatele pro práci s touto databází, přiřadíme mu práva a v posledním kroku importujeme schéma používané aplikací. Schéma je dostupné na přiloženém CD v cestě `/webapp/schema.sql`

```
mysql -u root
CREATE DATABASE installations;
CREATE USER `installations`@`localhost` IDENTIFIED BY 'secret-pass';
GRANT ALL PRIVILEGES ON installations.* TO
`installatons`@`localhost`;
FLUSH PRIVILEGES;
QUIT;
mysql -u root installations < schema.sql
```

Pro zabezpečení databáze musíme nastavit heslo uživatele root, to provedeme následujícím příkazem, kde míst NEWPASSWORD nastavíme dostatečně bezpečné heslo

```
mysqladmin -u root password NEWPASSWORD
```

3.5.4.2 Nasazení obslužného deamona

Daemon pro svou funkci využívá program arping, který musíme nejprve nainstalovat.

```
apt-get install arping
```

Soubor backend.sh z přiloženého CD, adresáře `/daemon,` umístíme do adresáře `/usr/local/bin/` na řídícím serveru. Definici systemd služby ze stejného zdrojového adresáře na CD jménem `installations-backend.service` nahrajeme do adresáře `/etc/systemd/system/`. Následně načteme znovu definici systemd služeb, spuštíme daemona a ověříme zda služba naběhla. Výstupem posledního příkazu by měl být stav `active (running).`

```
systemctl daemon-reload
systemctl start installations-backend
systemctl status installations-backend
```

Protože skript obsahuje přístupové údaje do databáze, neměl by být čitelný pro běžné uživatele, proto bychom měli zmenit jeho vlastníka na uživatele root a odebrat práva skupině a ostatním

```
chown root:root /usr/local/bin/backup.sh
chmod 500 /usr/local/bin/backup.sh
```

3.5.4.3 Instalace PHP

Pro běh webové aplikace je potřebné mít na serveru nainstalované PHP. Aplikace byla psána na aktuální podporované verzi PHP 7.2, která není v naší distribuci dostupná. Proto použijeme repozitář deb.sury.org [77] spravovaný stejným správcem, který pečuje i o distribuční balíčky PHP.

```
apt-get -y install apt-transport-https lsb-release ca-certificates
wget -O /etc/apt/trusted.gpg.d/php.gpg
https://packages.sury.org/php/apt.gpg
sh -c 'echo "deb https://packages.sury.org/php/ $(lsb_release -sc)
main" > /etc/apt/sources.list.d/php.list'
apt-get update
```

Tímto jsme získali k dispozici aktuální repozitář a můžeme nainstalovat PHP a moduly potřebné pro běh aplikace. Jedná se o modul pro práci s relační databází a pro podporu databáze sqlite využívané frameworkem.

```
apt-get install php7.2 php7.2-mysql php7.2-sqlite
```

Stav instalace ověříme pomocí zobrazení verze PHP příkazem níže, jehož výstupem je verze nainstalovaného PHP, která musí být minimálně 7.2

```
php -v
```

3.5.4.4 Instalace webové serveru Apache

Pro zpřístupnění stránek je potřeba mít na serveru nainstalovaný webový server zajišťující interpretaci PHP kódu a zpřístupnění výsledných stránek uživatelům. Společně s webovým serverem nainstalujeme i modul pro obsluhu php skrptů

```
apt-get install apache2 libapache2-mod-php
```

Adresářová struktura projektu Nette očekává, že kořenová složka webové stránky bude směřovat do podsložky `www` v adresáři s aplikací. Abychom zachovali konvence, umístíme aplikaci do podadresáře v `/var/www/html/` zachovávající jméno aplikace. a upravíme konfiguraci serveru, aby hledal obsah na správném umístění. Vytvoříme definici nové stránky v adresáři `/etc/apache2/sites-enabled/` zvolený soubor musí mít příponu `.conf`. Certifikáty zmíněné v direktivách začínajících `SSLCertificate` získáme postupem popsaným v kapitole 3.5.4.6.

`/etc/apache2/sites-enabled/webinterface.conf`

```
<VirtualHost *:80>
  Redirect permanent / https://labs.kiv.zcu.cz/
</VirtualHost>
<VirtualHost *:443>
  DocumentRoot /var/www/html/webinterface/www
  SSLEngine on
  SSLCertificateFile
     /etc/letsencrypt/certs/live/labs.kiv.zcu.cz.pem
  SSLCertificateKeyFile
/etc/letsencrpyt/certs/private/live/labs.kiv.zcu.cz.key
  <Directory /var/www/html/webinterface/www>
    Options Indexes FollowSymLinks
    AllowOverride All
    Require all granted
  </Directory>
</VirtualHost>
```

Nyní odstraníme z aktivních stránek výchozí konfiguraci webserveru, aktivujeme námi nové vytvořenou a ještě povolíme modul pro podporu souborů `.htaccess`. Tato podpora je vyžadována použitým frameworkem.

```
a2dissite 000-default
a2ensite webinterface
a2enmod rewrite
```

Aby se změny projevily, musíme restartovat server, a následně ověříme, že po změně konfigurace naběhl, kontrolu provedeme příkazem `systemctl`, kde očekáváným výsledkem je stav `active (running)` případně navgací na adresu serveru v prohlížeči.

```
systemctl restart apache2
systemctl status apache2
```

3.5.4.5 Nasazení webové aplikace

Nasazení aplikace spočívá v překopírování obsahu složky `/webapp` z přiloženého CD do složky `/var/www/html/webinterface/` a upravení informací pro připojení k databázi. Tyto informace se nacházejí v souboru `app/config/local.neon`. Ukázková konfigurace následuje.

`app/config/local.neon`

```
database:
  dsn: 'mysql:host=127.0.0.1;dbname=installations'
  user: installations
  password: secret-pass
```

3.5.4.6 Zabezpečení webové aplikace

Pro možnost využití HTTPS potřebujeme vygenerovat TLS certifikát, k tomuto účelu využijeme nástroj certbot [78]. od certifikační autority Let‘s Encrypt [79], který nejprve nainstalujeme.

`/etc/apt/source.list.d/certbot.conf`

```
deb http://deb.debian.org/debian stretch-backports main
```

```
apt-get install certbot python-certbot-apache -t stretch-backports
certbot certonly --apache -d labs.kiv.zcu.cz
```

Vlastní přístup do aplikace je chráněn uživatelským jménem a heslem. Heslo musíme nastavit v konfiguraci webové aplikace

`/var/www/html/app/config/local.neon`

```
    security:
      users: diplomka: diplomka
```

Přístup do aplikace omezíme pouze na adresy univerzity pomocí nástroje `iptables` [80] případně je možno použít specifičtější rozsah.

```
iptables -A INPUT -p tcp -s 147.228.0.0/16 --dport 80 -j ACCEPT
iptables -A INPUT -p tcp -s 147.228.0.0/16 --dport 443 -j ACCEPT
```

3.6 Testování

V prvotních fázích vývoje probíhalo testování lokálně pomocí virtuálních strojů. V pokročilé fázi realizace se přistoupilo také na testy v reálném prostředí laboratoří. Během těchto testů se podařilo odhalit drobné odlišnosti mezi virtuálními stroji a fyzickou infrastrukturou, jako například delší doba nutná pro získání adresy v reálné infrastruktuře. Tyto problémy byly následně odladěny tento konkrétní byl vyřešen přidáním čekání do posloupnosti úkolů při instalaci MS Windows popsané v kapitole 3.4.3.1.

3.6.1 Lokální testovací prostředí

Pro lokální testování byl využíván nástroj VirtualBox [81] ve kterém jsem si vytvořil virtuální stroj s GNU/Linuxem dle popisu v kapitole 3.2. Tento stroj simuloval řídící server a postupně byly nasazovány jednotlivé komponenty řídícího serveru a docházelo k ladění konfigurace. Omezením virtuálního prostředí byla absence schopnosti Wake-on-Lan mezi virtuálními stroji. Lokální testování umožnilo zkrátit vývojový cyklus, protože nebylo nutné přenášet soubory na vzdálený server. Vzhledem k omezenému místu na testovacím stroji šlo lokálně vytvořit pouze jeden řídící server a dva klienty.

Pro snazší komunikaci a možnost vzdáleného připojování dostal Virtuální stroj simulující řídící server celkem 3 síťové karty.

- Karta s překladem adres – tato karta sloužila pro veškerou komunikaci z a do internetu, jako je například stahování instalovaných balíčků.
- Host only network – přípe propojení virtuálního stroje s hostitelským systémem. Díky tomuto spojení jsem se mohl na server připojovat pomocí

SSH, které nabízelo vyšší uživatelský komfort oproti virtuálnímu terminálu poskytovanému přímo VirtualBoxem. V pozdějších fázích vývoje přes toto rozhraní probíhalo testování webového rozhraní pro ovládání instalací.

- Interní síť – tato síť slouží pouze pro vzájemné propojení virtuálních strojů. Umožňuje spustit na řídícím serveru DHCP server, jehož parametry máme plně pod kontrolou a je možné je snadno měnit dle potřeby vývoje.

Druhý virtuální server sloužil nejprve k odladění možností PXE menu a kontrole, že PXE poskytuje veškerou požadovanou funkcionalitu. Následně zde probíhaly testy instalací jednotlivých systémů ze sítě a možností přizpůsobení jednotlivých instalačních metod. Tento stroj byl připojen pouze do vnitřní sítě a adresu s dalšími parametry získával z DHCP serveru na řídícím serveru.

Aby měl instalovaný stroj přístup do internetu, bylo nutné na řídícím serveru povolit IP Forwarding a zároveň pomocí `iptables` nastavit NAT

```
sysctl  -w net.ipv4.ip_forward=1
iptables -t nat -A POSTROUTING -o enp0s3 -j MASQUERADE
```

3.6.2 Testování v laboratořích KIV

Pro testování na infrastruktuře v laboratořích KIV byl poskytnut virtuální server na serverové infrastruktuře katedry, který byl připojen do stejné sítě jako instalované stanice. Server sloužil jako řídící server a byly na něj nainstalovány komponenty dle popisu v kapitolách 3.2, 3.3, 3.4 a 3.5.4. Testy probíhaly na strojích celkem ve třech laboratořích a to konkrétně UC-326, UC-332 a UC-336.

Během testu se objevili drobné odlišnosti, především s možností předávání řízení lokálnímu zavaděči. Dle příznaků se nejspíše jednalo o problematickou spolupráci PXE zavaděče a pevných disků připojených přes AHCI. Zde pomohla změna konfigurace, která už je promítnutá do definice PXE menu popsané v 3.2.4.1.

Podařilo se však v praxi potvrdit, že všechny modely použitých zařízení umožňují nastavit různé pořadí zavaděčů dle metody spouštění. Zatím co u ručního spuštění

uživatelem v laboratoři může rovnou naběhnout lokální zavaděč, při probuzení ze sítě dojde z zavedení ze sítě pomocí PXE. Toto výrazně zvyšuje uživatelský komfort, protože nejsou matení síťovým zavaděče v úvodu.

Během testů byl identifikován drobný problém s rozdělením laboratoří do dvou různých podsítí. U laboratoří, které byly ve stejné podsíti jako řídící server vše fungovalo dle předpokladů, ale v laboratoři UC-326 nebylo možné vzdáleně probouzet stanice, neboť Wake-on-Lan je omezen na L2 vrstvu. Jako řešení bylo do řídícího serveru přidané druhé síťové rozhraní přiřazené do odpovídající sítě a byl upraven pomocný skript, aby používal odpovídající síťové rozhraní pro odchozí magic pakety.

Cílem testů bylo ověřit funkčnost řešení v jednotlivých laboratořích, které se liší použitým hardwarem a/nebo síťovou konfigurací. Zároveň bylo otestováno zpracování chybových stavů, které mohou během instalace nastat. Jednotlivé testovací scénáře pro otestování implementované funkčnosti jsou popsány v Příloha F

4 Závěr

Cílem práce bylo vytvořit systém pro řízení automatické instalace pracovních stanic v prostředí KIV. Hlavním požadavkem na systém byla možnost vzdáleně provádět instalace dle zvolených profilů na jednotlivé stroje. Zároveň by měla být administrátorům dostupná aplikace s informacemi o stavu instalací.

V rámci práce byly prozkoumány možnosti automatické instalace GNU/Linuxu a MS Windows stejně jako hotová řešení pro správu a řízení instalací. Protože tato řešení nesplňují požadavky kladené zadavatelem, bylo přistoupeno k vývoji nového řešení. Vytvořené řešení kombinuje existující komponenty pro jednotlivé dílčí úlohy. K těmto komponentám byl vytvořen daemon pro řízení instalací a webové rozhraní, které umožňuje administrátorům spouštět instalace a sledovat jejich průběh.

Nový systém umožňuje administrátorům instalovat laboratoře dle různorodých požadavků na výuku a celý tento proces snadno monitorovat pomocí webového rozhraní. Zároveň je možné přidávat nebo upravovat existující profily instalace operačních systémů, které budou reflektovat změny v požadovaném softwarovém vybavení.

Systém byl vytvořen jako modulární, což v budoucnu umožní nahradit jednotlivé komponenty v případě změny požadavků. Díky této modularitě může být systém dále rozvíjen a může být přidávána další funkcionalita, která umožní zefektivnit práci administrátorů.

Vyvinuté řešení bylo testováno v laboratořích KIV a došlo s jeho pomocí k přeinstalování počítačů ve třech laboratořích napříč KIV. Vytvořené řešení je připraveno na zavedení do ostrého provozu v rámci správy laboratoří KIV a může tak dojít k nahrazení manuálních kroků během instalace laboratoří, což bylo hlavním cílem této práce.

Příloha A Seznam zdrojů

[1] Join a Computer to a Domain [online]. Microsoft, 2017 [cit. 2019- 07-14]. Dostupné z: https://docs.microsoft.com/cs-cz/windows-server/identity/ad-fs/deployment/join-a-computer-to-a-domain

[2] Clonezilla: The Free and Open Source Software for Disk Imaging and Cloning [online], 2019 [cit. 2019-06-25]. Dostupné z: https://clonezilla.org/

[3] Acronis True Image 2019 [online]. [cit. 2019-06-25]. Dostupné z: https://www.acronis.cz/produkt/true-image-2019/

[4] Magic Packet Technology: White Paper [online]. 1998 [cit. 2019-06-25]. Dostupné z: https://www.amd.com/system/files/TechDocs/20213.pdf

[5] WOL Agent [online]. 2018 [cit. 2019-06-25]. Dostupné z: https://wol.aquilatech.com/WOLAgent/

[6] Etherwake: Details of package [online]. 2013 [cit. 2019-06-25]. Dostupné z: https://packages.debian.org/stretch/etherwake

[7] Wakeonlan: Perl script for waking up computers via Wake-On-LAN magic packets [online]. 2013 [cit. 2019-06-25]. Dostupné z: https://github.com/jpoliv/wakeonlan

[8] Nirsoft - WakeMeOnLan [online]. 2019 [cit. 2019-06-25]. Dostupné z: https://www.nirsoft.net/utils/wake_on_lan.html

[9] WakeOnLANx [online]. 2019 [cit. 2019-06-25]. Dostupné z: https://wakeonlanx.com/wakeonlanx/

[10] IPMI, V1.0: Overview, Progress, and Implementation [online]. intel, 1998 [cit. 2019-06-25]. Dostupné z: https://www.intel.co.uk/content/www/uk/en/servers/ipmi/ipmi-overview.html

[11] Intelligent Platform Management Interface Specification v2.0: rev. 1.1 E7 Markup [online]. intel, 2015 [cit. 2019-06-25]. Dostupné z: https://www.intel.co.uk/content/www/uk/en/servers/ipmi/ipmi-intelligent-platform-mgt-interface-spec-2nd-gen-v2-0-spec-update.html

[12] iDRAC 9 [online]. Dell, 2019 [cit. 2019-06-25]. Dostupné z: https://www.dell.com/support/article/us/en/04/sln311300/idrac9-home

[13] Linux Journal. Robert F. Young, 2001, (81-86). ISSN 1075-3583.

[14] STANEK, William. Windows 10: Essentials for Administration. Stanek & Associates, 2016. ISBN 9781575455266.

[15] ISOLINUX - Syslinux Wiki [online]. 2019 [cit. 2019-07-14]. Dostupné z: https://wiki.syslinux.org/wiki/index.php?title=ISOLINUX

[16] The Syslinux Project [online]. 2016 [cit. 2019-07-14]. Dostupné z: https://wiki.syslinux.org/

[17] RUSSELL, Jesse a Ronald COHN. Preboot Execution Environment. Book on Demand, 2012. ISBN 9785512123882.

[18] PXELINUX: Sysliunx Wiki [online]. 2016 [cit. 2019-07-14]. Dostupné z: https://wiki.syslinux.org/wiki/index.php?title=PXELINUX

[19] CROFT, Bill a John GILMORE. BOOTSTRAP PROTOCOL (BOOTP). RFC 951. Internet Requests for Comment. RFC Editor, 1985. ISSN 2070-1721. Dostupné také z: http://www.rfc-editor.org/rfc/rfc951.txt

[20] DROMS, R. Dynamic Host Configuration Protocol. RFC 2131. Internet Requests for Comment. RFC Editor, 1997. ISSN 2070-1721. Dostupné také z: http://www.rfc-editor.org/rfc/rfc2131.txt

[21] MILLS, D., U. DELAWARE, J. MARTIN, J. BURBANK a W. KASCH. Network Time Protocol Version 4: Protocol and Algorithms Specification. RFC 5905. Internet Requests for Comment. RFC Editor, 2010. ISSN 2070-1721. Dostupné také z: http://www.rfc-editor.org/rfc/rfc5905.txt

[22] ALEXANDER, S. a R. DROMS. DHCP Options and BOOTP Vendor Extensions. RFC 2132. Internet Requests for Comment. RFC Editor, 1997. ISSN 2070-1721. Dostupné také z: http://www.rfc-editor.org/rfc/rfc2132.txt

[23] POSTEL, J. a J. REYNOLDS. FILE TRANSFER PROTOCOL (FTP). RFC 959. Internet Requests for Comment. RFC Editor, 1985. ISSN 2070-1721. Dostupné také z: http://www.rfc-editor.org/rfc/rfc959.txt

[24] TRANSMISSION CONTROL PROTOCOL. RFC 793. Internet Requests for Comment. RFC Editor, 1981. ISSN 2070-1721. Dostupné také z: http://www.rfc-editor.org/rfc/rfc793.txt

[25] SOLLINS, K. THE TFTP PROTOCOL (REVISION 2). RFC 1350. Internet Requests for Comment. RFC Editor, 1992. ISSN 2070-1721. Dostupné také z: http://www.rfc-editor.org/rfc/rfc1350.txt

[26] MALKIN, G. a A. HARKIN. TFTP Blocksize Option. RFC 2348. Internet Requests for Comment. RFC Editor, 1998. ISSN 2070-1721. Dostupné také z: http://www.rfc-editor.org/rfc/rfc2348.txt

[27] POSTEL, J. User Datagram Protocol. RFC 768. Internet Requests for Comment. RFC Editor, 1980. ISSN 2070-1721. Dostupné také z: http://www.rfc-editor.org/rfc/rfc768.txt

[28] GOURLEY, David a Brian TOTTY. HTTP: the definitive guide. Sebastopol, CA: O'Reilly, 2002. ISBN 9781565925090.

[29] HTTP: A Protocol for networked information [online]. W3C I, 1992 [cit. 2019-06- 25]. Dostupné z: https://www.w3.org/Protocols/HTTP/HTTP2.html

[30] The Internet Engineering Task Force [online]. 2019 [cit. 2019-07-14]. Dostupné z: https://www.ietf.org

[31] BERNERS-LEE, T., R. FIELDING a H. FRYSTYK. Hypertext Transfer Protocol -- HTTP/1.0. RFC 1945. Internet Requests for Comment. RFC Editor, 1996. ISSN 2070-1721. Dostupné také z: http://www.rfc-editor.org/rfc/rfc1945.txt

[32] BELSHE, M., R. PEON a M. THOMSON. Hypertext Transfer Protocol Version 2 (HTTP/2). RFC 7540. Internet Requests for Comment. RFC Editor, 2015. ISSN 2070-1721. Dostupné také z: http://www.rfc-editor.org/rfc/rfc7540.txt

[33] Debian: Univerzální operační systém [online]. 2019 [cit. 2019-07-14]. Dostupné z: https://www.debian.org/

[34] Contents of the preconfiguration file (for stretch) [online]. Debian, 2019 [cit. 2019- 06-25]. Dostupné z: https://www.debian.org/releases/stretch/example-preseed.txt

[35] Using preseeding [online]. [cit. 2019-06-25]. Dostupné z: https://www.debian.org/releases/stable/i386/apbs02.html.en
[36] DebianInstaller [online]. 2019 [cit. 2019-07-14]. Dostupné z: https://wiki.debian.org/DebianInstaller
[37] FAI Guide (Fully Automatic Installation) [online]. [cit. 2019-06-25]. Dostupné z: https://fai-project.org/fai-guide/#_abstract
[38] STERN, Hal, Mike EISLER a Ricardo LABIAGA. Managing NFS and NIS. 2nd ed. Sebastopol, CA, c2001. ISBN 978-1565925106.
[39] HAYNES, T. a D. NOVECK. Network File System (NFS) Version 4 Protocol. RFC 7530. Internet Requests for Comment. RFC Editor, 2015. ISSN 2070-1721. Dostupné také z: http://www.rfc-editor.org/rfc/rfc7530.txt
[40] SHEPLER, S., M. EISLER a D. NOVECK. Network File System (NFS) Version 4 Minor Version 1 Protocol. RFC 5661. Internet Requests for Comment. RFC Editor, 2010. ISSN 2070-1721. Dostupné také z: http://www.rfc-editor.org/rfc/rfc5661.txt
[41] Network File System overview: Microsof Docs [online]. 2018 [cit. 2019-07-14]. Dostupné z: https://docs.microsoft.com/en-us/windows-server/storage/nfs/nfs-overview
[42] Samba [online]. [cit. 2019-06-25]. Dostupné z: https://www.samba.org/
[43] RPM Package Manager [online]. [cit. 2019-06-25]. Dostupné z: https://rpm.org/
[44] SMYTH, Neil. Red Hat Enterprise Linux 8 Essentials: Learn to Install, Administer and Deploy RHEL 8 Systems. Payload Media, Incorporated, 2019. ISBN 9780986027390.
[45] Fedora [online]. Red Hat, 2019 [cit. 2019-06-25]. Dostupné z: https://getfedora.org/cs/
[46] HOW DO YOU PERFORM A KICKSTART INSTALLATION? [online]. [cit. 2019- 06-25]. Dostupné z: https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/ installation_guide/sect-kickstart-howto
[47] Kickstart generator [online]. [cit. 2019-06-25]. Dostupné z: https://access.redhat.com/labs/kickstartconfig/
[48] Kerberos: The Network Authentication Protocol [online]. 2019 [cit. 2019-07-14]. Dostupné z: https://web.mit.edu/kerberos/
[49] Microsoft Deployment Toolkit documentation [online]. 2019 [cit. 2019-07-14]. Dostupné z: https://docs.microsoft.com/cs-cz/sccm/mdt/
[50] System Center Configuration Manager [online]. Microsoft, 2019 [cit. 2019-06-25]. Dostupné z: https://www.microsoft.com/cs-cz/cloud-platform/system-center- configuration-manager
[51] Windows PE (WinPE) [online]. Microsoft, 2019 [cit. 2019-06-25]. Dostupné z: https://docs.microsoft.com/en-us/windows-hardware/manufacture/desktop/winpe- intro
[52] Windows 10 deployment scenarios and tools [online]. Microsoft, 2019 [cit. 2019-06- 25]. Dostupné z: https://docs.microsoft.com/en-us/windows/deployment/windows-deployment-scenarios-and-tools

[53] Wimlib: the open source Windows Imaging (WIM) library [online]. 2019 [cit. 2019- 06-25]. Dostupné z: https://wimlib.net/index.html
[54] DISM: Deployment Image Servicing and Management [online]. Microsoft, 2019 [cit. 2019-06-25]. Dostupné z: https://docs.microsoft.com/en-us/windows-hardware/manufacture/desktop/dism---deployment-image-servicing-and-management-technical-reference-for-windows
[55] Myths About Samba [online]. [cit. 2019-06-25]. Dostupné z: https://www.samba.org/ samba/docs/myths_about_samba.html
[56] [MS-SMB]: Server Message Block (SMB) Protocol [online]. Microsoft, 2019 [cit. 2019-06-25]. Dostupné z: https://docs.microsoft.com/en-us/openspecs/windows_protocols/ms-smb/f210069c- 7086-4dc2-885e-861d837df688
[57] AIO Boot – All-in-One bootable software | Multiboot USB Creator [online]. [cit. 2019-06-25]. Dostupné z: https://www.aioboot.com/en/
[58] AIO Boot – Change history [online]. [cit. 2019-06-25]. Dostupné z: https://www.aioboot.com/en/history-changes/
[59] FOGUserGuide [online]. [cit. 2019-06-25]. Dostupné z: https://wiki.fogproject.org/wiki/index.php?title=FOGUserGuide#Background_on_FOG
[60] FOG Project: repository [online]. [cit. 2019-06-25]. Dostupné z: https://github.com/FOGProject/
[61] FOG Project [online]. [cit. 2019-06-25]. Dostupné z: https://fogproject.org/
[62] Managing FOG [online]. 2019 [cit. 2019-06-25]. Dostupné z: https://wiki.fogproject.org/wiki/index.php?title=Managing_FOG
[63] Debian GNU/Linux Installation Guide [online]. Debian Installer team [cit. 2019-08-09]. Dostupné z: https://www.debian.org/releases/stretch/amd64/index.html.en
[64] TFTPD: packages details [online]. 2019 [cit. 2019-07-14]. Dostupné z: https://packages.debian.org/stretch/tftpd
[65] Comboot/menu.c32: Syslinux wiki [online]. 2019 [cit. 2019-06-25]. Dostupné z: https://wiki.syslinux.org/wiki/index.php?title=Comboot/menu.c32
[66] Download and install the Windows ADK [online]. 2019 [cit. 2019-06-25]. Dostupné z: https://docs.microsoft.com/en-us/windows-hardware/get-started/adk-install
[67] Stáhnout image disku s Windows 10 (soubor ISO) [online]. Microsoft Corporation [cit. 2019-08-09]. Dostupné z: https://www.microsoft.com/cs-cz/software-download/windows10ISO
[68] Components [online]. 2019 [cit. 2019-06-25]. Dostupné z: https://docs.microsoft.com/en-us/windows-hardware/customize/desktop/unattend/ components-b-unattend
[69] KMS client setup keys [online]. [cit. 2019-06-25]. Dostupné z: https://docs.microsoft.com/en-us/windows-server/get-started/kmsclientkeys

[70] ST. LAURENT, Simon a Michael JAMES FITZGERALD. XML Pocket Reference: Extensible Markup Language. 3 edition. O'Reilly Media, 2005. ISBN 978-0596100506.
[71] PHP Hypertext Preprocessor [online]. [cit. 2019-08-10]. Dostupné z: https://php.net/
[72] Nette Framework [online]. [cit. 2019-06-25]. Dostupné z: https://nette.org/
[73] MEYER, Eric A. a Estelle WEYEL. CSS: the definitive guide : visual presentation for the web. Fourth edition. Sebastopol, CA: O´Reilly, 2018. ISBN 978-1449393199.
[74] Bootstrap [online]. [cit. 2019-06-25]. Dostupné z: https://getbootstrap.com/
[75] Feather -Simply beautiful open source icons [online]. [cit. 2019-06-25]. Dostupné z: https://feathericons.com/
[76] Prezentační vzory z rodiny MVC [online]. Borek Bernard, 2009 [cit. 2019-08-10]. Dostupné z: https://www.zdrojak.cz/clanky/prezentacni-vzory-zrodiny-mvc/
[77] DEB.SURY.ORG [online]. [cit. 2019-06-25]. Dostupné z: https://deb.sury.org/
[78] Let's Encrypt - Free SSL/TLS Certificates [online]. 2019 [cit. 2019-08-10]. Dostupné z: https://letsencrypt.org/
[79] Certbot [online]. 2019 [cit. 2019-08-10]. Dostupné z: https://certbot.eff.org/
[80] GHEORGHE, Lucian. Designing and Implementing Linux Firewalls with QoS using netfilter, iproute2, NAT and L7-filter. Packt Publishing, 2006. ISBN 978-1904811657.
[81] Oracle VM VirtualBox [online]. [cit. 2019-06-25]. Dostupné z: https://www.virtualbox.org/

Příloha B Seznam zkratek

ACL	access control list
AIO	all-in-one
BCD	Boot Configuration Data
BOOTP	bootstrap protocol
DHCP	Dynamic Host Configuration Protocol
DISM	Deployment Image Servicing and Management
FAI	Fully Automatic Installation
FTP	File Transfer Protocol
HTTP	HyperText Transfer Protocol
IETF	Internet Engineering Task Force
IPMI	Intelligent Platform Management Interface
MDT	Microsoft Deployment Toolkit
MS	Microsoft
NFS	Network File System
NTP	Network Time Protocol
PXE	Preboot eXecution Environment
SMB	Server Message Block
TCP	Transmission Control Protocol
TFTP	Trivial File Transfer Protocol
UDP	User Datagram Protocol
W3C	World Wide Web Consortium
WDS	Windows Deployment Services
Windows ADK	Windows Assessment and Development Kit
Windows PE	Windows Preinstallation Environment
Windows SIM	Windows System Image Manager
WoL	Wake-on-Lan

Příloha C Seznam obrázků

Příloha D Tvorba konfigurace pro instalaci MS Windows

Pro tvorbu konfiguračních souborů pro automatickou instalaci a prostředí Windows PE potřebujeme sadu nástrojů Windows ADK. Veškeré kroky provádíme na počítači s MS Windows. Výsledné soubory následně přeneseme na řídící počítač.

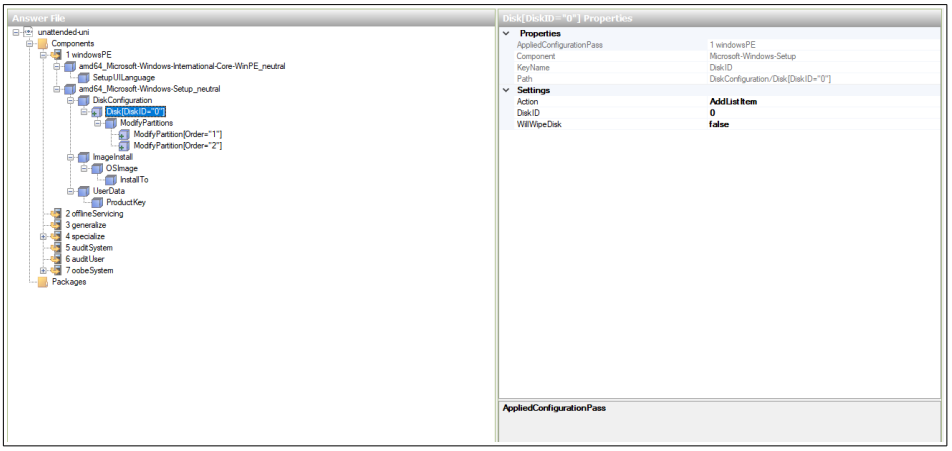
D.1.1 Instalace

Ze stánky https://docs.microsoft.com/cs-cz/windows-hardware/get-started/adk-install#winADK stáhneme Windows ADK verze 1903 a nainstalujeme. Při instalaci vybereme volby dle Obr 3.3

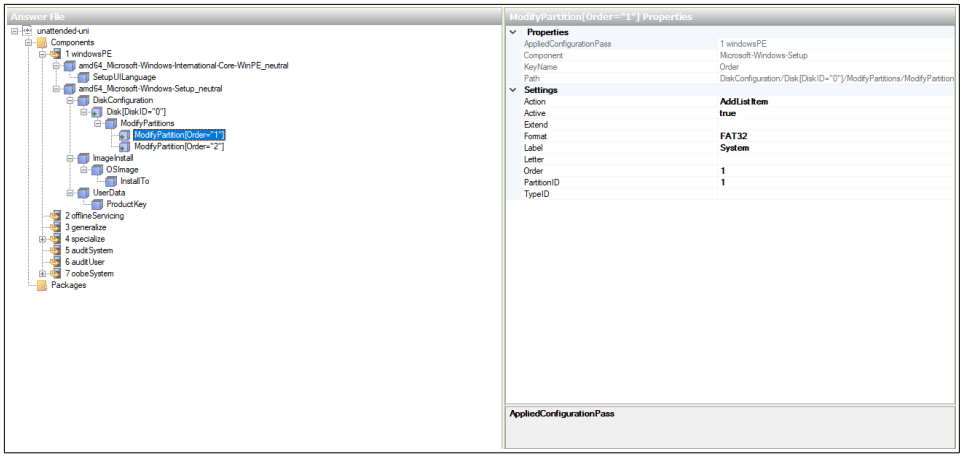
D.1.2 Konfigurace

Pro vytoření konfiuračního souboru pro automatickou instalaci spustíme z nabídky start nástroj Windows System Image Manager a provedeme prvotní nastvení dle popisu v kapitole 3.4.2. jednotlivé volby a možnosti nastavení shrnují následující obrázky.

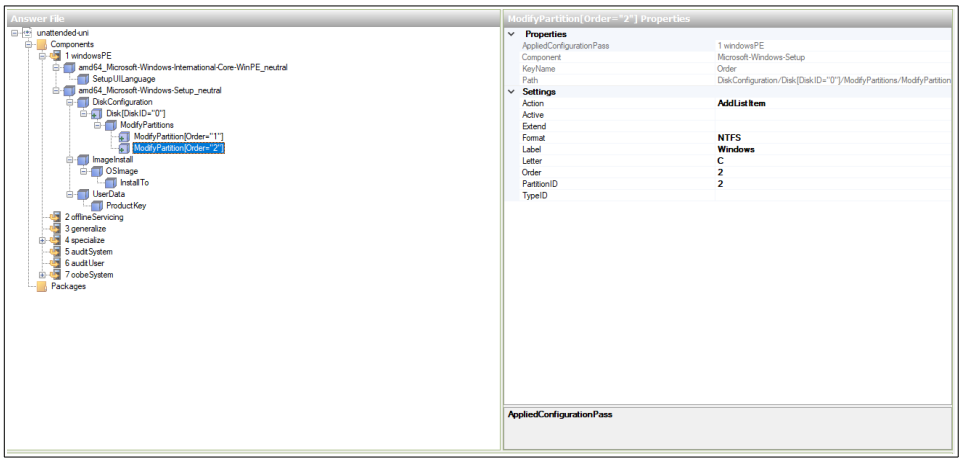
Kroky na Obr D.1 Obr D.2 Obr D.3 Obr D.4 a Obr D.5 se týkají fáze Windows PE kdy se procádí základní nastavení pro instalace. Důležité je v kroku na Obr D.1 nastavit parametr WillWipeDisk na false, aby nedošlo ke smazání již provedené instalace GNU/ Linuxu !!!



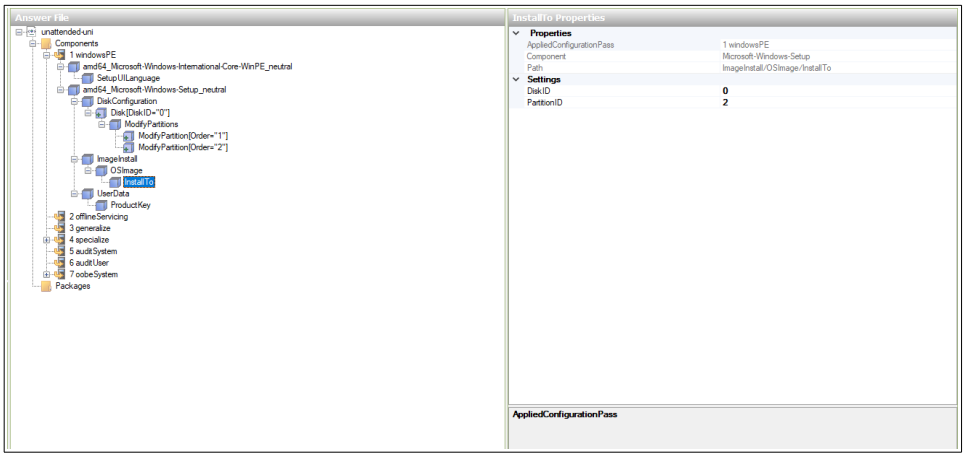
Obr D.1: Výběr pevného disku



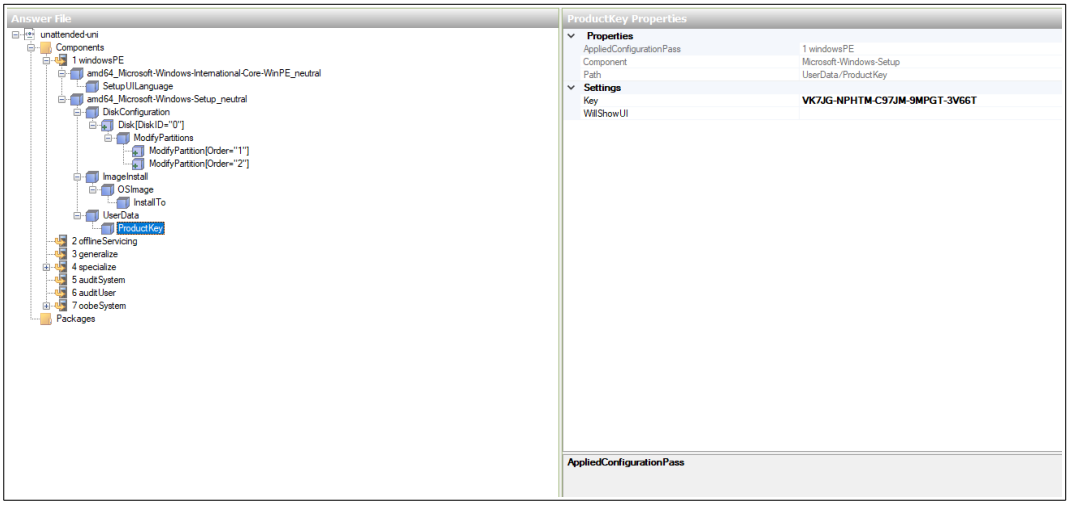
Obr D.2: Nastavení servisního oddílu



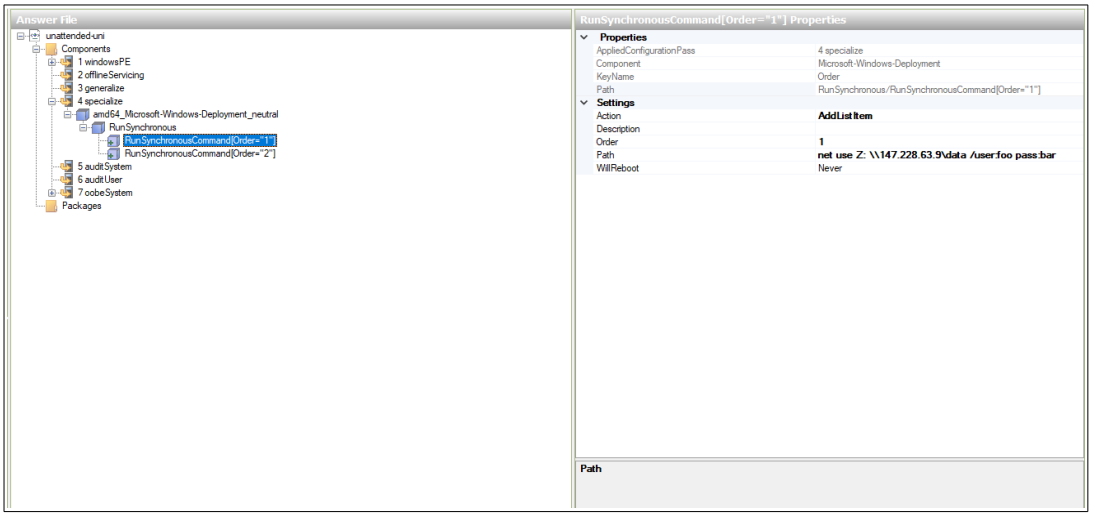
Obr D.3: Nastavení systémového oddílu



Obr D.4: Výběr cíle instalace

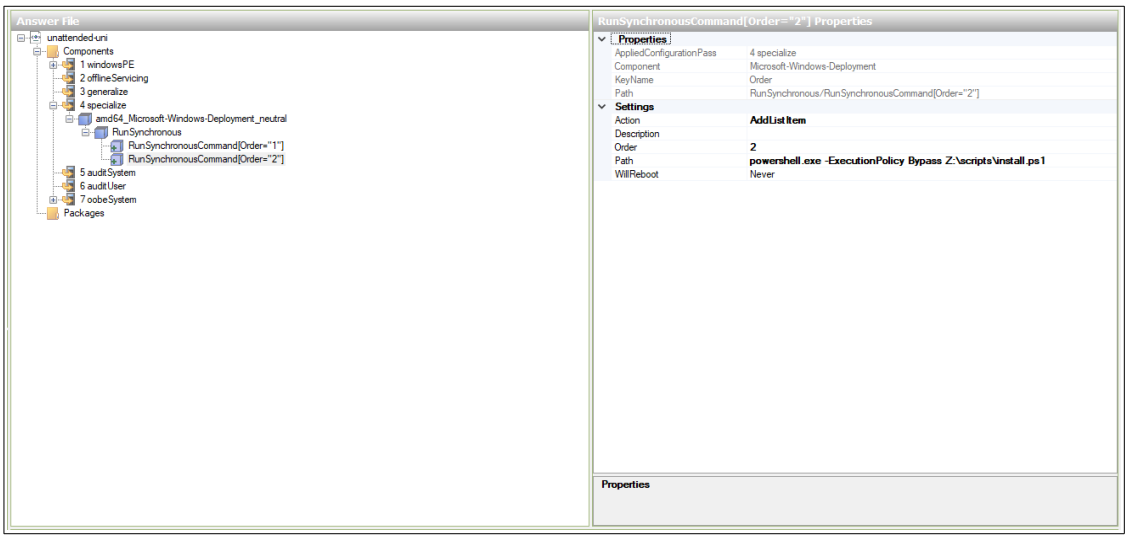


Obr D.5: Vložení klíče produktu



Obr D.6: Připojení síťového disku

V následující fázi provedeme spuštění skriptu pro instalaci potřebných balíčků. Skript install.ps1 prohledává zadaný adresář a provádí instalaci všech msi balíčků, které v tomto adresáři najde. Před tím je však nutné síťovou složku připojit jako disk, tento úkon je na Obr D.6 a vlastní spuštění skriptu na Obr D.7

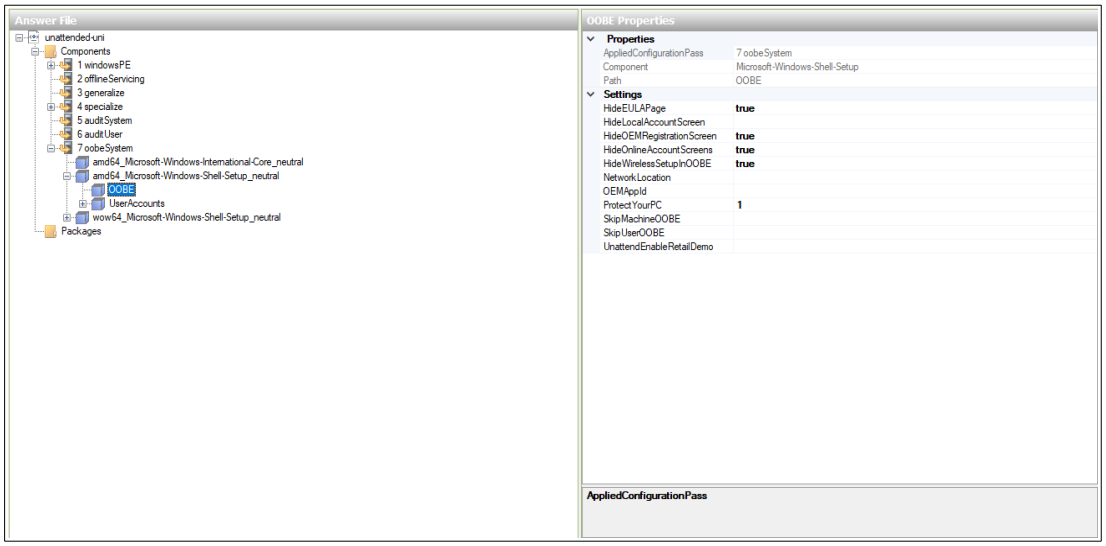


Obr D.7: Spuštění instalačního skriptu

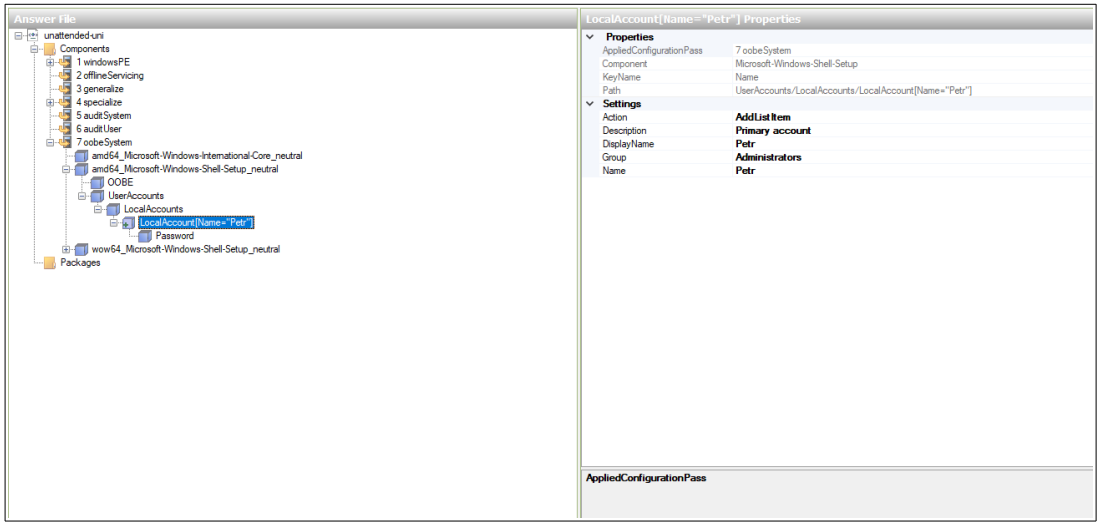
Vlastní obsah skriptu `install.ps1` je následující

```
$FILES = Get-ChildItem Z:\packages -File
$FILES = $FILES.name | Select-String -Pattern msi
foreach ($FILE in $FILES)
{
  Start-Process msiexec.exe -Wait -ArgumentList "/i Z:\packages\
$FILE /quiet"
}
```

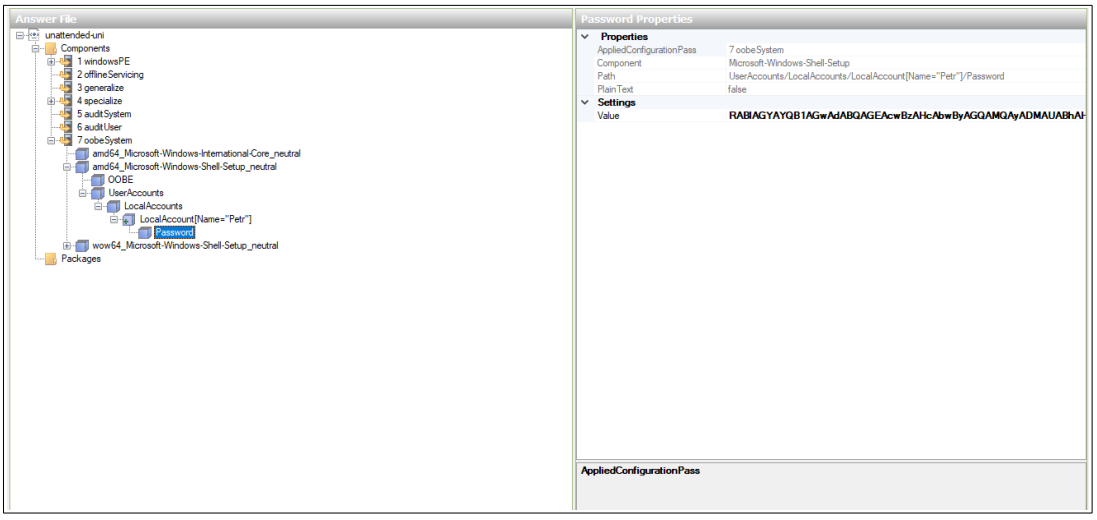
V poslední části se provádí nastavení pro zabránění zobrazení uvítacách obrazovek při prvním spuštění Obr D.8, a vytvořené servisního uživatele Obr D.9 s nastavením jeho hesla Obr D.10



Obr D.8: Skrytí uvítacích a licenčních obrazovek



Obr D.9: Vytvoření servisního uživatele



Obr D.10: Tvorba hesla pro servisního uživatele

D.1.3 Zabezpečení

Pozor ačkoliv heslo vypadá ve výsledném souboru jako šifrované, jedná se pouze o převod do formátu Base64. Tento formát jde velmi snadno převést na čitelný text !!!

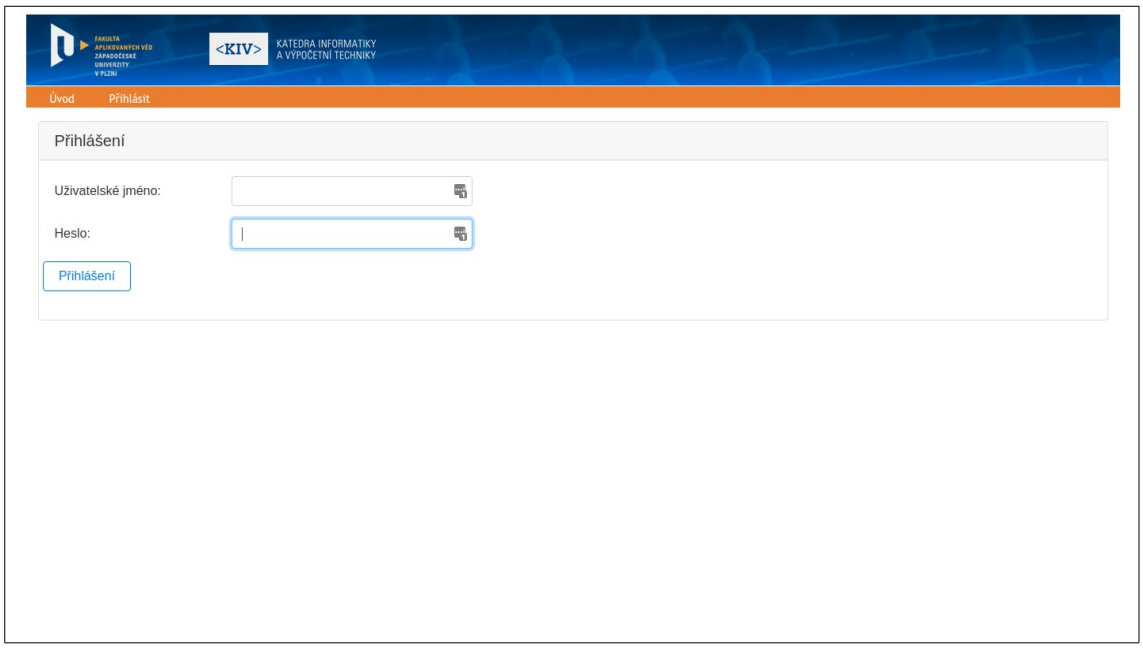
Příloha E Uživatelská příručka

Aplikace se ovládá přes webové rozhraní, které je k dispozici na adrese https://labs.kiv.zcu.cz a je dostupné pouze z adresního rozsahu univerzity. Po otevření stránky v prohlížeči se nám zobrazí úvodní stránka zobrazena na Obr E.1.

Obr E.1: Úvodní stránka webového rozhraní

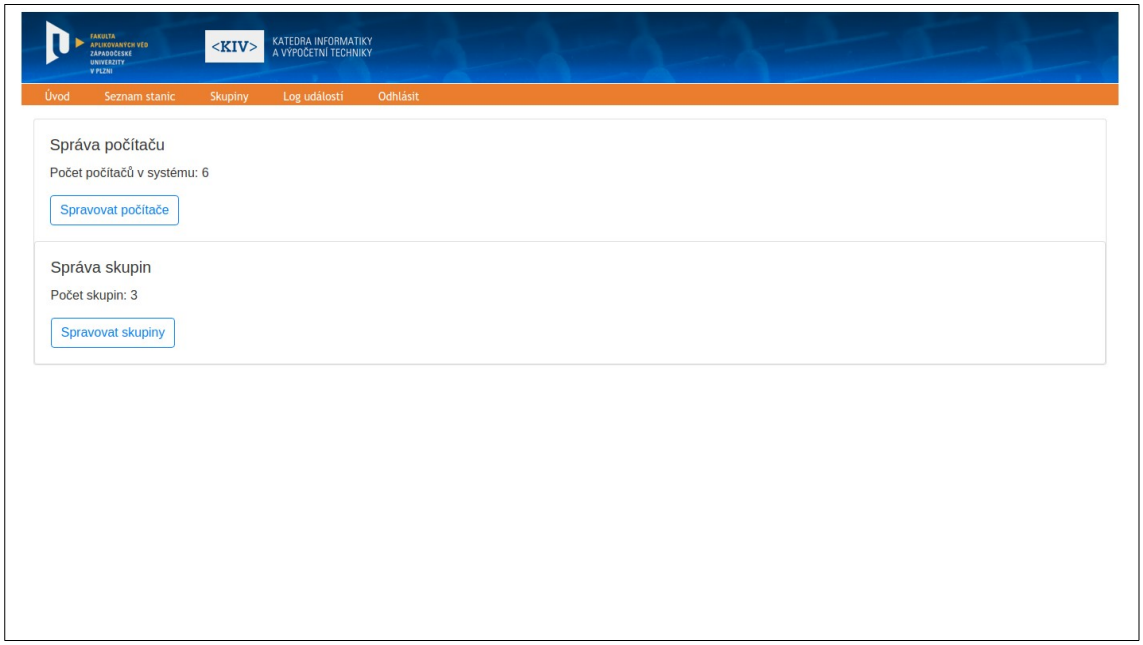
Pro práci s aplikací se musíme nejprve přihlásit na přihlašovací obrazovce (Obr E.2), ke které se dostaneme pomocí tlačítka přihlásit, nebo z položky menu v hlavičce webu. Výchozí přihlašovací údaje jsou:

```
Přihlašovací jméno: diplomka
Heslo: diplomka
```



Obr E.2: Přihlašovací obrazovka

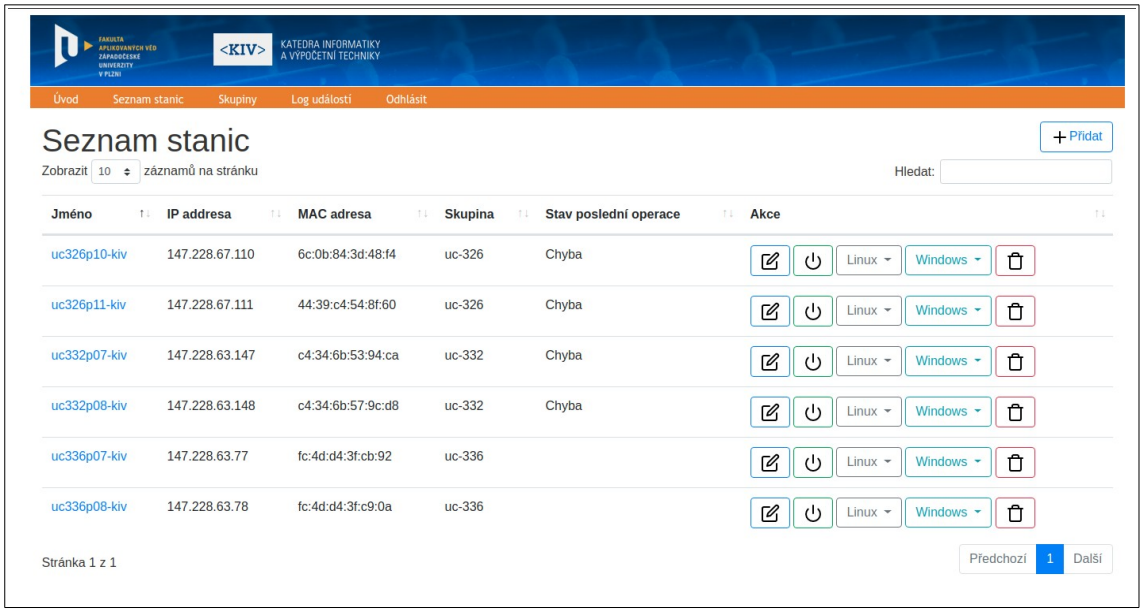
Po přihlášení se dostaneme na domovskou stránku (Obr E.3), kde máme k dipozici základní informace o systému a rozcestník. Odkazy na všechny součásti webového rozhraní jsou po celou dobu dostupná v hlavičce webu.



Obr E.3: Domovská stránka po přihlášení

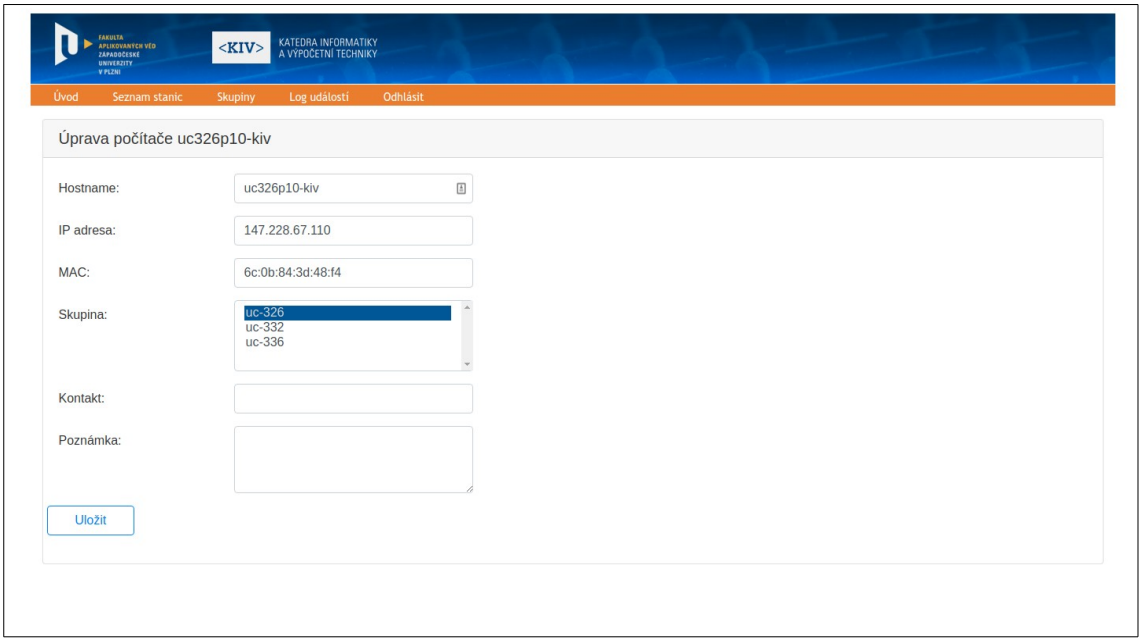
Na stránce se správou počítačů vyobrazené na Obr E.4 máme k dispozici výpis všech počítačů v systému. Pomocí záhlaví tabulky můžeme položky řadit dle

jednotlivých parametrů, případně vyhledávat dle jména přes vyhledávací pole v pravém horním rohu tabulky. U každého počítače máme k dispozici možnost editace, dále jednotlivé akce dostupné v systému a poslední položkou je smazání počítače. Pomocí tlačítka **Přidat** můžeme přidat další počítač do systému.



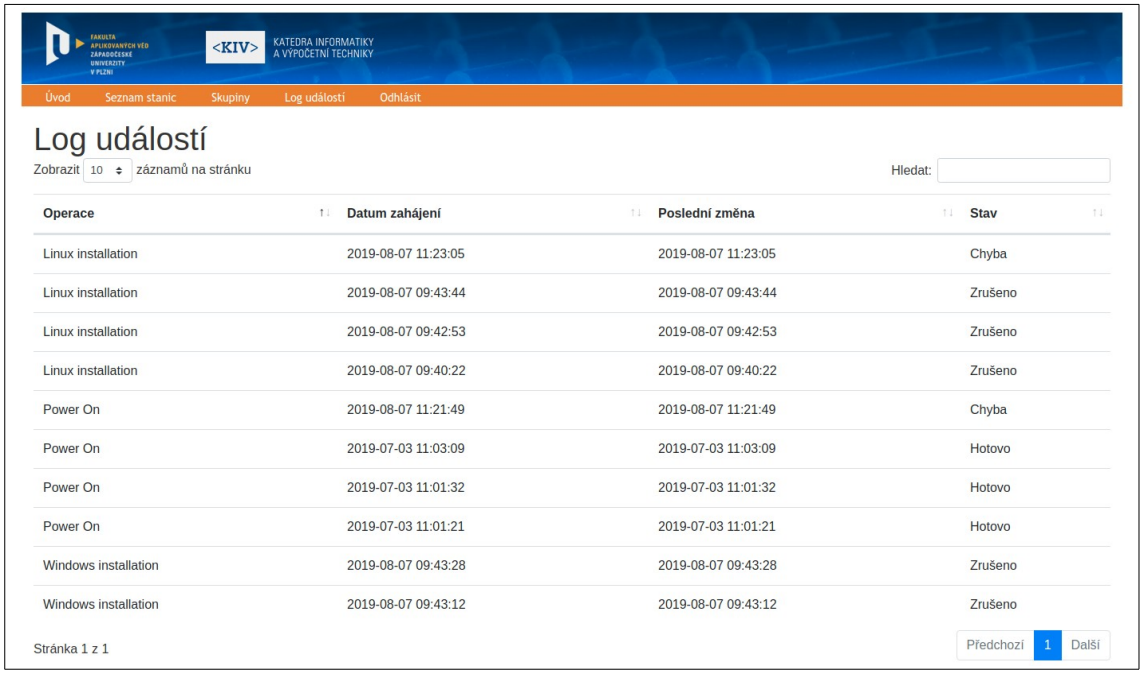
Obr E.4: Seznam počítačů v systému

Pro přidání i editaci slouží stejný formulář, jen pro editaci je předvyplněn současnými hodnotami jednotlivých polí. Verzi pro editaci zobrazuje Obr E.5



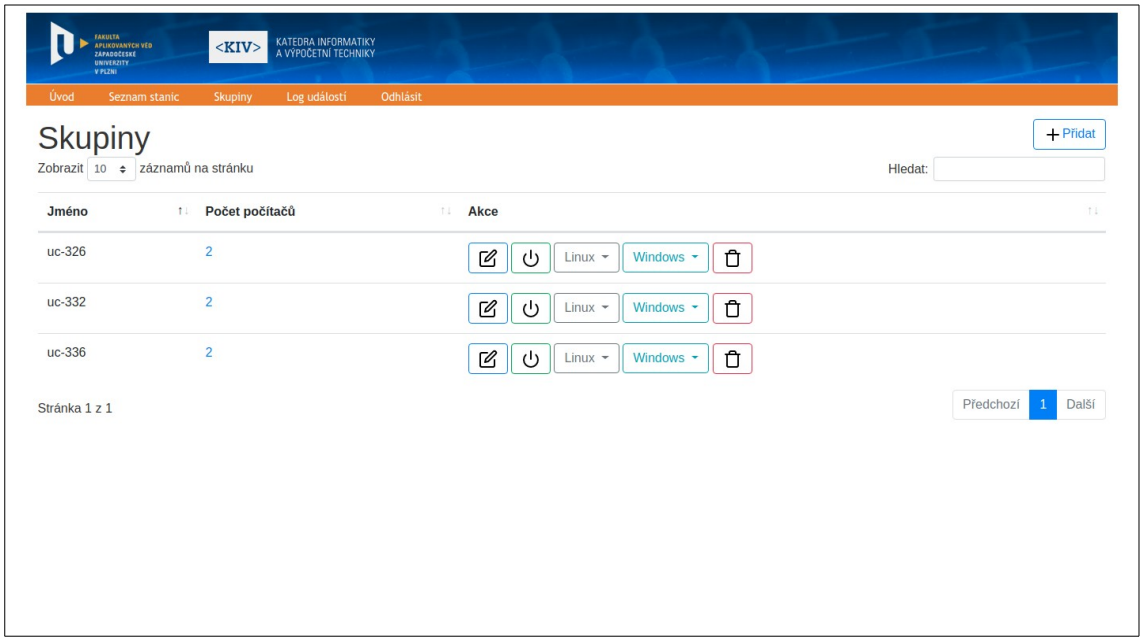
Obr E.5: Formulář pro editaci/přidání počítače

Jméno počítače slouží jako proklik na záznam událostí vztažených k danému počítači. Ukázka je na Obr E.6



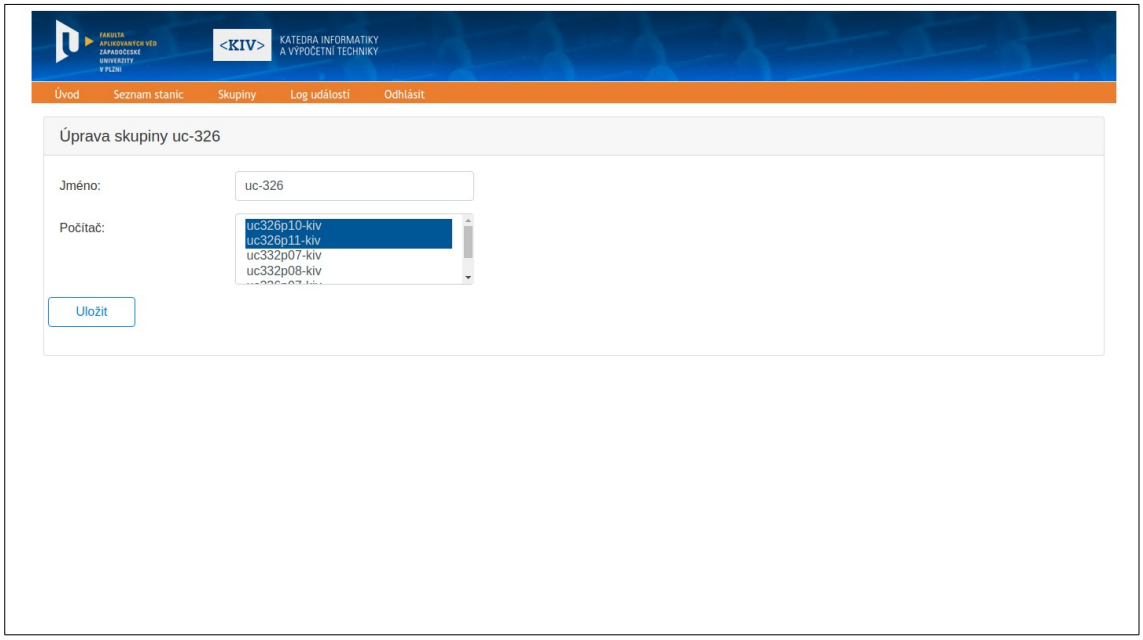
Obr E.6: Log událostí vztažený k danému počítači

Další součástí webového rozhraní je správa skupin počítačů. Skupiny slouží k seskupování počítačů se stejnými vlastnostmi a ke spouštění akcí nad celou skupinou počítačů. Seznam počítačů je vidět na Obr E.7



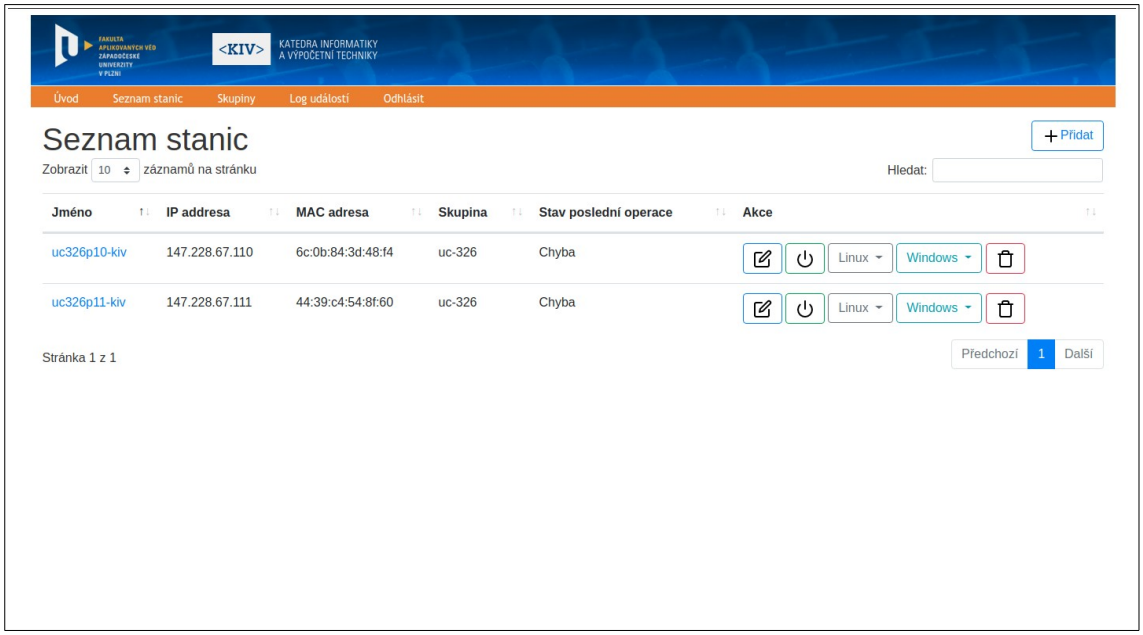
Obr E.7: Skupiny počítačů

Zde jsou opět sdílené formuláře pro editaci a přidání nov skupiny. Přidání je možné přes tlačítko **Přidat** v pravém horním rohu. Formulář je zobrazen na Obr E.8



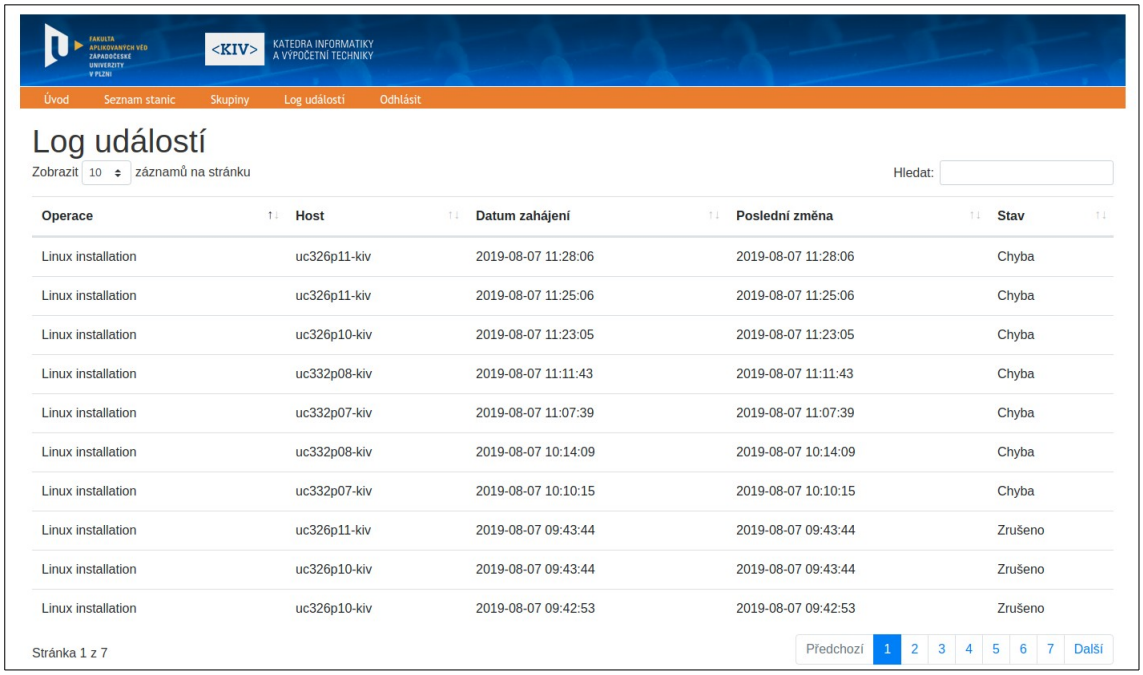
Obr E.8: Formulář pro přídání/editaci skupiny počítačů

Přes počet počítačů se dá prokliknout na seznam počítačů zařazených do dané skupiny. Zobrazení je obdobné jako seznam všech počítačů, jen zobrazuje pouze počítače ze skupiny jak je vidět na Obr E.9



Obr E.9: Seznam počítačů patřících do jedné skupiny

Poslední obrazovkou je log všech událostí provedených v systému, včetně času zadání a času poslední změny stavu u dané operace. Tabulku je možné řadit pomocí šipek v záhlaví tabulky nebo filtrovat pomocí vyhledávacího pole v pravém horním rohu. Obrazovka je zobrazena na Obr E.10



Obr E.10: Log všech událostí v systému

Příloha F Testovací scénáře

Pro testování implementované funkcionality v kapitole 3 byly připraveny následující testovací scénáře včetně testu ošetření chybových stavů

F.1 Zapnutí počítače

Test zapnutí počítače z webového rozhraní.

Krok	Operace	Očekávaný výsledek
1	Přihlášení do webové aplikace	Zobrazení úvodní obrazovky
2	Přepnutí na stránku se seznamem počítačů	Zobrazení stránky se seznamem počítačů
3	Vyhledání testovacího počítače	Je zobrazen pouze testovaný počítač
4	Kontrola stavu počítače	Na počítači neprobíhá žádná akce
5	Stisknutí tlačítka pro zapnutí počítače	Stav se změní na „Naplánováno“
6	Vyčkat na dokončení operace	Stav se změní na „Dokončeno“, Počítač se začal spouštět

F.2 Zapnutí počítače s chybou

Test schopnosti systému zpracovat chybu při zapnutí počítače z webového rozhraní.

Krok	Operace	Očekávaný výsledek
1	Odpojení testovacího počítače	Testovací počítač není připojen do sítě
2	Přihlášení do webové aplikace	Zobrazení úvodní obrazovky
3	Přepnutí na stránku se seznamem počítačů	Zobrazení stránky se seznamem počítačů
4	Vyhledání testovacího počítače	Je zobrazen pouze testovaný počítač
5	Kontrola stavu počítače	Na počítači neprobíhá žádná akce
6	Stisknutí tlačítka pro zapnutí počítače	Stav se změní na „Naplánováno“
7	Vyčkat na dokončení operace	Stav se změní na „Chyba“, Počítač je stále vypnutí

F.3 Instalace GNU/Linuxu – výchozí profil

Test automatické instalace GNU/Linuxu z výchozího profilu – standardní průběh

Krok	Operace	Očekávaný výsledek
1	Přihlášení do webové aplikace	Zobrazení úvodní obrazovky
2	Přepnutí na stránku se seznamem počítačů	Zobrazení stránky se seznamem počítačů
3	Vyhledání testovacího počítače	Je zobrazen pouze testovaný počítač
4	Kontrola stavu počítače	Na počítači neprobíhá žádná akce
5	Stisknutí tlačítka pro Instalace GNU/Linuxu s profilem „Výchozí“	Stav se změní na „Naplánováno“
6	Vyčkat na start instalace	Stav se změní na „Probíhá“, Počítač se spustí a rozeběhne se automatická instalace
7	Vyčkat na dokončení operace	Stav se změní na „Dokončeno“, Počítač je vypnut
8	Manuální spuštění počítače	Počítač se začne spouštět s výchozím zavaděčem
9	Výběr nainstalovaného OS	Začne se načítat nainstalovaný OS
10	Vyčkat na naběhnutí systému	Zobrazení přihlašovací obrazovky
11	Přihlášení do sytému	Jsme přihlášeni v systému s výchozí konfigurací

F.4 Instalace GNU/Linuxu – výchozí profil s chybou

Test automatické instalace GNU/Linuxu z výchozího profilu – chybový průběh

Krok	Operace	Očekávaný výsledek
1	Přihlášení do webové aplikace	Zobrazení úvodní obrazovky
2	Přepnutí na stránku se seznamem počítačů	Zobrazení stránky se seznamem počítačů
3	Vyhledání testovacího počítače	Je zobrazen pouze testovaný počítač
4	Kontrola stavu počítače	Na počítači neprobíhá žádná akce
5	Stisknutí tlačítka pro Instalace GNU/Linuxu s profilem „Výchozí“	Stav se změní na „Naplánováno“

6	Vyčkat na start instalace	Stav se změní na „Probíhá“, Počítač se spustí a rozeběhne se automatická instalace
7	Odpojení testovacího počítače	Instalace se přeruší
8	Vyčkat na detekci chyby	Stav se změní na „Chyba“

F.5 Instalace GNU/Linuxu – alternativní profil

Test automatické instalace GNU/Linuxu z alternativního profilu – standardní průběh

Krok	Operace	Očekávaný výsledek
1	Přihlášení do webové aplikace	Zobrazení úvodní obrazovky
2	Přepnutí na stránku se seznamem počítačů	Zobrazení stránky se seznamem počítačů
3	Vyhledání testovacího počítače	Je zobrazen pouze testovaný počítač
4	Kontrola stavu počítače	Na počítači neprobíhá žádná akce
5	Stisknutí tlačítka pro Instalace GNU/Linuxu s jedním z alternativních profilů	Stav se změní na „Naplánováno“
6	Vyčkat na start instalace	Stav se změní na „Probíhá“, Počítač se spustí a rozeběhne se automatická instalace
7	Vyčkat na dokončení operace	Stav se změní na „Dokončeno“, Počítač je vypnut
8	Manuální spuštění počítače	Počítač se začne spouštět s výchozím zavaděčem
9	Výběr nainstalovaného OS	Začne se načítat nainstalovaný OS
10	Vyčkat na naběhnutí systému	Zobrazení přihlašovací obrazovky
11	Přihlášení do sytému	Jsme přihlášeni v systému s nainstalovanými aplikacemi odpovídajícími zvolenému profilu

F.6 Instalace GNU/Linuxu – alternativní profil s chybou

Test automatické instalace GNU/Linuxu z alternativního profilu – chybový průběh

Krok	Operace	Očekávaný výsledek
1	Přihlášení do webové aplikace	Zobrazení úvodní obrazovky
2	Přepnutí na stránku se seznamem počítačů	Zobrazení stránky se seznamem počítačů
3	Vyhledání testovacího počítače	Je zobrazen pouze testovaný počítač
4	Kontrola stavu počítače	Na počítači neprobíhá žádná akce
5	Stisknutí tlačítka pro Instalace GNU/ Linuxu s alternativním profilem	Stav se změní na „Naplánováno“
6	Vyčkat na start instalace	Stav se změní na „Probíhá“, Počítač se spustí a rozeběhne se automatická instalace
7	Odpojení testovacího počítače	Instalace se přeruší
8	Vyčkat na detekci chyby	Stav se změní na „Chyba“

F.7 Instalace MS Windows – výchozí profil

Test automatické instalace MS Windows z výchozího profilu – standardní průběh

Krok	Operace	Očekávaný výsledek
1	Přihlášení do webové aplikace	Zobrazení úvodní obrazovky
2	Přepnutí na stránku se seznamem počítačů	Zobrazení stránky se seznamem počítačů
3	Vyhledání testovacího počítače	Je zobrazen pouze testovaný počítač
4	Kontrola stavu počítače	Na počítači neprobíhá žádná akce
5	Stisknutí tlačítka pro Instalace MS Windows s profilem „Výchozí“	Stav se změní na „Naplánováno“
6	Vyčkat na start instalace	Stav se změní na „Probíhá“, Počítač se spustí a rozeběhne se automatická instalace
7	Vyčkat na dokončení operace	Stav se změní na „Dokončeno“, Počítač je zapnut a je zobrazena přihlašovací obrazovky
11	Přihlášení do sytému	Jsme přihlášeni v systému s výchozí konfigurací

F.8 Instalace MS Windows – výchozí profil s chybou

Test automatické instalace MS Windows z výchozího profilu – chybový průběh

Krok	Operace	Očekávaný výsledek
1	Přihlášení do webové aplikace	Zobrazení úvodní obrazovky
2	Přepnutí na stránku se seznamem počítačů	Zobrazení stránky se seznamem počítačů
3	Vyhledání testovacího počítače	Je zobrazen pouze testovaný počítač
4	Kontrola stavu počítače	Na počítači neprobíhá žádná akce
5	Stisknutí tlačítka pro Instalace MS Windows s profilem „Výchozí“	Stav se změní na „Naplánováno“
6	Vyčkat na start instalace	Stav se změní na „Probíhá“, Počítač se spustí a rozeběhne se automatická instalace
7	Odpojení testovacího počítače	Instalace se přeruší
8	Vyčkat na detekci chyby	Stav se změní na „Chyba“

F.9 Instalace MS Windows – alternativní profil

Test automatické instalace MS Windows z alternativního profilu – standardní průběh

Krok	Operace	Očekávaný výsledek
1	Přihlášení do webové aplikace	Zobrazení úvodní obrazovky
2	Přepnutí na stránku se seznamem počítačů	Zobrazení stránky se seznamem počítačů
3	Vyhledání testovacího počítače	Je zobrazen pouze testovaný počítač
4	Kontrola stavu počítače	Na počítači neprobíhá žádná akce
5	Stisknutí tlačítka pro Instalace MS Windows s alternativním profilem	Stav se změní na „Naplánováno“
6	Vyčkat na start instalace	Stav se změní na „Probíhá“, Počítač se spustí a rozeběhne se automatická instalace
7	Vyčkat na dokončení operace	Stav se změní na „Dokončeno“, Počítač je zapnut a je zobrazena přihlašovací obrazovky
11	Přihlášení do sytému	Jsme přihlášeni v systému s nainstalovanými aplikacemi odpovídajícími zvolenému profilu

F.10 Instalace MS Windows – alternativní profil s chybou

Test automatické instalace MS Windows z alternativního profilu – chybový průběh

Krok	Operace	Očekávaný výsledek
1	Přihlášení do webové aplikace	Zobrazení úvodní obrazovky
2	Přepnutí na stránku se seznamem počítačů	Zobrazení stránky se seznamem počítačů
3	Vyhledání testovacího počítače	Je zobrazen pouze testovaný počítač
4	Kontrola stavu počítače	Na počítači neprobíhá žádná akce
5	Stisknutí tlačítka pro Instalace MS Windows s profilem alternativním profilem	Stav se změní na „Naplánováno“
6	Vyčkat na start instalace	Stav se změní na „Probíhá“, Počítač se spustí a rozeběhne se automatická instalace
7	Odpojení testovacího počítače	Instalace se přeruší
8	Vyčkat na detekci chyby	Stav se změní na „Chyba“