

Distribuce aplikací pomocí nástroje SCCM

Distributing Applications by Using SCCM

Bc. Josef Skulina

Diplomová práce

Vedoucí práce: Ing. Lukáš Vojáček, Ph.D.

Ostrava, 2021

Abstrakt

Tato diplomová práce slouží jako detailní manuál k vytvoření vlastního testovacího, či reálného, prostředí firmy, které je centralizovaně spravované vzdáleně pomocí SCCM serveru.

Na začátku práce se dozvíme o většině funkcionalit, které SCCM nabízí, včetně jeho historie. Jednotlivé funkcionality jsou blíže rozebrány, ovšem největší pozornosti je kladeno na téma Software Library, tedy způsobům, pomocí kterých lze přes SCCM distribuovat jakýkoliv software od driverů, přes běžné aplikace až po aktualizace systému. Po popisu System Center Configuration Manageru jsou popsány typy instalačních souborů, včetně krátké ukázky a historie dané technologie. K vytvoření instalačních balíčků, a tedy zabalení instalátorů, je potřeba skriptovací jazyk schopný konfigurací systémových položek, a proto jsou některé technologie skriptů zmíněny. Poslední část práce obsahuje praktickou ukázkou vytvoření virtuálního testovacího prostředí na platformě VMM, na kterém jsem demonstrovány jednotlivé případy užití aplikačního modelu, package modelu a task sekvence a jsou ukázány další funkce SCCM.

Klíčová slova

Visual Basic Script, VBS, System Center Configuration Manager, SCCM, MSI, EXE, InstallShield, DC, AD, instalační balíček, testovací prostředí, virtualizace, VMM, App-V, infrastruktura

Abstract

This master thesis is used as a detailed guide for a creation of your own test or real environment of any kind of company, that is centralized and remotely managed by SCCM server.

In the beginning we can get information about functionalities that SCCM is providing and also a brief history of it. Each functionality is deeply described, although the main focus is on Software Library that contains ways, how to distribute any kind of software, such as drivers, common applications and also system updates via SCCM. After this chapter we can find a description of different installation files technologies including examples of small pieces of codes and also history of each technology. For a creation of installation package we need any kind of scripting language, that is able to do basic configurations of system components and that does the packaging of installation file, so that is why we can find a mention of few scripting languages. Final part of this thesis is focusing on a creation of virtualized test environment running on VMM with a demonstration of usage of application model, package model and also task sequence and at the end there are shown other features of SCCM.

Key words

Visual Basic Script, VBS, System Center Configuration Manager, SCCM, MSI, EXE, InstallShield, DC, AD, installation package, test environment, virtualization, VMM, App-V, infrastructure

Obsah

Seznam použitých zkratk.....	5
Seznam ilustrací a zdrojových kódů.....	6
1 Úvod	7
2 System Center Configuration Manager.....	8
2.1 Historie a obecný popis	8
2.2 Assets and Compliance.....	9
2.3 Aplikační management.....	9
2.3.1 Package model	9
2.3.2 Aplikační model.....	15
2.3.3 Porovnání obou modelů.....	22
2.4 Softwarové aktualizace	22
2.5 Operační systém.....	23
2.6 Reporting, monitoring a další nástroje.....	23
2.7 Správa politik a GPO	24
2.8 SCCM Features	25
3 Instalační balíček	28
3.1 Typy instalačních souborů a jejich přepínače.....	28
3.1.1 Microsoft Windows Installer	28
3.1.2 Inno Setup	30
3.1.3 NSIS	31
3.1.4 InstallShield	32
3.1.5 App-V.....	33
3.2 Skriptovací jazyky	34
3.2.1 PowerShell	34
3.2.2 Visual Basic Script.....	34
3.2.3 AutoIT.....	35
4 Realizace skutečného prostředí	36
4.1 Virtuální prostředí	36
4.1.1 DC server	36
4.1.2 SCCM server	37
4.1.3 Testovací klienti	40
4.2 Výběr skriptovacího jazyka, nasazení a test balíčku.....	41

4.3	Automatické vytváření balíčků pro aktualizace.....	56
4.4	Linuxové zařízení pod správou SCCM.....	57
5	Závěr.....	60
6	Seznam použité literatury a zdrojů.....	62
7	Seznam příloh.....	64

Seznam použitých zkratek

AD – Active Directory

ADR - Automatic Deployment Rule

aj – a jiné

API – Application Programming Interface

APP-V – Application Virtualization

atd – a tak dále

atp – a tak podobně

BIOS – Basic Input / Output System

CPU – Central Processing Unit

EXE – Executable File

GB – Gigabyte

GHz - Gigahertz

GPO – Group Policy Object

ID – Identifier

IT – Information Technology

JRE – Java SE Runtime Environment

MIF – Management Information Format

MSI – Microsoft Installer

MST – Microsoft Transform

např – například

NSIS – Nullsoft Scriptable Install System

OS – Operating System

PHP – Personal Home Page, nebo také Hypertext Preprocessor

PL/SQL – Procedural Language/Structured Query Language

RAM – Random-access Memory

SCCM – System Center Configuration Manager

SMS – Systems Management Server

SQL – Structured Query Language

SSRS – SQL Server Reporting Services

TB – Terabyte

tzv – takzvaný

USB – Universal Serial Bus

UUID - Universally Unique Identifiers

VBA – Visual Basic for Applications

VBS – Visual Basic Script

WMI – Windows Management Instrumentation

WSH – Windows Script Host

Seznam ilustrací a zdrojových kódů

Obrázek 1: Package model - General	10
Obrázek 2: Package model – přehled programů.....	11
Obrázek 3: Package model program - General	11
Obrázek 4: Package model - přehled deploymentů.....	13
Obrázek 5: Package model deployment - General.....	13
Obrázek 6: Package model – celkový přehled	15
Obrázek 7: Aplikační model - General Information	15
Obrázek 8: Aplikační model deployment type - General	17
Obrázek 9: Aplikační model deployment – General	21
Obrázek 10: Nastavení virtuálního stroje pfSense.....	36
Obrázek 11: SCCM – Přehledové okno.....	38
Obrázek 12: Process Explorer	40
Obrázek 13: SCCM - Devices	41
Obrázek 14: Scénář nasazení aplikace	44
Obrázek 15: Scénář nasazení aplikace – Vytvoření kolekce.....	46
Obrázek 16: Scénář nasazení aplikace – Deployment General.....	46
Obrázek 17: Scénář nasazení aplikace - Přehled aplikace.....	47
Obrázek 18: Software Center	48
Obrázek 19: Kontrola zakázání automatických aktualizací	49
Obrázek 20: Smart Capture - Welcome	50
Obrázek 21: Smart Capture – Customization.....	51
Obrázek 22: Smart Capture – Okno projektu.....	51
Obrázek 23: Microsoft Application Virtualization Sequencer.....	52
Obrázek 24: Software Center - App-V package.....	54
Obrázek 25: Task Sequence - McAfee.....	55
Zdrojový kód 1: Výpis nápovědy pro msiexec.exe	28
Zdrojový kód 2: Základní instalační kód	28
Zdrojový kód 3: Příklad instalace s několika přepínači	29
Zdrojový kód 4: Ukázka netriviálního kódu.....	30
Zdrojový kód 5: Kód po použití TRANSFORMS přepínače.....	30
Zdrojový kód 6: NSIS přepínač pro změnu instalační složky	31
Zdrojový kód 7: Časté použití NSIS v kombinaci s ExecWait	31
Zdrojový kód 8: InstallShield parametr pro použití INS souboru	32
Zdrojový kód 9: InstallShield parametr pro vytvoření ISS souboru	32
Zdrojový kód 10: InstallShield parametr pro použití ISS souboru.....	32
Zdrojový kód 11: Ukázka funkce CreateKillProcess()	42
Zdrojový kód 12: Obsah souboru /etc/resolv.conf	58
Zdrojový kód 13: Obsah souboru /etc/hosts	58
Zdrojový kód 14: Instalační parametry SCCM klienta	58
Tabulka 1: Návrátové kódy	20

1 Úvod

Cíl této diplomové práce je seznámení se s nástrojem pro centrální správu serverů a koncových zařízení pomocí System center configuration manager, vytvoření vlastního testovacího prostředí, které obsahuje minimálně AD server, SCCM server a jednoho testovacího klienta s operačním systémem Windows a druhého klienta s OS Linux. Ve vytvořeném prostředí je provedena ukázka vytvoření instalačního balíčku a následně jeho nasazení.

V první části se nachází popis funkcionalit SCCM včetně jeho historie. Jednotlivé funkce jsou detailně popsány, ovšem největší důraz u popisu je kladen na část, která se nazývá Software Library. Tato část System center configuration manager je zodpovědná za finální distribuce aplikací, ovladačů či systémových aktualizací, a to několika způsoby. Prvním způsobem je nasazení softwaru jako aplikační model, který začíná vytlačovat a někdy zcela nahrazovat package model, kterým je druhým způsobem nasazení. Posledním je vytvoření vlastní task sekvence, která většinou obsahuje krok, který spouští jiný aplikační nebo package model, ovšem dokáže provést základní instalace i bez nich a obecně se používá na zřetězení aplikací, skriptů, package modelů či jiných logických prvků, jako jsou ověření, zda je počítač připojen na VPN, či nikoliv.

Po této kapitole následuje popis instalačních souborů, zejména jsou rozebrány nejnámější technologie používané v dnešní době jako InstallShield, MSI a App-V. U většiny technologií jsou základní ukázky zdrojových kódů s vysvětlivkami. Následuje kapitola, která se zaměřuje na skriptovací jazyky, které jsou schopné základních konfigurací systému, jako je správa souborů, registrů, práv složek, přístupu do WMI a popř. AD. Mezi takové technologie skriptů patří PowerShell a VBS, který je použit v praktické části této diplomové práce. Obecně jsou ale skriptovací jazyky velice nebezpečný nástroj často používaný hackery a při nedostatečné znalosti mohou vést k fatálním chybám vedoucím k zhroutilí celého počítače nebo v horším případě i prostředí.

V poslední části práce je manuál krok za krokem vedoucí k vytvoření virtuálního prostředí na platformě VMM, které obsahuje PfSense k oddělení interní sítě firmy od námi vytvořené sítě, dále SCCM server, doménový řadič zodpovědný za celou AD, jeden počítač k otestování vytvořeného instalačního balíčku několika způsoby a jednoho Linuxového klienta.

2 System Center Configuration Manager

Následující kapitola popisuje funkce, možnosti a požadavky SCCM jakožto software, který slouží ke správě a konfiguraci koncových zařízení. Dále dojde k zmínění jednotlivých změn, které se během historie vývoje SCCM provedly.

2.1 Historie a obecný popis

Systems Management Server, zkráceně SMS, jak bylo SCCM nazýváno během prvního vydání v roce 1994, se dočkal mnoha změn. Verzi 1.0 bylo možné spravovat MS-DOS, Windows for Workgroups, Windows NT, Macintosh a OS/2 zařízení běžící na Windows NT Serveru, NetWare, LAN Manager a Pathworks síti. 1995 byl rokem vydání druhé verze 1.1, ve které přibyla migrace stávajících systémů na nový Windows 95. Následující rok vyšla další verze 1.2, do které přibyla vzdálená správa přes Remote Control, SNMP a monitorování sítě.

Dalším milníkem byl rok 2007, kdy došlo k přejmenování na System Center Configuration Manager 2007, v kterém byla přidána podpora Windows Vista a Windows Server 2008. Poslední verzí, která v názvu nese i rok vydání, je 2012. Ta jako první vyšla ve dvou typech – Standard a Datacenter. Rozdílem je hlavně cenová kategorie, kde Standard licence na dva roky se pohybovala okolo 1.500\$, kdežto Datacenter vyšel na 3.600\$. Rozdíl v těchto dvou částkách byl zejména kvůli poplatkům za neomezené množství virtuálních strojů běžících na SCCM serveru u Datacenter verze, kdežto Standard mohl mít pouze dva virtuální stroje.

Číslo verze se začalo skládat podle roku a měsíce vydání. V listopadu 2015 proto veřejně vyšla verze 1511, která nabízela podporu Windows 10 a byl zaveden Windows servicing. Následující verze 1602 poskytla řízení mobilních zařízení a také Office 365 servicing. Do prosince 2019 vyšlo dalších mnoho verzí, které přinesly různá vylepšení, opravu chyb a dokonalejší monitorování a správu celého prostředí. Po tomto datu došlo k dalšímu přejmenování SCCM na Endpoint Configuration Manager. Ten je již dostupný pro vzdělávací účely, ovšem v této diplomové práci je použit SCCM s verzí 1802, a to kvůli možnosti správy Linuxových zařízení¹, která byla odebrána ve verzi 1902.

Aktuální verze poskytuje rozšířenou správu a celkovou konfiguraci jakéhokoliv zařízení připojeného do sítě, ve které se SCCM nachází. Mezi podporované systémy patří Microsoft Windows, macOS (Mac OS X) a také systémy mobilních zařízení Windows Phone, Android, iOS (iPhone OS) a Symbian. Mezi hlavní funkce se řadí softwarová distribuce, patchovací management pro aktualizaci operačních systémů, vytváření jakéhokoliv OS image, reportování chyb a kontrolních zpráv či monitorování proběhlých instalací. Jednotlivé funkce jsou rozebrány v podkapitolách níže.²

SCCM pro svou existenci vyžaduje nainstalovaný SQL Server. Podle verze SQL Serveru se liší, kde se může nacházet. Od verze 2012, přes 2014, 2016 až po 2017 v edici Standard a Enterprise může být nainstalován na centrální administrační site, nebo primary site nebo secondary site. U Express edici je možnost omezena pouze na secondary site. Během instalace jakéhokoliv SQL Serveru musí vždy být

¹ TULLOCH, M. *Introducing Windows Server 2012 R2*. 1. vyd. Redmond: Microsoft Press, 2013. ISBN 978-0-7356-8278-8. s. 20.

² TULLOCH, M. - PERRIMAN, S. *Introducing Microsoft System Center 2012 R2: Technical Overview*. 1. vyd. Redmond: Microsoft Press, 2013. ISBN 978-0-7356-8283-2. s. 39.

zvolena collation „SQL_Latin1_General_CP1_CI_AS“. Jediná výjimka je Čína, která používá jiný standart. Před instalací SCCM musí tento SQL server mít nastavenou minimální velikost paměti a AD musí mít dedikovaný účet k přístupům do databáze.³

2.2 Assets and Compliance

První podstatnou částí jsou právě aktiva a dodržování předpisů. Pojem aktivum vyjadřuje všechna aktivní a neaktivní zařízení, všechny uživatelé a jejich skupiny a v neposlední řadě kolekce.⁴ Kolekce je skupina SCCM, která obsahuje buďto uživatele, nebo zařízení. Může být statická, kdy pro členy skupiny platí pravidlo přímého členství, nebo dynamická, kde se členové průběžně mění vůči jiné propojené skupině nebo jiného pravidla. Každý záznam v aktivech musí být unikátní. U kolekcí je nutné určit její limitující kolekci, o jaký typ kolekce se jedná a jak často dochází k jejímu obnovování. Naplnění takové kolekce se provede pomocí přímého vložení záznamu, nebo vytvořením dotazu na AD, nebo vložení jiné kolekce. Compliance se zaměřuje na pravidla jako minimální hardwarové požadavky pro danou aplikaci, na životní cyklus produktu, na politiky antivirů Microsoft Defender a také na míru využívání softwaru v SCCM.

2.3 Aplikační management

Nejdůležitější komponentou SCCM je právě distribuce aplikací. Proč by někdo potřeboval automatizovat instalaci v prostředí? Na tuto otázku je spousta kladných odpovědí. Zamysleme se nad scénářem, kdy IT pracovník v malé firmě o kapacitě 40 koncových zařízeních (počítačů, mobilů) potřebuje nainstalovat například aplikaci Google Chrome. Fyzicky chodí od jednoho zařízení ke druhému a ručně instaluje aplikaci. Jenže druhý den je třeba nainstalovat i Mozilla Firefox. Scénář se opakuje a firma tak přichází o jednoho technika. Jak tuto situaci vyřešit? Co kdyby firma neobsahovala pouze 40 zařízení, ale 400 nebo 4.000 nebo dokonce 40.000 takových zařízení? Odpověď je jednoduchá – použít server, který se bude starat o distribuci takových aplikací plošně. To znamená, že technik nebude muset chodit od jednoho zařízení ke druhému, ale připojí se vzdáleně k tomuto serveru a pomocí něj se dostane na jakékoliv zařízení vzdáleně. Tohle je právě prvotní myšlenka, proč bylo vymyšleno SCCM.

Příběh a důvod distribuce máme za sebou, teď se posuneme dále k jádru věci. Distribuci aplikací můžeme provést dvěma způsoby. Prvním způsobem je nasazení instalačního souboru jako package model, a naopak druhým způsobem je aplikační model.

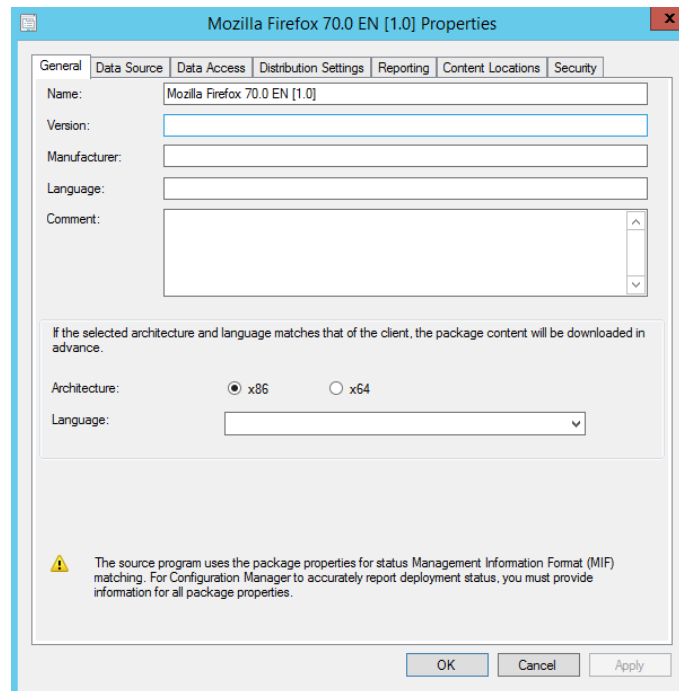
2.3.1 Package model

Tento model je součástí SCCM od původní verze a je zaměřen zejména na distribuci na zařízení. Jeho nevýhodou je, že SCCM nedokáže detekovat, jestli se instalační balíček nainstaloval a splnila se tak podmínka, která určuje, zda balíček je nainstalován korektně. Na druhou stranu může být tato

³Co je Configuration Manager? - Configuration Manager. [online]. Dostupné z: <https://docs.microsoft.com/cs-cz/mem/configmgr/core/understand/introduction/> [cit. 2021-04-24].

⁴MARTINEZ, S. - DAALMANS, P. - BENNETT, B. *Mastering System Center 2012 R2 Configuration Manager*. 1. vyd. Indianapolis: John Wiley & Sons, Inc., 2014. ISBN 978-1-118-82170-1. s. 536.

nevýhoda i výhodou, protože některé aplikace není žádoucí detekovat. Package model je také razantně jednodušší pro administrátory.⁵ Následně jsou popsány jednotlivé části package modelu.



Obrázek 1: Package model - General

Po otevření možností balíčku se zobrazí okno **General**. Toto okno nese informace o daném instalačním balíčku, a to zejména o jakou aplikaci se jedná, její verzi, která firma ji vytvořila, jaký jazyk uživatelského rozhraní obsahuje, popř. můžeme doplnit další komentář.⁶

Dalším oknem je **Data Source**. Zde jednoznačně určujeme, z jaké složky se budou média posílat na distribuční body a odkud budou poté spouštěny pro koncové zařízení. Složkou je ve většině případů myšlen nějaký typ úložiště, často to je úložiště vzdálené. Pokud nedefinujeme tuto cestu, balíček nebude možné nainstalovat. Dále je zde možnost periodicky nastavovat aktualizaci distribučních bodů. Soubory jsou totiž ze specifikované složky automaticky posílány na distribuční body pouze jednou, a to při vytvoření deploymentu. Soubor lze aktualizovat i ručně, ale tato možnost zlepšuje automatizaci v případě, že se soubory často mění. V případech, že chceme poslat soubory přímo na lokální disk a odtud je spustit, musíme aktivovat zaškrťovací políčko „Persist content in the client cache“.

Content Locations list zobrazuje, na kterých distribučních bodech nebo skupinách se balíček aktuálně nachází. V případě, že je balíček uložen na distribuční bodu, můžeme odtud provést odstranění médií, jejich validaci či aktualizaci. Tohle byly podstatná okna nastavení balíčku. Dále se podíváme a popíšeme jeho programy.

⁵ MARTINEZ, S. - DAALMANS, P. - BENNETT, B. *Mastering System Center 2012 R2 Configuration Manager*. 1. vyd. Indianapolis: John Wiley & Sons, Inc., 2014. ISBN 978-1-118-82170-1. s. 372.

⁶*Packages and programs in Configuration Manager*. [online]. Dostupné z: <https://docs.microsoft.com/en-us/configmgr/apps/deploy-use/packages-and-programs/> [cit. 2021-04-21].

Icon	Name	Command Line	Run	Disk Space Requirement	User Description
	Installation	Launcher.exe -i	Normal	Unknown	
	OS_Installation	Launcher.exe -os	Normal	Unknown	
	Uninstallation	Launcher.exe -u	Normal	Unknown	

Obrázek 2: Package model – přehled programů

Každý balíček může obsahovat libovolný počet programů. Program nese informaci o tom, jakým způsobem má SCCM zacházet s poskytnutými médii, které akce se u programu mají vykonat, jak dlouho bude běh trvat a kolik místa na disku balíček zabere.

Installation Properties

OpsMgr Maintenance Mode

General Requirements Environment Advanced Windows Installer

Name: Installation

Your use of software deployed by ConfigMgr may be subject to license terms. You should review any applicable license terms prior to deploying software.

User description:

Command line: Launcher.exe -i

Start in:

Run: Normal

After running: No action required

Category:

OK Cancel Apply

Obrázek 3: Package model program - General

General okno určuje, který konkrétní soubor se má spustit z lokace, kterou jsme definovali v nastavení balíčku, když je tento program vyvolán. Tuto informaci vepíšeme do „Command line“ pole, kde současně nadefinujeme i instalační přepínač, který má být použit. Mimo jiné obsahuje spoustu dalších textových polí a přepínačů:

- **Run:** přepínač určuje, jakou viditelnost má instalační balíček. Na výběr je ze čtyř možností.
 - **Normal:** viditelnost nechává na politikách definovaných přímo klientem SCCM.
 - **Minimized:** zobrazí pouze informační okna nebo upozornění, nebo jiné zprávy předávané do taskbaru.

- **Maximized:** zobrazí všechny instalační aktivity a je vhodný pro testování balíčku.
- **Hidden:** zakryje všechny instalační aktivity, a proto je vhodný pro plně automatizované instalace bez interakce uživatele.⁷
- **After running:** Definuje, co se má odehrát po instalaci balíčku. Na výběr je ze čtyř možností.
 - **No action required:** Scénář, kdy nepotřebujeme restartování systému ani odhlášení uživatele k dokončení instalace.
 - **Configuration Manager restarts computer:** Dojde k vynucení restartu, pokud je nutný k dokončení instalace a samotný instalační soubor jej nevynutí.
 - **Program controls restart:** Instalační soubor může předat informaci o restartování konfiguračnímu manažerovi a popř. provede restart.
 - **Configuration Manager logs user off:** Odhlášení uživatele konfiguračním manažerem, i když to není potřeba.

Requirements okno definuje informace o přibližné velikosti, kterou instalační balíček zabere na disku, a také povoluje, jak dlouho může instalace maximálně trvat. Poslední možností je výběr systémů, které jsou balíčkem podporovány, abychom předešli případné nekompatibilitě aplikace a operačního systému.⁸

Task sequence obsahuje kroky, které se během ní mají vykonat a v překladu znamená pořadí úkolů. Jedná se o kroky jako ověřování, spouštění souborů, restartování systému, podmínky, aj. Dále máme na výběr ze tří možností, v jakém režimu proběhne instalace:

- **Only when a user is logged on:** Případ, kdy je alespoň jeden uživatel připojen, tedy má aktivní relaci, na cílovém zařízení. Tato možnost se hodí v případě, že náš instalační balíček obsahuje akci, která provádí modifikaci souborů nebo registrů aktuálně přihlášeného uživatele, nebo samotná instalace běží pod uživatelem. Jedna z aplikací, která na tomto principu funguje, je Microsoft Teams.
- **Whether or not a user is logged on:** Tímto výběrem může, ale nemusí být uživatel přihlášen. Jedná se o nejčastěji používanou možnost, jelikož většina softwarů není závislá na přihlášení uživatele.
- **Only when no user is logged on:** Scénář, kdy žádný uživatel nesmí být připojen k zařízení. Tato možnost se používá v případě, že je instalační balíček použit v OS task sequence, nebo se jedná o instalaci BIOS aktualizace nebo aktualizace ovladačů. K jednotlivým výše zmíněným případům se pojí i to, jestli balíček má být spuštěn pod právy daného uživatele, nebo se mají použít práva systému – ty nejvyšší.

Často se v praxi stává, že před samotnou instalací aplikace je potřeba mít předinstalované nějaké prekvizity. Nejčastěji se jedná o C++ knihovny, Oracle JRE, ovšem můžeme potřebovat

⁷ MEYLER, K. – HOLT, B. – OH, M. – SANDYS, J. – RAMSEY, G. *System Center 2012 Configuration Manager: UNLEASHED*. 1. vyd. USA: Pearson Education, Inc., 2013. ISBN 978-0-672-33437-5. s. 550.

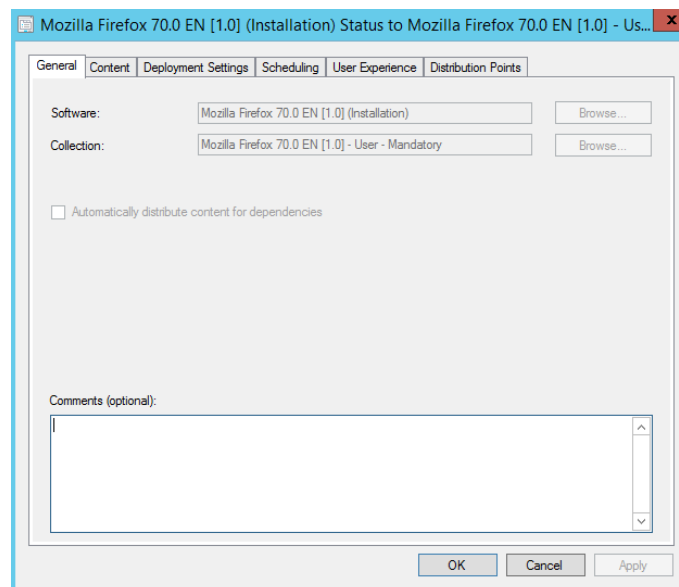
⁸ MEYLER, K. – HOLT, B. – OH, M. – SANDYS, J. – RAMSEY, G. *System Center 2012 Configuration Manager: UNLEASHED*. 1. vyd. USA: Pearson Education, Inc., 2013. ISBN 978-0-672-33437-5. s. 552.

nainstalovat i jiný balíček. K tomu slouží zaškrtačací tlačítko „Run another program first“, které provede danou akci předem.

Icon	Program	Collection	Purpose	Deployment Start Time	Compliance %
	Installation	Mozilla Firefox 70.0 EN [1.0] - User - Man...	Required	2019-11-11 10:41	100,0
	Installation	DWM TEST computers	Available	2019-11-11 11:08	0,0
	Installation	Mozilla Firefox 70.0 EN [1.0] - Device - M...	Required	2020-01-10 09:45	40,5

Obrázek 4: Package model - přehled deploymentů

Pokud chceme balíček dostat na některé klientské zařízení, je třeba pro něj vytvořit deployment. Deploymentů může být několik a jedná se o propojení mezi kolekcí a balíčkem, popř. aplikačním modelem. Obecné okno obsahuje pouze popis, o jaký program se jedná, a také to, na jakou kolekci tento program míří. Je vhodné pojmenovávat balíček a kolekci podobným jménem pro přehlednost prostředí, viz. obrázek.



Obrázek 5: Package model deployment - General

V **Deployment Settings** nastavujeme, jaká akce je způsobena programem a jakým způsobem se má program vykonat na koncovém zařízení:

- **Action:** U balíčků je automaticky zvolen Install, kdežto u aplikačního modelu může být zvolen i Uninstall, jelikož jsou definované jak kroky pro instalaci, tak pro odinstalaci.
 - **Install:** Dojde k provedení instrukcí pro instalaci.
 - **Uninstall:** Dojde k provedení instrukcí pro odinstalaci.
- **Purpose:** Způsob vykonání programů, musí být určen již při vytváření. Když je deployment vytvořen, není možné způsob změnit a tento deployment musí být smazán a musí být vytvořen nový.
 - **Required:** Způsobí to, že instalace nebo odinstalace instalačního balíčku bude provedena bez rozhodnutí uživatele.
 - **Available:** Dá možnost uživateli si dobrovolně balíček nainstalovat ze Software Center.

- **Deploy automatically according to schedule whether or not a user is logged on:** Tlačítko, které nám umožní předem nakopírovat instalační balíček na lokální disk primárního zařízení uživatele ještě předtím, než je distribuce naplánována.
- **Send wake-up packets:** Zaslání wake-up packetu v případě, že je zařízení vypnuto.
- **Require administrator approval if users request this application:** Aktivuje schvalování instalací vybraných instalačních balíčků administrátorem.

Pomocí okna **Scheduling** nastavujeme, kdy se má balíček zobrazit jako dostupný k instalaci v Software Center, od jakého data má být přerušeno deployment z důvodu expirace a také kdy se má instalace provést. Buďto specifikujeme konkrétní datum a čas, nebo můžeme vybrat možnost ASAP, nebo provedení v okamžiku, kdy dojde k odhlášení nebo přihlášení uživatele. Dále můžeme nastavit, co se má stát, když se instalace dokončí.

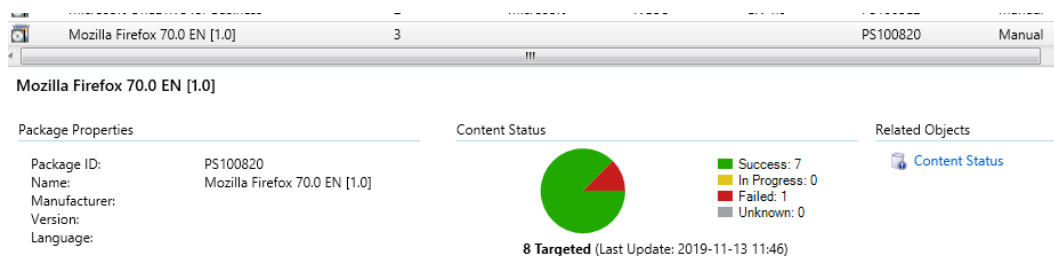
- **Rerun behaviour:** Výběrové menu, kde zvolíme, co se má stát po znovu spuštění deploymentu.
 - **Never rerun advertised program:** Nikdy nespustí znovu stejný deployment.
 - **Always rerun program:** Vždy znovu spustí deployment, i když skončil úspěšně či nikoliv. Dojde k zacyklení.
 - **Rerun if failed previous attempt:** Spustí deployment znovu v případě, že došlo k neúspěchu během instalace.
 - **Rerun if succeeded on previous attempt:** Poslední možností je spuštění, pokud program v předchozím běhu byl úspěšný.

User Experience okno dovoluje, nebo naopak zakazuje, jaké chování je pro uživatele povoleno – zda může dojít k instalaci dalšího balíčku v případě, že již jeden balíček čeká na restart, nebo jestli může dojít k instalaci mimo dobu aktivního používání.

Posledním oknem **Distribution Points** určíme, jestli instalační balíček má být spuštěn rovnou z distribučního bodu, nebo jestli má být stažen lokálně na disk uživatele a až poté spuštěn právě z disku.⁹ Mezi další nastavení patří:

- **Allow clients to share content with other clients on the same subnet:** Tlačítko umožňuje sdílení instalačního balíčku, který je lokálně uložen u jednoho uživatele, pro jiného uživatele v síti, tzn. nedojde ke stažení z distribučních bodů, ale právě z disku prvního uživatele, tzn. první uživatel se stane distribučním bodem pro jiného.
- **Allow clients to use a fallback source location for content:** Pokud média instalačního balíčku nejsou dostupná v rámci hraniční skupiny daného zařízení, dojde k hledání médií v jiných skupinách.

⁹ MEYLER, K. – HOLT, B. – OH, M. – SANDYS, J. – RAMSEY, G. *System Center 2012 Configuration Manager: UNLEASHED*. 1. vyd. USA: Pearson Education, Inc., 2013. ISBN 978-0-672-33437-5. s. 535.

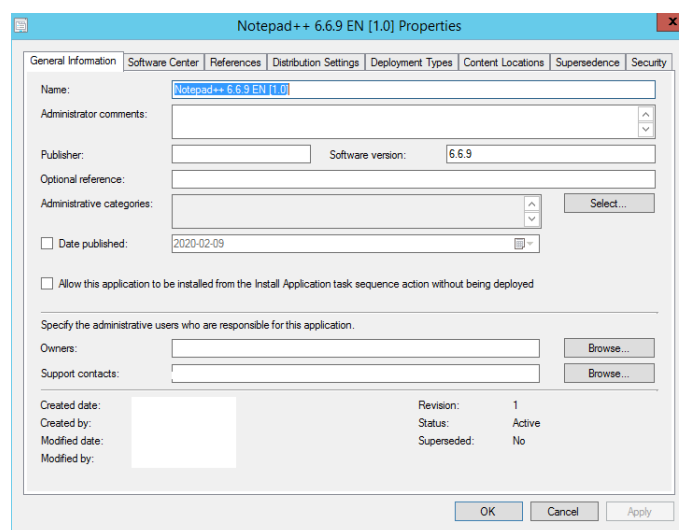


Obrázek 6: Package model – celkový přehled

Když je balíček vytvořen a má alespoň jeden program, který má nastavený deployment na libovolnou kolekci, v záložce **Summary** uvidíme koláčový graf, který znázorňuje, jak úspěšný byl přenos dat ze specifikovaného úložiště na distribuční body. Na obrázku vidíme, že přenos proběhl úspěšně v sedmi případech a v jednom bohužel zhasoval. Důvodem může být plná kapacita tohoto jednoho bodu, nefunkční připojení, nebo bod může být odpojen od elektřiny. Každý balíček má svůj „Package ID“, který je pro něj jedinečný.

2.3.2 Aplikační model

Model se v rámci SCCM objevil až v roce 2012. Distribucí použitou pro aplikační model jsou uživatelé, a to zejména z důvodu, že je tento model mnohem více uživatelsky přívětivý. Detailněji rozlišuje návratové instalační kódy a chybové hlášky, může nastavit povolení instalace balíčku administrátorem, obsahuje více popisků v Software Center a také dokáže pomocí detekčního pravidla rozpoznat, zda je balíček nainstalovaný správně.¹⁰



Obrázek 7: Aplikační model - General Information

V ukázce aplikačního modelu byl připraven software Notepad++. V obecném okně zadáváme název balíčku – pozor, ten musí být jedinečný a pokud není, SCCM nás upozorní a neumožní vytvořit duplicitní model. Dále verzi softwaru a jeho vydavatele, datum publikování na SCCM, je zde možnost

¹⁰ MEYLER, K. – HOLT, B. – OH, M. – SANDYS, J. – RAMSEY, G. *System Center 2012 Configuration Manager: UNLEASHED*. 1. vyd. USA: Pearson Education, Inc., 2013. ISBN 978-0-672-33437-5. s. 565-566.

povolání instalace aplikace rovnou z task sequence a v poslední řadě obsahuje informace o tom, kdo a kdy tento aplikační model vytvořil.¹¹

Software Center okno je extrémně důležité, jelikož koncový uživatel uvidí informace zde vyplněné ve svém Software Centru na jeho zařízení. Výhodou oproti package modelu je přidání vlastní ikonky.

- **Selected language:** Umožňuje změnit názvy popisků do jiného jazyka, v praxi je k dispozici většinou pouze jeden jazyk, protože výchozí jazyk je dán globálním nastavením SCCM.
- **Localized application name:** Lokální aplikační jméno bude zobrazeno jako hlavní název softwaru v Software Centru.
- **User categories:** Pomáhá rozlišovat, do které kategorie software spadá. Může se jednat např. o internetový prohlížeč, nástroj pro zpracování fotek, aj.
- **Localized description:** Popisek, kterým kupříkladu přiblížíme, k čemu se software používá, jestli obsahuje licenci nebo pro které uživatele se hodí.¹²

Přehledná tabulka zobrazující vazby na tento model se skrývá v okně **References**, kde můžeme na jednom místě zobrazit buďto supersedované modely, nebo aplikace, na kterých je tento model závislý.

Nejmocnější nástroj aplikačního modelu je supersedence. Abychom si přiblížili, co se pod slovíčkem supersedence skrývá, musíme si nejdříve vysvětlit, co se běžně v prostředí stává. Představme si situaci, kdy do celého prostředí s jednotkami tisíc počítači byla poslána verze 6.6.0 Notepadu++, bylo splněno detekční pravidlo tohoto modelu a my se rozhodli poslat do prostředí novou verzi 6.6.9. Nová verze odinstalovala starou a narušila její detekční pravidlo. Model staré verze zaznamenal, že detekční pravidlo není splněno, a tak začal znovu instalovat starou verzi, která odstranila tu novou. Model nové verze zaznamenal, že detekční pravidlo není splněno, a tak začal znovu instalovat novou verzi. Došlo k zacyklení, z kterého není konce. Takhle chyba nastane pouze v případě, že oba deploymenty obou modelů míří na stejnou kolekci, tedy stejné uživatele nebo zařízení. To se běžně stává, protože předchozí verze je v produkci a nová verze jde většinou na testery, kteří mají za úkol balíček potvrdit nebo vrátit s návrhy dalších modifikací. A právě tito testeři jsou často afektováni smyčkou popsanou výše. Nastavení supersedence zamezí této smyčce, jelikož u nového modelu nastavíme, že nahrazuje ten starý. Problém je vyřešen, protože pokud uživatel nebo zařízení je v obou kolekcích, použije se nová verze, pokud jen v kolekci pro starou, použije se stará a pokud jen v kolekci pro novou, použije se nová verze. V okně se vybírá model, který má být supersedován, jeho deployment type, deployment type nové verze a také jestli se má použít odinstalace předchozí verze, nebo jestli se nemá odinstalace řešit z důvodu, že instalace nové verze počítá s odinstalací předchozích verzí.¹³

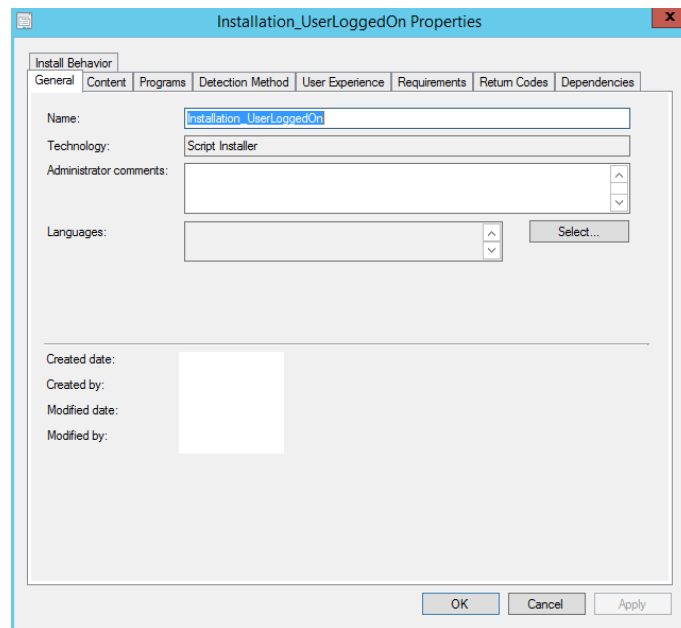
¹¹ *Create applications in Configuration Manager*. [online]. Dostupné z: <https://docs.microsoft.com/en-us/configmgr/apps/deploy-use/create-applications/> [cit. 2021-04-21].

¹² MEYLER, K. – HOLT, B. – OH, M. – SANDYS, J. – RAMSEY, G. *System Center 2012 Configuration Manager: UNLEASHED*. 1. vyd. USA: Pearson Education, Inc., 2013. ISBN 978-0-672-33437-5. s. 578.

¹³ MARTINEZ, S. - DAALMANS, P. - BENNETT, B. *Mastering System Center 2012 R2 Configuration Manager*. 1. vyd. Indianapolis: John Wiley & Sons, Inc., 2014. ISBN 978-1-118-82170-1. s. 296.

Aplikační model podporuje spousty typů tohoto modelu:

- Windows Installer (*.msi file)
- Windows app package (*.appx, *.appxbundle, *.msix, *.msixbundle)
- Windows app package (in the Windows Store)
- Microsoft Application Virtualization 4
- Microsoft Application Virtualization 5
- Windows Phone app package (*.xap file)
- Windows Phone app package (in the Windows Phone store)
- App Package for iOS (*.ipa file)
- App Package for iOS from App Store
- App Package for Android (*.apk file)
- App Package for Android on Google Play
- Mac OS X
- Web Application
- Windows Installer through MDM (*.msi)



Obrázek 8: Aplikační model deployment type - General

Tak jako package model obsahoval programy, aplikační model obsahuje deployment type. Zásadní rozdíl je v tom, že jeden program obsahuje jeden příkaz, kdežto jeden deployment type dokáže pojmout až tři příkazy. V obecném okně pouze definujeme název a technologii tohoto deployment typu a můžeme si zobrazit, kdo a kdy jej vytvořil. Mezi možné technologie patří:

- **Microsoft Application Virtualization:** Vhodné při použití aplikačního modelu, který je typově App-V.
- **Windows Installer:** Využití při jednoduchém aplikačním modelu s MSI souborem.
- **Script Installer:** Při komplexnějším řešení, kdy jednoduché soubory nejsou dostatečné a je nutné použít např. VBS, PowerShell a jiné skriptovací jazyky.

- **Windows Mobile Cabinet:** Primárně při posílání instalačních balíčků na mobilní zařízení, které nejsou spravovány přes Exchange server.
- **Nokia SIS File:** Pro instalační balíčky na Nokia Symbian, které obsahují Nokia Symbian Installation Source (SIS).

Další okno **Content** definuje, kde se nacházejí instalační média. V mnoha případech se jedná o sdílený disk – share, který je viditelný v celé AD, ovšem ne všichni mají práva z něj číst.

- **Uninstall content settings:** Přepínačem můžeme nastavit jinou lokaci odinstalačních médií. V praxi jsou instalační a odinstalační média v jedné lokaci kvůli přehlednosti.
- **Persist content in the client cache:** Slouží k trvalému uložení médií aplikace v cache paměti cílového počítače.¹⁴ SCCM musí obsahovat politiku, která definuje, kolik paměti může cache paměť obsahovat. Většinou se jedná o pět nebo deset GB. V případech, kdy nedochází k čištění této dočasné složky, nebo se složka přeplní většími médii, instalace nové aplikace nemá dostatek místa. Představme si cache, která již má 9,8 GB zaplněno a limit SCCM je nastaven na 10 GB. Dojde k poslání aplikace, která celkem zabírá 500 MB. Deployment předá informaci SCCM o nedostatku místa a SCCM začne promazávat cache. Záleží na nastavení SCCM, podle kterých priorit začne mazat. Většinou začne s mazáním nejstarších médií. K čemu tedy slouží tohle zaškrtačkové tlačítko? SCCM nebude během nedostatku místa mazat z cache ty aplikace, které mají toto tlačítko zaškrtnuto.
- **Allow fallback source location for content:** Povoluje, jestli může dojít ke stažení souborů z výchozího distribučního bodu, což bývá většinou primary site – tedy server, na kterém běží samotné SCCM.

Prioritně se vždy stahuje z distribučních bodů, v kterých se klientský počítač nachází, popřípadě ze sousedních distribučních bodů. Pokud média nejsou nalezena ani v jednom ze zmíněných případů výše, stahování by zhavarovalo. Proto existuje třetí možnost, a to stáhnout média z primary site, kde se média nacházejí vždy a taky by tato site měla být v provozu 24 hodin denně. Proč využít toto dělení? Zvažme zákazníka, který má pobočky po celém světě. Sídlo této firmy a její primary site se nachází ve Finsku. Zaměstnanec z USA si chce nainstalovat balíček ze Software Centra, a tak by mělo dojít k stažení médií z primary site z Finska. V tomto případě dochází k zatížení sítě a k menší rychlosti stahování. Proto si nechala firma vybudovat další distribuční bod přímo v USA. Tím dojde k stažení z bližšího distribučního bodu a síť nebude tak zatěžována. Poslední výběrové menu určuje, jestli má dojít k stažení médií ze sousedních a výchozích bodů, nebo se média mají spustit vzdáleně.

V okně **Programs** se definují příkazy, které se vykonají při instalaci a odinstalaci. SCCM počítá s tím, že soubory jako bat, cmd, vbs atp., se nachází ve složce, kterou jsme uvedli do „Content location“. Pokud tomu tak není, deployment zhavaruje. Možnost opravy aplikace se objevila u verze SCCM 1810 a ulehčuje práci zejména u konečných uživatelů, na které míří mandatory deployment, jelikož si sami můžou opravit nefunkční aplikaci.

¹⁴ MEYLER, K. – HOLT, B. – OH, M. – SANDYS, J. – RAMSEY, G. *System Center 2012 Configuration Manager: UNLEASHED*. 1. vyd. USA: Pearson Education, Inc., 2013. ISBN 978-0-672-33437-5. s. 584.

Detection method okno představuje hlavní výhodu aplikačního modelu oproti package modelu. SCCM je pomocí detekce schopno ověřit, zda se instalační balíček nainstaloval správně, či nesprávně – bylo / nebylo splněno detekční pravidlo. Microsoft nabízí tři možnosti, jak toto pravidlo ověřit. Jednotlivé možnosti jsou rozebrány níže. Pokud by ovšem nestačily tyto tři ověření, je zde ještě čtvrtá, nejkompexnější, metoda, a to vložení vlastního skriptu. Na výběr je ze tří skriptovacích jazyků – PowerShell, VBScript a JScript.

První možností je pravidlo na existenci souboru nebo složky. Pokud nám nestačí pouze existence složky či souboru, můžeme blíže specifikovat buďto datum vytvoření, nebo datum poslední modifikace. Druhou možností je detekce dle registru, kde je potřeba specifikovat, o který konkrétní klíč a hodnotu se jedná. U hodnoty se musí určit její typ a můžeme pomocí logických operátorů blíže určit, jaké data má hodnota obsahovat. Na výběr máme z pěti větvi:

- **HKEY_CLASSES_ROOT:** Obsahuje informace o přidružení přípon souboru, stejně jako programový identifikátor (ProgID), ID třídy (CLSID) a data ID rozhraní (IID).
- **HKEY_CURRENT_CONFIG:** Sám neobsahuje žádné informace, na místo toho funguje jako ukazatel nebo zástupce na klíče v registru, které právě uchovávají informace o aktuálně používaném hardwarovém profilu.
- **HKEY_CURRENT_USER:** Obsahuje konfigurační informace pro Windows a jednotlivé aplikace vůči aktuálně přihlášeného uživatele.
- **HKEY_LOCAL_MACHINE:** Nese informaci o nainstalovaném softwaru a také o samotném operačním systému Windows. Kromě konfiguračních dat obsahuje také informace o aktuálně zjištěném hardwaru a ovladačích zařízení.
- **HKEY_USERS:** Obsahuje informace o konfiguraci specifické pro všechny aktuálně aktivní uživatele v počítači – přihlášené i odhlášené.

Poslední nejjednodušší možností je detekování pouze produktového kódu MSI souboru. To se hodí, když aplikace nepoužívá žádné skripty, ale jen obyčejný instalační soubor. Opět je zde další možnost, a to logické porovnání verze softwaru, které MSI nainstaluje. Tato informace je totiž uložena v jedné z vlastností tohoto typu souboru.

Instalace samotná může být spuštěna v několika režimech, které volíme v **User Experience** okně. Mezi základní tři patří instalace pod aktuálně přihlášeným uživatelem, instalace pod lokálním administrátorem, nebo pod systémovým profilem.¹⁵ Rozdíly mezi těmito režimy jsou jednotlivá práva, které mají výše uvedené účty. Aktuálně přihlášený uživatel bude mít práva nejmenší a systémový profil nejvyšší. Pomocí SCCM můžeme zvolit, který z profilů se má pro instalaci balíčku zvolit. Na výběr máme právě z těchto dvou, popř. ještě existuje třetí možnost, a to při deploymentu na stroje zvolit instalaci pod systémem, a naopak při deploymentu na uživatele instalace pod uživatelem. Nedílnou součástí instalace je čas, kdy má k instalaci dojít. Na výběr máme buďto když je uživatel odhlášený, nebo když je uživatel přihlášený, nebo na tom nezáleží. Pokud je instalační soubor schopen tzv. tiché instalace, není potřeba žádné uživatelské interakce. V případě, že instalátor není schopen sám vykonat instalaci

¹⁵ MEYLER, K. – HOLT, B. – OH, M. – SANDYS, J. – RAMSEY, G. *System Center 2012 Configuration Manager: UNLEASHED*. 1. vyd. USA: Pearson Education, Inc., 2013. ISBN 978-0-672-33437-5. s. 587.

bez zásahu uživatele, je možné použít skriptovací jazyk sloužící k automatizaci instalací. Jedním z takových jazyků je AutoIT, který je probrán v kapitole níže. Dalšími parametry jsou:

- **Installation program visibility:** Tímto výběrovým menu vybíráme viditelnost instalace pro koncového uživatele.
 - **Normal:** Výchozí viditelnost závislá na systému a nastavení programu.
 - **Minimalized:** Zobrazí běh programu pouze během instalace v taskbaru, v kterém je zobrazeno okno, ale toto okno již není maximalizované uživateli.
 - **Maximized:** Vhodné pro testování balíčků a aplikačních modelů, jelikož dojde k zobrazení instalačního okna uživateli, nebo také když je nutná interakce uživatele.
 - **Hidden:** Zcela zakryje instalaci, vhodné pro plně automatické instalace balíčků.
- **Allow users to view and interact with the program installation:** Dává uživatelům během instalace práva konfigurovat tuto instalaci v případě, že instalace neběží v tichém režimu. Konfigurací je myšlena například změna instalační složky, během které se dá vytvářet i mazat další složky, a proto tato možnost je Microsoftem označena jako nedostatečně bezpečná, jelikož se dá zneužít hackery.
- **Maximum allowed run time (minutes):** Pole k určení maximální doby instalace.
- **Estimated installation time (minutes):** Pole k určení odhadované doby instalace.

Pokud dojde během instalace k překročení limitu maximální doby, deployment zhavaruje. Estimovaná doba je pouze viditelný údaj v software centru pro uživatele. Poslední možnost udává, jakým způsobem se má řešit předávání návratového kódu. SCCM může přenechat řízení restartu počítače právě na instalační soubor, nebo může vynutit povinný restart po instalaci, nebo nejčastěji používaná varianta, a to rozhodnout dle konkrétního kódu. Jednotlivé návratové kódy jsou popsány v tabulce níže.

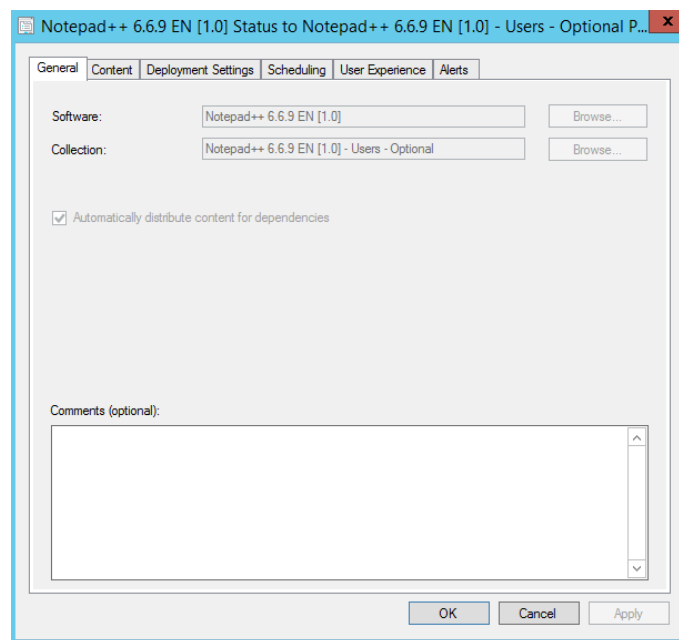
Tabulka 1: Návratové kódy

Hodnota návratového kódu	Vysvětlení
0	Akce dopadla úspěšně.
87	Jeden z parametrů je neplatný.
1602	Uživatel přerušil instalaci.
1603	Závažná chyba během instalace.
1612	Zdroj instalace tohoto produktu není k dispozici. Ověřte, zda zdroj existuje a zda k němu máte přístup.
1618	Jiná instalace aktuálně probíhá. Před pokračováním dokončete tuto instalaci.
3010	K dokončení instalace je nutné restartovat systém.

Pro provedení instalace, odinstalace nebo opravy aplikace můžeme definovat požadavky, které musí být splněny na cílovém počítači a udávají se v okně **Requirements**. Může se jednat o velikost fyzické paměti, operační systém, velikost místa na disku, rychlost CPU, organizační jednotka, v které se počítač nachází, nebo si můžeme definovat vlastní požadavek.¹⁶

Návratové kódy ovlivní, co se má stát po dokončení instalace. Instalační soubor tak předává informaci SCCM, jak dopadla instalace a je už pouze na něm, jaké další kroky podnikne. V nabídce se nachází pět variant, které můžeme definovat pro jakýkoliv výstupní kód. První variantou je označení instalace jako úspěšná bez restartu počítače, dále selhání instalace bez restartu, rychlé zopakování pokusu o instalaci, které se často provádí, když již jedna instalace probíhá, a nakonec dva typy restartu. Tvrdý restart zamezí instalaci dalších aplikací, dokud nebude počítač restartován. Jemný restart toto omezení opomíjí. Čas, za jak dlouho dojde k restartu, je dán globální politikou SCCM.

Posledním oknem pro nastavení deployment typu je okno **Dependencies**, tedy závislostí. Jedná se o jiné, již nasazené aplikace na SCCM, které musí být nainstalované na cílovém počítači před instalací naší aplikace. Často se jedná o prerekvizity jako Microsoft Visual C++, Oracle Java Runtime Environment, Microsoft Visual Studio Tools for Office Runtime, Adobe Acrobat a jiné. Ovšem pomocí téhle možnosti můžeme zřetěžit jakékoliv aplikace.



Obrázek 9: Aplikační model deployment – General

General okno obsahuje pouze informace o tom, o jakou aplikaci se jedná a také na kterou kolekci tento deployment míří. Dále je volitelně možno dopsat komentář. Okno **Content** obsahuje list všech distribučních bodů, na které byl obsah aplikace poslán.

V nastavení deploymentu určíme, který z programů deployment typu se má použít. Na výběr je pouze instalace a odinstalace. Možnost opravy můžeme přidat pomocí zaškrtnutí tlačítka.

¹⁶ MEYLER, K. – HOLT, B. – OH, M. – SANDYS, J. – RAMSEY, G. *System Center 2012 Configuration Manager: UNLEASHED*. 1. vyd. USA: Pearson Education, Inc., 2013. ISBN 978-0-672-33437-5. s. 590.

Scheduling okno umožňuje poslat aplikaci na klientský stroj až v námi definovaný čas. Na výběr máme plánování pro aplikaci, která byla poslána jako volitelná, a proto od data uvedeného bude viditelná v Software Centru. Pro povinné aplikace vypadá okno jinak, a to zejména v dalších možnostech plánování. Mimo času viditelnosti přibude i datum expirace deploymentu a čas, od kterého musí být aplikace nainstalována.

Mnoho počítačů s operačním systémem Windows 10 má lokálně nastavené okno aktivního použití během kterého může dojít k instalaci softwaru, aktualizací systému či patchování, nebo se tento čas nastaví pomocí politiky. Mimo něj tyto akce nejsou možné a musí se čekat do dalšího dne. SCCM ovšem umožňuje alespoň nainstalovat aplikace. Toho se hojně využívá při masových distribucích do celých firem během tzv. „maintenance service break“. Tyto a další nastavení se nastavují v **User Experience** okně.¹⁷

2.3.3 Porovnání obou modelů

Mezi hlavní rozdíly obou modelů patří zejména detekce, zda aplikace je nainstalovaná či nikoliv. Tuhle možnost package model neobsahuje. Dalším k zmínění hodným rozdílem je supersedence, tedy definování, že novější verze má nahradit předchozí verzi. Package model se hodí více při poslání na počítače, kdežto aplikační model při distribuci na uživatele. K tomu se pojí i viditelnost jednotlivých instalačních balíčků v Software Centru. U aplikace totiž můžeme přidat více popisků, a hlavně nahrát ikonu viditelnou pro koncového uživatele.

2.4 Softwarové aktualizace

Konfigurační manažer poskytuje set nástrojů a zdrojů, pomocí kterých lze spravovat komplexní úlohy při zjišťování a aplikování aktualizací klientských počítačů v enterprise prostředí. Prvním nezbytným krokem je správně nastavená synchronizace softwarových aktualizací vlastního konfiguračního manažeru s Microsoft aktualizacemi. To může být provedeno pomocí plánování např. jednou za týden, nebo synchronizaci lze provést ručně v daný okamžik. Ta se provede vždy nejprve pro top-level vrstvu prostředí (primary site, terminal site) a až poté pro nižší vrstvy. Které aktualizace se mají vybrat se rozhoduje v **Software Update Point Component** a rozhoduje se pouze pro nejvyšší vrstvu prostředí. U nižších vrstev dochází k replikaci databáze vrstvy vyšší a poté se zahájí instalace těch aktualizací, které obdržela top-level vrstva.¹⁸

Když jsou aktualizace nainstalované na primary site a nižší vrstvách, může dojít k instalaci i na klientských počítačích. Před tím, než k samotné instalaci dojde, jsou vždy obnoveny politiky a je proveden sken softwarových aktualizací. Ve zkratce si SCCM pomocí query na WMI jednotlivých počítačů zjistí, jestli daná aktualizace byla provedena, či nikoliv. Tím se předejde přehlcení síťové kapacity, jelikož jsou staženy jen potřebné aktualizace. Aktualizace se může nacházet v několika stavech:

- **Required:** Dává najevo, že aktualizace ještě není nainstalovaná, nebo nainstalovaná je, ale k dokončení je třeba restartovat počítač, či znovu vynutit sken aktualizací.

¹⁷ MEYLER, K. – HOLT, B. – OH, M. – SANDYS, J. – RAMSEY, G. *System Center 2012 Configuration Manager: UNLEASHED*. 1. vyd. USA: Pearson Education, Inc., 2013. ISBN 978-0-672-33437-5. s. 657-658.

¹⁸ MARTINEZ, S. - DAALMANS, P. - BENNETT, B. *Mastering System Center 2012 R2 Configuration Manager*. 1. vyd. Indianapolis: John Wiley & Sons, Inc., 2014. ISBN 978-1-118-82170-1. s. 257.

- **Not required:** Říká, že tuto aktualizaci není možné nainstalovat na dané zařízení z důvodu nekompatibility, nebo jiných omezení.
- **Installed:** Tento stav znamená, že aktualizace byla v pořádku provedena.
- **Unknown:** Vyjadřuje několik různých scénářů, které mohly nastat.
 - Sken byl chybně proveden.
 - Sken byl proveden, ale nebyla o tom předána zpráva vyšší vrstvě.
 - Sken byl proveden, ale zpráva byla zcela poškozena.¹⁹

2.5 Operační systém

V kategorii operačního systému v nabídce SCCM nalezneme správu ovladačů, obrazy operačních systémů, task sequence a také boot image. Ovladače lze připravit i v rámci instalačního balíčku, ovšem není to zcela správná cesta, jelikož takové balení se často provádí naslepo, bez fyzické přítomnosti hardwaru.²⁰ Proto dochází k vytvoření oficiálních ovladačů v této kategorii, které zajišťují architekti daného prostředí, kteří fyzicky otestují, zda ovladač funguje, nebo ne. Task sequence obsahují jak sekvence systému, tak sekvence dle vlastních potřeb. Sekvencí je minimálně tolik, kolik prostředí obsahuje verzi systému a jeho architektury. Obecně se využívají k zřetězení více instalací, propojení aplikačního a package modelu a k dalším logickým operacím a ošetřením. Boot image se vytváří právě z OS task sequence a jejím výsledkem je vytvoření ISO souboru pro následné použití např. k vytvoření virtuálního stroje na Hyper-V serveru, který může být spravován pomocí System Center Virtual Machine Manager.²¹

2.6 Reporting, monitoring a další nástroje

Pro zpřehlednění prostředí a zjednodušení získávání dat o uživateli, počítačích a dalších částech spravovaného prostředí je vhodné nainstalovat pokročilé nástroje pro předávání zpráv.²²

- **SQL Server Reporting Services:** Zkráceně SSRS, obsahuje již předem nakonfigurované reporty, které se nejčastěji používají v prostředích, ovšem je zde možnost si rozšířit stávající reporty, nebo si dokonce vytvořit vlastní.
- **Power BI Report Server:** Poskytuje modernější vizualizaci a vyšší výkon než první zmíněný reportovací systém. V základu ovšem obsahuje stejné funkce. Ty nejběžnější reporty si můžeme rozdělit do několika kategorií.
 - **Asset Intelligence:** První kategorie, která se bude starat o zjištění hardwarových informací, jako názvy jednotlivých výrobců notebooků, vyhledání všech primárních zařízení jedno uživatele, vyhledání všech USB

¹⁹Introduction to software updates in Configuration Manager. [online]. Dostupné z: <https://docs.microsoft.com/en-us/configmgr/sum/understand/software-updates-introduction/> [cit. 2021-04-21].

²⁰ MARTINEZ, S. - DAALMANS, P. - BENNETT, B. *Mastering System Center 2012 R2 Configuration Manager*. 1. vyd. Indianapolis: John Wiley & Sons, Inc., 2014. ISBN 978-1-118-82170-1. s. 452.

²¹ TULLOCH, M. *Understanding Microsoft Virtualization Solutions: From the Desktop to the Datacenter*. 2. vyd. Redmond: Microsoft Press, 2010. s. 47.

²² MARTINEZ, S. - DAALMANS, P. - BENNETT, B. *Mastering System Center 2012 R2 Configuration Manager*. 1. vyd. Indianapolis: John Wiley & Sons, Inc., 2014. ISBN 978-1-118-82170-1. s. 16.

zařízení u jednoho nebo více počítačů, vyhledání licencí se specifickým statusem, aj.

- **Device management:** Se soustředí na mobilní zařízení.
- **Driver management:** Pro reporty obsahující seznamy všech ovladačů v prostředí, seznamy všech boot images, výpis všech podporovaných zařízení pro specifický driver, aj.
- Kategorie je spousta a téměř by nestačila ani celá diplomová práce k tomu, aby je bylo možné všechny rozepsat detailněji.²³
- **Monitoring:** SCCM nabízí grafické rozhraní pro zpřehlednění a pro jednodušší odstraňování problémů vzniklých během instalací instalačních balíčků, softwarových aktualizací či jiných deploymentů. Jedná se pouze o čtecí rozhraní, takže nenabízí žádné nástroje, pomocí kterých by šlo ze stejného místa daný problém ihned vyřešit.
- **CMTrace** je nástroj pro zobrazování a monitorování log souborů. Log soubory se opět dělí na několik kategorií. Těmi hlavními jsou klientské, serverové, operačního systému a softwarových aktualizací. Pomocí logů můžeme detailně procházet, které instalační balíčky se nainstalovaly, kdy došlo k updatu politik, zda došlo k Wake-on-LAN, aj.

2.7 Správa politik a GPO

Často se stává, že si zákazník nepřeje platit za další instalační balíček kvůli vysoké ceně, a tak volí raději variantu, kdy dojde k distribuci souboru, složky nebo jiných registrů pomocí GPO. GPO je možné použít u jednoduchých instalací nebo kopírování, pro složitější akce je třeba vytvořit v SCCM balíček. Politiky se řídí a konfiguruje na AD serveru nebo na doménovém řadiči pomocí nástroje Group Policy Management. Politiky se dělí na vrstvy, a proto je při vytváření důležité správně politiku zařadit. Nejprve se totiž aplikuje nejnižší vrstva, která se jmenuje lokální politiky, které si můžeme představit jako jakékoliv nastavení na počítači. Poté dojde k provedení politik podle polohy či AD skupiny, ve které se zařízení nachází. Následuje doménová politika a po ní politika organizačních jednotek. V případě, že jsme v lokální politice nastavili například jako výchozí prohlížeč Google Chrome, doménová politika nařizuje, že výchozí má být Microsoft Edge, ale politika organizační jednotky říká, že to bude Mozilla Firefox, bude jako výchozí nastaven Firefox.

Mezi základní politiky (v SCCM kontextu) patří zejména zákaz instalací pro běžné uživatele, je zakázaný účet hosta, nastavení komplexnosti a povinnosti hesla pro každého uživatele, zákaz připojení přenositelných médií, zákaz vynucených restartů (vše se totiž musí řídit přes SCCM, konkrétně pomocí patchovacího a aplikačního managementu), aj.

²³*Introduction to reporting in Configuration Manager.* [online]. Dostupné z: <https://docs.microsoft.com/en-us/configmgr/core/servers/manage/introduction-to-reporting/> [cit. 2021-04-21].

2.8 SCCM Features

Při výchozí instalaci SCCM serveru je mnoho funkcionalit automaticky zapnuto. Následně si detailněji rozebereme, které to jsou a určitě přijdeme na to, že o některých jsme ani nevěděli, přitom jsou běžnou součástí každého konfiguračního manažeru. Jednotlivé funkcionality jsou popsány v abecedním pořadí a jsou součástí verze 1802.

- **Approve application requests for users per device:** Zviditelní zaškrtačkové tlačítko u deploymentu aplikace a v podstatě zapříčiní to, že aplikace, která byla poslána jako volitelná, musí být schválena administrátorem.²⁴ Prvním krokem je, že koncový uživatel si vyhledá takovou aplikaci v Software Center, klikne na tento záznam a místo tlačítka Install uvidí, že se jedná o aplikaci se schválením. Proto musí vyplnit text s patřičným komentářem a požádat o povolení kliknutím na tlačítko Request. Administrátor, nebo obecně kdokoliv, kdo má přístup do Configuration Manageru a má práva na schvalování žádostí ve složce Software Library a podsložce Application Management v Approval Requests. Tato záložka obsahuje všechny žádosti i s komentáři uživatelů.
- **Feature Client Peer Cache:** Vždy automaticky zapnuta funkcionality, která slouží ke sdílení obsahu instalačních balíčků, aktualizací a jiných souborů a složek z lokální cache složky jednoho zařízení na druhé zařízení. Jedná se o stejný princip, který je používán u torrentů, kde se každé zúčastněné zařízení stává nosičem a zprostředkovatelem médií.
- **Cloud Management Gateway:** Funkcionality poskytující správu klientů na internetu pomocí SCCM, i když nejsou součástí infrastruktury. V jednoduchosti jde o propojení prostředí spravované Configuration Managerem s virtuálními stroji cloudu, které běží na Microsoft Azure. O tom, že firma musí odebírat služby Microsoft Azure není pochyb.
- **Conditional Access for managed PCs:** Poskytuje další úroveň v přístupu zařízení přes Exchange online a SharePoint online. V případě, že tato funkcionality je vypnutá, jedinou možností správy je Microsoft Intune, nebo to, že zařízení je připojeno v doméně.²⁵
- **Create and run scripts:** Povoluje vytváření a vykonávání PowerShell skriptů k úpravě jak Configuration Manageru, tak i koncových zařízení. Součástí je i monitorování těchto skriptů, nahlížení do Active Directory atp.
- **Data warehouse service point:** V případě, že bychom chtěli zpracovávat historická data z dlouhodobého hlediska v přehlednější formě, než která je aktuálně poskytována SCCM serverem, můžeme povolit toto rozšíření, pomocí kterého se tato data budou ukládat do databáze. Z databáze bude poté možné vytvářet buďto výchozí reporty, nebo námi definované.
- **Enable third party update support on clients:** Povolí aktualizace, kterou jsou poskytovány třetí stranou. V praxi se jedná o velice specifický software, jako jsou ovladače od Dellu, Lenova a jiných výrobců notebooků. V SCCM je na výběr ze dvou katalogů, kde prvním je **Partner catalog**, v kterém najdeme software dodavatelů registrovaných u Microsoftu (již zmíněný Dell,

²⁴ MARTINEZ, S. - DAALMANS, P. - BENNETT, B. *Mastering System Center 2012 R2 Configuration Manager*. 1. vyd. Indianapolis: John Wiley & Sons, Inc., 2014. ISBN 978-1-118-82170-1. s. 296.

²⁵ TULLOCH, M. - PERRIMAN, S. *Introducing Microsoft System Center 2012 R2: Technical Overview*. 1. vyd. Redmond: Microsoft Press, 2013. ISBN 978-0-7356-8283-2. s. 44.

Lenovo atp.). Druhým katalogem je **Custom catalog**, do kterého můžeme přidat software dle vlastního uvážení, ovšem s tím omezením, že katalog používá HTTPS a aktualizace těchto katalogů musí obsahovat digitálně podepsané certifikáty. Takový katalog dodává Adobe.

- **Microsoft Operations Management Suite (OMS) Connector:** Dodává možnost importování a synchronizace dat z kolekcí Configuration Manageru do OMS běžícím na Microsoft Azure. Bez předplatného Microsoft Azure tato funkcionality ztrácí význam.
- **Passport for Work:** Nabízí koncovému uživateli možnost přihlášení se do svého zařízení pomocí gesta namísto zadávání hesla. Takovým gestem je myšlen PIN kód, biometrická autentizace nebo přihlášení přes externí zařízení.
- **Phased deployment:** Automatizované a koordinované deploymenty se sekvenčním zaváděním softwaru pro více kolekcí při masových distribucích do celého prostředí. Při distribuci dochází v prvních částech k poslání balíčku na tzv. piloty. Pokud deployment proběhne s určitým procentem úspěšnosti, dojde k zaslání balíčku na další piloty a pokud zde bude instalace úspěšná, dojde k distribuci do celého prostředí. Tomu se říká rozdělení na vlny, kde počet vln záleží na daném zákazníkovi, jelikož někteří jsou opatrnější a někteří méně. Jednotlivé vlny musely být obslouženy manuálně SCCM administrátorem, kdežto phased deployment automatizuje tuto činnost, a to takovým způsobem, že když dojde k úspěšnosti např. 85 % při první vlně, automaticky pošle balíček na druhou vlnu atp. ve stejném duchu.
- **Run Task Sequence Step feature:** Nabízí spuštění jedné task sequence druhou task sequence tak, že první se stává rodičem druhé. Jedná se o princip rodiče a potomka.
- **Task Sequence content pre-caching:** Poskytuje předběžné stažení závislých médií OS task sequence do cache paměti koncového zařízení. Tímto předčasným stažením můžeme regulovat vytiženost sítě, když např. provedeme nakopírování těchto médií v době aktivního použití.

SCCM má velmi silnou vazbu s antivirem Windows Defender. Pro zařízení s operačním systémem Windows 10 je pouze nutné poskytovat aktualizace tohoto antiviru, jelikož se nachází jako již předinstalovaná aplikace. Pro zařízení s nižší verzí OS Windows musíme provést více kroků nastavení na SCCM. Prvním krokem je vytvoření nové role primary site, která se jmenuje Endpoint Protection.²⁶ Druhým bude nastavení upozornění na výskyt např. malwaru, které budou zobrazeny v monitoringu, nebo mohou být zaslány emailem. Dalším krokem je vydefinování zdroje aktualizací pro Endpoint Protection klienty. Poté je nutné specifikovat výchozí antimalware politiky, popř. vytvořit vlastní. Posledním krokem je změna nastavení samotného Endpoint Protection v kontextu specifických modifikací. Všechny výchozí politiky a nastavení najdeme v Configuration Manageru v složce Assets and Compliance a podsložce Endpoint Protection v záložce Antimalware Policies.

²⁶ MARTINEZ, S. - DAALMANS, P. - BENNETT, B. *Mastering System Center 2012 R2 Configuration Manager*. 1. vyd. Indianapolis: John Wiley & Sons, Inc., 2014. ISBN 978-1-118-82170-1. s. 672.

Scénář, který popisuje, co se stane, když je zjištěn virus na některém koncovém zařízení, závisí na tom, jak jsou nastaveny politiky Endpoint Protection. Výchozí nastavení je následující. Dojde ke kategorizaci virusu podle „malware definition file“, který obsahuje čtyři kategorie:

- **Severe**
- **High**
- **Medium**
- **Low**

Při aktualizacích dochází právě k upgradu tohoto souboru. U prvních tří kategorií dojde ke karanténě viru, ale ne k jeho smazání. U kategorie Low se nestane nic a virus je povolen. Tuto politiku je možné změnit. Také můžeme vytvořit upozornění k tomu, když dojde k detekci viru, a toto upozornění bude zobrazeno v Monitoring složce.

3 Instalační balíček

Balíček je název pro sadu instalací a konfigurací, které mohou být upraveny pomocí skriptů, většinou dle přání zákazníka, ovšem může být modifikován i pouze podle standardů poskytovatele této služby. Nejčastějšími modifikacemi jsou nastavení licenčních či databázových serverů, změna jazyka rozhraní aplikace, speciální změny funkcionalit softwaru, aj.

3.1 Typy instalačních souborů a jejich přepínače

Malá část softwarů nepotřebuje instalační proces před spuštěním, jelikož vendor poskytuje již předinstalované a rozbalené soubory. Většina ovšem instalaci potřebuje, jelikož během ní dochází k různým ošetřením, zejména kontrola systému a architektury, práv uživatele k instalaci, aj. Bez instalačních souborů by nemohly vzniknout balíčky. Instalačním souborem je myšlen spustitelný soubor, který může být různé technologie, které jsou popsány níže, nebo jakýkoliv soubor, s kterým je potřeba provést nějakou akci.

3.1.1 Microsoft Windows Installer

Nejpoužívanější formát instalačních souborů, který se používá pro instalaci, upgrade a odinstalaci aplikací. První zmínka se objevila již v roce 2000, kdy byl tento typ instalátoru poprvé použit pro Microsoft Windows 2000 a Microsoft Office 2000. Cílem vytvoření bylo zjednodušení a zpřehlednění instalace. Od té doby se Windows Installer stal nutnou součástí každé firmy, která chtěla mít svůj software oficiálně certifikovaný pro Windows platformy.²⁷

S tímto instalačním souborem lze pracovat pomocí aplikace `msiexec.exe`, která je již předinstalovaná a nachází se ve složce `%SYSTEMROOT%\System32`. Tato aplikace obsahuje spoustu instalačních přepínačů. Náповědu a několik ukázek použití je možné vypsát pomocí příkazového řádku přes následující přepínač:

```
Msiexec.exe /help
```

Zdrojový kód 1: Výpis nápovědy pro `msiexec.exe`

Instalaci pomocí SCCM ovšem chceme tak, aby byla bez interakce uživatele, tedy tzv. silent. Není totiž žádoucí, aby user měl možnost změnit konfiguraci námi definovanou, a také nechceme, aby byl jakkoliv obtěžován. MSI proto nabízí přepínače pro bezobslužnou instalaci „/qn“, „/qb“, „quiet“ a jiné, které se liší viditelností pro koncového uživatele:

```
Msiexec.exe /i <CestaKMSI> /qn
```

Zdrojový kód 2: Základní instalační kód

Pokud instalaci spustíme pouze s tichými přepínači zmíněnými výše, dojde k vykonání instalace tak, jak je definovaná v instalačním souboru. Pro změnu některých parametrů je potřeba

²⁷ *Windows Installer - Win32 apps*. [online]. Dostupné z: <https://docs.microsoft.com/en-us/windows/win32/msi/windows-installer-portal/> [cit. 2021-04-24].

přidat další přepínače. MSI definuje několik základních přepínačů, které musí obsahovat všechny instalátory této technologie. Mezi ně patří zejména:

- **ARPCOMMENTS:** Přidá komentář do záznamu aplikace v ovládacích panelech v položce Programy a funkce, viditelnost komentářů není běžně zobrazena a tento krok se musí vykonat ručně.
- **ARPNOREMOVE:** Zakáže možnost odinstalace pomocí tlačítka Odinstalovat v položce Programy a funkce.
- **ALLUSERS:** Definuje, jestli instalace proběhne pouze pro aktuálně přihlášeného uživatele (dojde k změně instalační složky z %ProgramFiles% na %APPDATA%), nebo pro celý systém (tedy pro všechny uživatele a instalační složka zůstane stejná).
- **REBOOT:** Po instalaci může instalace vynutit restart systému, což je nežádoucí, jelikož restarty se řeší výhradně pomocí SCCM. Tomuto scénáři chceme předejít, proto tento přepínač dokáže zamezit restartu.
- **TARGETDIR:** Nastaví složku, do které bude aplikace nainstalována.

Ukázka instalace, která proběhne v tichém režimu a bude přístupná pro všechny uživatele systému, a která nastaví komentář aplikace, zamezí zobrazení tlačítka pro odinstalaci, zamezí restartu a nastaví instalační složku na „C:\test“:

```
Msiexec.exe /i <CestaKMSI> /qn ARPCOMMENTS="Komentář" ARPNOREMOVE=1  
ALLUSERS=1 REBOOT=ReallySuppress TARGETDIR="C:\InstalačníSložka"
```

Zdrojový kód 3: Příklad instalace s několika přepínači

Aplikace Msiexec nabízí spoustu dalších přepínačů a parametrů na konfiguraci instalátoru. Pokud je nabídka nedostatečná, je možné přidat další vlastní parametry. Z většiny se jedná o definování licenčních serverů, konkrétních licencí, nebo změnu jiných hodnot během instalace. Msiexec samozřejmě nepracuje pouze se soubory a zástupci k aplikacím, které nainstaluje, ale je zde možné přidat registry, servisní služby, knihovny, fonty atd.

Jelikož se v praxi používají spousty parametrů během instalace, dochází k tomu, že se kód příkazového řádku stane nepřehledný. Tenhle problém lze vyřešit vytvořením MST transformačním souborem. Tento soubor obsahuje pouze změny, které se mají aplikovat do MSI instalátoru, a soubor lze použít pomocí příkazu „TRANSFORMS=CestaKMSTSouboru“.

V následujícím kódu jsou demonstrovány výhody MST souboru. Představme si kód příkazového řádku, který není triviální. Název použitého serveru, uživatelského jména, hesla i vzdáleného úložiště byl upraven kvůli možným citlivým informacím:²⁸

```
Msiexec.exe /i SuperOffice8.msi /qn IACCEPTSQLNCLICENSETERMS=YES
REBOOT=ReallySuppress ALLUSERS=1 CONFIG_PROP_ARCHIVEPATH=\\SHARE\so_arc_tv4_dk
CONFIG_PROP_DATABASE=SuperOffice8_TV3_DK CONFIG_PROP_DRIVER="SQL Server Native
Client 10.0" CONFIG_PROP_DATAPATH=ODBC:superoffice8_tv3_dk
CONFIG_PROP_GLOBALPREFIX=CRM78 CONFIG_PROP_ODBCNAME=superoffice8_tv3_dk
CONFIG_PROP_PREF_LANG=SV CONFIG_PROP_PREF_WORDPROCESSOR=Word2010
CONFIG_PROP_PREF_MAILCLIENT=Outlook2010 CONFIG_PROP_SERVER=SERVER\PROD04
CONFIG_PROP_PWD=7C:C7F5/ERSA== CONFIG_PROP_USERID=7C:574J/JWRP44
```

Zdrojový kód 4: Ukázka netriviálního kódu

Nyní dojde k ukázce využití MST souboru, v kterém jsou uloženy změny, které jsou napsány výše:

```
Msiexec.exe /i <CestaKMSI> TRANSFORMS="MujMSTSoubor.mst" /qn
```

Zdrojový kód 5: Kód po použití TRANSFORMS přepínače

Rozdíl přehlednosti obou použitých kódů je viditelný na první pohled. K vytvoření MST souboru je ovšem potřeba speciální software, který dokáže otevřít a modifikovat MSI soubory. Mezi nejvíce používané freeware aplikace patří InstEd It! a mezi komerčně používané je vhodné zmínit Smart Package Studio a Wise. Tyto aplikace ovšem nenabízí pouze otevírání MSI a vytváření MST souborů, ale dokáže i vytvořit zcela nový soubor, buďto ručně nebo pomocí capture funkce. Její princip je takový, že se uloží obraz systému (soubory, registry), provede se instalace a udělá se další obraz. Z rozdílu mezi oběma obrazy se vytvoří právě nový MSI soubor.

3.1.2 Inno Setup

Inno setup je freeware technologie instalátorů, kterou v roce 1997 vyvinuli dva developeri Jordan Russel a Martijn Laan. Je určena zejména pro systém Microsoft Windows a je kompatibilní již se zastaralými verzemi OS NT 4.0, Windows 95 až po dnešní Windows 10.²⁹

Výhodou Inno Setupu je jednoduché pochopení skriptu, který vytvoří finálního instalačního průvodce. Od založení vzniklo spousta dalších rozšíření do původní podoby Inno Setupu, např. My Inno Setup Extensions, který vytvořil již zmíněný Martijn Laan. Klíčové funkce zůstaly zachovány. Přibyla rozšířená podpora pro instalaci 64-bitových softwarů na 64-bitové operační systémy, podpora

²⁸ SKULINA, J. Absolvování individuální odborné praxe. Ostrava, 2018. 30 stran. Bakalářská práce. VŠB-TUO, Fakulta elektrotechniky a informatiky. Vedoucí práce Ing. Dohnálek Pavel.

²⁹ *Inno Setup*. [online]. Dostupné z: <https://jrsoftware.org/isinfo.php> [cit. 2021-04-24].

totální odinstalace, schopnost vytvoření registrů Windows nebo INI souborů, nutnost zadání hesla před instalací, aj.

Všechny přepínače, které smí být během instalace použity, musí být napsány ve skriptu během kompilace. Proto všechny následující parametry mohou, ale nemusí, fungovat. Nejčastějším přepínačem, který slouží k tiché instalaci, je „/SILENT“ a „/VERYSILENT“. Parametrem „/NORESTART“ dojde k zamezení restartu po instalaci nebo odinstalaci. Inno setup také dokáže zavřít jiné aplikace, které během instalace přistupují k souborům, které jsou současně potřebné pro instalaci, a to „/CLOSEAPPLICATIONS“ parametrem. „/DIR“ přepínač změní instalační složku a „/NOICONS“ zakáže vytvoření zástupců aplikace ve Start Menu složce.

3.1.3 NSIS

Nullsoft Scriptable Install System je volně dostupná technologie, sloužící pro instalaci softwaru, od firmy Nullsoft. Důvodem, proč vznikl NSIS, byla distribuce Winampu. Tato technologie podporuje mnohem více kompresních formátů než výše zmíněný Microsoft Windows Installer, také nabízí uživatelské rozhraní v několika jazykových mutacích a je běžně používán jako náhrada za komerční instalátory typu InstallShield. NSIS je navržen tak, aby byl malý a flexibilní, zejména konkurence schopný, a díky těmto vlastnostem vyhrál ocenění „SourceForge’s project of the month“.³⁰

NSIS projekty mohou být modifikovány v textovém editoru, protože jsou skripty napsané v C, C++ nebo Delphi jazyce. To u MSI technologie není možné, jelikož je potřeba sofistikovaný software. Samozřejmě, že tyto skripty k projektům je poté třeba zkompileovat a zvalidovat, což textový editor sám o sobě neumí, a proto byly přidány add-iny do aplikací typu Eclipse (EclipseNSIS), Microsoft Visual Studio IDE či HM NIS Edit.

Po zkompileování dojde k vytvoření EXE souboru. Zpětná editace tohoto souboru není možná. Proto bude blíže rozebrán zejména popis instalačních parametrů, a ne parametrů pro vytvoření a kompilaci nových NSIS souborů. Prvně je nutné zmínit, že NSIS přepínače jsou case-sensitive. Mezi základní přepínač patří „/S“, který provede instalaci bez uživatelské interakce.

Dalším přepínačem je „/D=InstalacniSlozka“, který provede změnu složky, kde bude software nainstalován. Níže je ukázka použití obou přepínačů:

```
MujNSISInstaller.exe /S /D=%ProgramFiles%\MojeSlozka
```

Zdrojový kód 6: NSIS přepínač pro změnu instalační složky

Pro odinstalaci aplikace se musí použít zvláštní přepínač „/S _?=InstalacniSlozka“. Často se tento switch kombinuje s funkcí ExecWait, která čeká, až se vykoná a ukončí spustitelný soubor. Když je odinstalace dokončena, mohou být po ní vykonány další kroky.

```
$INSTDIR=%ProgramFiles%\MojeSlozka  
ExecWait '„$INSTDIR\MujNSISUninstaller.exe" /S _?=$INSTDIR'
```

Zdrojový kód 7: Časté použití NSIS v kombinaci s ExecWait

³⁰ NSIS Users Manual. [online]. Dostupné z: <https://nsis.sourceforge.io/Docs/> [cit. 2021-04-24].

3.1.4 InstallShield

Softwarová technologie vytvořena v roce 1990 firmou The Stirling Group. Ta se během své historie mnohokrát přejmenovala až do dnešního názvu Flexera Software. Nástroj InstallShield slouží k vytvoření instalátorů a softwarových balíčků, které jsou hlavně cíleny na platformy Microsoft Windows a serverové platformy, ovšem použití zasahuje až na různá mobilní zařízení. Aktuálně nejnovější verzí je verze InstallShieldu z roku 2020. Z důvodu, že je tento instalátor hned po MSI technologii nejrozšířenější, jsou zde blíže popsány jeho instalační přepínače.³¹

InstallShield instalační soubor vždy musí obsahovat jeden EXE soubor a jeden INS soubor. INS soubor je předem zkompilovaný skript, potřebný pro instalaci. V případě, že při instalaci není specifikováno jinak, je automaticky použit INS soubor ze stejné složky, kde se nachází EXE soubor. Je samozřejmě možné použít i INS soubor, který se nachází v jiné složce:

```
InstallShieldSetup.exe -fC:\Test\mujINS.ins
```

Zdrojový kód 8: InstallShield parametr pro použití INS souboru

Druhou variantou instalace je použití tzv. response file, který má příponu ISS. Ten je běžně používán při instalaci bez interakce uživatele. Princip vytvoření souboru je, že se nejdříve projde celá instalace a přidají se specifikované modifikace (změna instalační složky, přidání licenčního serveru atd.) a po instalaci je námi zadaná konfigurace instalace uložena do nově vytvořeného ISS souboru.³²

Jak již bylo zmíněno, prvním krokem je vytvoření tohoto souboru. K tomu slouží přepínač „-r“, který defaultně vytvoří ISS soubor do %WINDIR% složky:

```
InstallShieldSetup.exe -r
```

Zdrojový kód 9: InstallShield parametr pro vytvoření ISS souboru

Druhým krokem je použití tohoto ISS souboru během instalace pomocí přepínače „-f1<CestaKSouboru>“. To zajistí využití naší konfigurace do instalace. Posledním krokem je přidání přepínače „/S“ nebo „-s“, který instalaci provede bez uživatelského rozhraní:

```
InstallShieldSetup.exe -s -f1<C:\Test\mujISS.iss>
```

Zdrojový kód 10: InstallShield parametr pro použití ISS souboru

InstallShield obsahuje více přepínačů, např. „-m“, který vytvoří MIF soubor se stejnojmennou příponou. Pro vložení sériového čísla do MIF souboru slouží „-m1[SerioveCislo]“ a pro změnu jazyka „-m2[ZkratkaJazyka]“. Angličtina je nastavená jako výchozí jazyková mutace. InstallShield ale nabízí mnohem více možností. Užitím přepínače „-SMS“, který je jako jediný ze všech ostatních přepínačů case-sensitive, zablokujeme připojení k internetu a také zabráníme tomu, že nastane zastavení procesu

³¹ *InstallShield Download Your Free Trial Today.* [online]. Dostupné z: <https://www.reverera.com/install/products/installshield.html/> [cit. 2021-04-24].

³² *Article: InstallShield Setup Silent Installation Switches.* [online]. Dostupné z: <http://www.itninja.com/blog/view/installshield-setup-silent-installation-switches/> [cit. 2021-04-21].

instalace dříve, než dojde k dokončení samotné instalace. Parametr „-z“ vypne ověření, jestli zařízení obsahuje alespoň 256 MB RAM paměti. Poslední parametr je „-uninst“, který slouží k odinstalaci softwaru.

3.1.5 App-V

Technologie Microsoft Application Virtualization je jako jedna z hlavních platform pro virtualizaci aplikací. Je to hlavně proto, že stále více organizací uvažují o virtualizaci aplikací jako o platné implementaci a také proto, že App-V představuje pravděpodobně nejatraktivnější řešení.³³ Původně bylo App-V vyvinuto společností Softricity, která měla sídlo v Bostonu, ovšem v roce 2006 došlo k odkoupení společností Microsoft. Tímto krokem vstoupil Microsoft na trh virtualizace a posléze k nim přidal další technologie Hyper-V, System Center Virtual Machine Manager a UE-V, které umožňuje uživatelům přístup ke svým desktopovým aplikacím prakticky odkudkoliv, na základě výběru zařízení.³⁴

Pomocí technologie App-V dochází k nasazení, nebo spíše streamu, v reálném čase na jakéhokoliv klienta, který má nainstalovaného App-V klienta, ze strany virtualizačního aplikačního serveru, např. SCCM. Rozdílem oproti klasické instalace je to, že instalační soubory neovlivňují operační systém, jelikož jsou uloženy buďto do lokální cache paměti, nebo do cache paměti uživatele. Jedná se o tzv. zásobník App-V, který sandboxuje prostředí, a proto nedochází k ovlivnění operačního systému včetně registrů. Je to jakási bublina žijící svůj život a tato bublina je pro každou App-V aplikaci samostatná. Pomocí tohoto oddělení lze na jednom stroji mít nainstalovaných více stejných aplikací i s jinou verzí. Výchozím úložištěm App-V balíčků je:

„C:\ProgramData\Microsoft\AppV\Client\Integration\<uuid>“

Obdobou App-V je VMware ThinApp nebo Citrix XenApp. Velice důležitou částí App-V je správné nastavení jak klientského stroje, tak i App-V sekvenceru. Nastavení klientského stroje, kde budeme výsledný balík streamovat, není příliš náročné a postačí přes Group Policy povolit App-V Client, nebo přes PowerShell spustit příkaz „Enable-Appv“.³⁵ U sekvenceru je nastavení komplikovanější, a to proto, že je nutné mít správně nastavené všechny registry a soubory. Prvně zkontrolujeme operační systém, na kterém budeme sekvencovat. Od verze 4.6 jsou podporované operační systémy Windows XP a vyšší a Windows Server 2003 a vyšší. Ačkoliv Microsoft na žádných svých stránkách nepíše o tipech a tricích k vyhnutí se problémům se sekvencováním, technici, kteří s touto technologií pracují každý den, sepsali několik kroků, které je vhodné dodržet. Prvním je vypnutí všech aktualizací systému, dále vypnutí Windows Defenderu, Bitlockeru i firewallu a poté alespoň třikrát restartovat operační systém.³⁶

³³ ALVAREZ, A. *Microsoft Application Virtualization Advanced Guide*. 1. vyd. Birmingham: Packt Publishing Ltd, 2012. ISBN 978-1-84968-448-4. s. 7.

³⁴ BOTT, E. *Introducing Windows 10 for IT Professionals, Preview Edition*. 1. vyd. Redmond: Microsoft Press, 2015. ISBN 978-0-7356-9696-9. s. 86.

³⁵ *Enable the App-V in-box client (Windows 10) - Windows Application Management*. [online]. Dostupné z: <https://docs.microsoft.com/en-us/windows/application-management/app-v/appv-enable-the-app-v-desktop-client/> [cit. 2021-04-24].

³⁶ TMURGENT TEAM *Packaging for App-V & MSIX: Packaging Concepts*, 1. vyd., neznámé: TMurgent Technologies. ISBN neznámé. s. 81-84.

3.2 Skriptovací jazyky

Když se řekne programovací jazyk, většina lidí si představí objektově orientované jazyky typu Java, C#, C++ atp. Málo koho napadne, že skriptovací jazyky patří také mezi programovací. Výhodou takového skriptu je jeho jednoduchost a pohodlnost při vývoji programu. Základními vlastnostmi jsou dynamická typová kontrola, chybové hlášení, které neukončí provádění skriptu, a v neposlední řadě není potřeba hlídat uvolňování paměti a nemusí se deklarovat proměnné.³⁷

Značnou výhodou skriptů je to, že nepotřebují speciální software pro kompilaci, jelikož se upravuje přímo kód, který může být otevřen v jakémkoliv textovém editoru. Nevýhodou může být nižší rychlost vykonání v porovnání s optimalizovanými přeloženými programy. Mezi nejznámější skriptovací jazyky patří PHP, Python, JavaScript, VBS, PowerShell, ovšem i dotazovací jazyk PL/SQL

3.2.1 PowerShell

Historie PS sahá až do roku 2006, kdy vyšla první verze pod záštitou Microsoftu. Od té doby je PowerShell stále na vzestupu a v dnešní době je hojně využit v mnoha oblastech. Tím, že je postaven na .NET Frameworku³⁸, je kompatibilní s operačním systémem Windows 7 a vyšší, Windows Server 2008 R2 a vyšší, Linux i macOS 10.12 a vyšší. Pro starší OS existuje rozšíření PowerShell 2.0, které je s těmito již nepodporovanými systémy také kompatibilní.

Windows PowerShell je založen na tzv. cmdlets, což jsou specializované třídy .NET, kde každá třída implementuje určitou operaci. Skriptem PS je kompozice cmdletů s podporou logických podmínek a díky těmto vlastnostem o něm hovoříme jako o nástupci příkazového řádku. Název přípony vznikl v době jeho prvního vydání, kdy PS nesl název PowerShell 1.0, a tak příponu pojmenovali „.ps1“.

Pro spuštění skriptu napsaného v PowerShellu je většinou nutné vypnout tzv. „execution policy“³⁹, které jsou často blokovány pomocí politik a mají zamezit spouštěním škodlivých skriptů. Obecně lze pomocí něj jakkoliv modifikovat soubory, složky, registry, ovladače, stahovat a instalovat aplikace a vytvářet vlastní okna a dokonce jednoduché aplikace.

3.2.2 Visual Basic Script

VBS je skriptovací jazyk používaný pro běžné skriptování ve WSH a je založen na jazyce Visual Basic. První zmínka je v roce 1996, aktuálně je součástí každého operačního systému Microsoft Windows od historické verze Windows 98 a je vyvíjen společností Microsoft.

VBS byl primárně určen pro vkládání kódu do www stránek, ovšem má i jiné využití. Mezi výhody patří hlavně kompatibilita s většinou současných Microsoft Windows s WSH, je objektově orientovaný a díky WMI jsou schopnosti VBS téměř neomezené. Proto je oblíbený mezi hackery, kteří přes VBA API dokážou přistoupit téměř k čemukoliv na zařízení napadeného uživatele. V této

³⁷ BOSANAC, D. *Scripting in Java™: Languages, Frameworks, and Patterns*. 1. vyd. Boston: Pearson Education, 2008. ISBN 978-0-321-32193-0. s. 23.

³⁸ DESHEV, H. *Pro Windows PowerShell*. 1. vyd. New York: Springer-Verlag, 2008. ISBN 978-1-4302-0546-3. s. 1.

³⁹ DESHEV, H. *Pro Windows PowerShell*. 1. vyd. New York: Springer-Verlag, 2008. ISBN 978-1-4302-0546-3. s. 140.

diplomové práci byl VBS použit k zabalení instalačních souborů, jelikož dokáže modifikovat registry, kopírovat a mazat soubory, přidávat lokální administrátory, zobrazovat základní vyskakovací okna, spouštět a zavírat procesy, a hlavně dokáže předávat informace o stavu instalace do event logu a následně SCCM.⁴⁰

3.2.3 AutoIT

AutoIT patří mezi volně dostupné skriptovací jazyky pro automatizaci instalací na OS Microsoft Windows. První verze byla vydána v roce 1999 vývojářem Jonathanem Bennetem a jeho týmem. Skvělým nástrojem pro psaní a kompilování AutoIT skriptů je SciTE4AutoIt3.

Pomocí AutoIT můžeme nasimulovat stisknutí kláves, pohyby a kliknutí myši a vytvoření jiných oken či podoken. Zdrojové kódy jsou psány v souboru s příponou „.au3“ a po kompilaci má soubor příponu EXE. Nejčastější použití tohoto jazyka jsou instalátory, u kterých není definována instalace bez uživatelského rozhraní. Když tato možnost chybí, je potřeba nastavit kliknutí na určité tlačítko či přepnout přepínač do druhé polohy.⁴¹

⁴⁰ *What is VBScript? Introduction & Examples.* [online]. Dostupné z: <https://www.guru99.com/introduction-to-vbscript.html> [cit. 2021-04-24].

⁴¹ *AutoIt Scripting Language.* [online]. Dostupné z: <https://www.autoitscript.com/site/autoit/> [cit. 2021-04-24].

4 Realizace skutečného prostředí

Následující kapitola obsahuje celkový postup k vytvoření virtuálního prostředí včetně instalace a konfigurace SCCM až po nasazení instalačního balíčku. Instalační balíček je detailně rozebrán a nasazení je provedeno více způsoby včetně vyzkoušení více softwarů.

4.1 Virtuální prostředí

Prostředí vytvořené v této diplomové práci obsahuje AD server, SCCM server a dva testovací klienty – jeden s OS Windows 10 a druhý s OS Linux. K vytvoření i tak malého prostředí je potřeba mít dostatečně dost hardwarových prostředků. Tyto prostředky byly poskytnuty firmou TietoEVRY Czech s.r.o, kde byl použit jeden z jejich hostů. Host nabízí 16 GB RAM, 8-jádrový procesor s frekvencí 3,40 GHz, 1 TB volného místa na disku a předinstalovaný Microsoft Hyper-V Server 2016. K práci na zmíněném hostu je zapotřebí být v interní síti firmy a také mít patřičná práva k přístupu a možnosti jakékoliv konfigurace.

Celé prostředí bude virtualizované a spravované pomocí System Center Virtual Machine Manager. To nabízí širší možnosti než klasický Hyper-V Manager. První krokem je vytvoření virtuálního stroje pfSense, který slouží jako softwarový router a bude použit k oddělení interní sítě firmy „eu“ a sítě, která bude použita pro prostředí diplomové práce. Tato síť bude pojmenovaná „vsb“. PfSense je open-source freeware a virtuálnímu stroji, na kterém pfSense poběží, stačí dle doporučení vendora 1 GB RAM, jedno jádro procesoru a ideálně 10 GB volného místa na disku. Během vytváření bylo zjištěno, že stroji během instalace musí být přiřazena generace 1, jinak k instalaci nedojde. Instalace bude provedena bez jakýkoliv konfigurací a výsledkem by mělo být následující nastavení (WAN na rozhraní hn0 a LAN na rozhraní hn1).

```
Starting syslog...done.
Starting CRON...done.
pfSense 2.4.5-RELEASE amd64 Tue Mar 24 15:25:50 EDT 2020
Bootup complete

FreeBSD/amd64 (pfSense.localdomain) (ttyu0)

Hyper-U Virtual Machine - Netgate Device ID: b8a84071c2c58843d754

*** Welcome to pfSense 2.4.5-RELEASE (amd64) on pfSense ***

WAN (wan)      -> hn0      -> v4/DHCP4: 10.14.131.52/23
LAN (lan)      -> hn1      -> v4: 192.168.1.1/24

0) Logout (SSH only)          9) pf Top
1) Assign Interfaces          10) Filter Logs
2) Set interface(s) IP address 11) Restart webConfigurator
3) Reset webConfigurator password 12) PHP shell + pfSense tools
4) Reset to factory defaults  13) Update from console
5) Reboot system              14) Enable Secure Shell (sshd)
6) Halt system                 15) Restore recent configuration
7) Ping host                   16) Restart PHP-FPM
8) Shell

Enter an option: █
```

Obrázek 10: Nastavení virtuálního stroje pfSense

4.1.1 DC server

Dalším nutným serverem je doménový řadič. Není nijak hardwarově náročný, takže mu bylo přiděleno pouze jedno jádro procesoru a dynamická velikost RAM paměti od 3072 MB do 4096 MB. Na server byl nejprve nainstalován Windows Server 2019 Standard a poté byla doinstalována funkcionality Active Directory Domain Services. Po dokončení instalace byl server povýšen na doménový řadič, kterému byl vytvořen nový strom „vsb.local“. Po restartu byl v Active Directory Users and Computers

vytvořen nový uživatel pro budoucí testování. Dále byla vytvořena nová zóna „1.168.192.in-addr.arpa“ pro Reverse Lookup v DNS nastavení a Pointer na „WIN-F86ME56NRCT.vsb.local.“ s IP adresou „192.168.1.102“, kde „WIN-F86ME56NRCT“ je název vytvořeného doménového řadiče.

Pro správnou funkčnost s SCCM serverem je potřeba nastavit další funkcionality. První je vytvoření System Management Container v ADSI Edit nabídce. Další úpravy týkající se doménového řadiče jsou popsány v následující kapitole, jelikož mohou být provedeny až po vytvoření SCCM serveru.

4.1.2 SCCM server

Posledním serverem je SCCM. Ten je ze všech nejvytíženější, jelikož na něm poběží samotné SCCM, SQL Server a bude obsahovat distribuční bod pro testovací prostředí. Kvůli těmto důvodům mu bylo přiděleny dvě procesorová jádra, dynamicky alokovaná velikost paměti od 4096 MB do 6144 MB a celkově 250 GB volného místa na disku. Před samotnou instalací Configuration Manageru se musí nainstalovat následující prerekvizity, komponenty a služby IIS:⁴²

- 1) .Net Framework 3.5 Features a všechny jeho podkomponenty
- 2) .Net Framework 4.7 Features a všechny jeho podkomponenty
- 3) BITS
- 4) Remote Differential Compression
- 5) Common HTTP Features, konkrétně Default Document, Static Content
- 6) Application Development, konkrétně .NET Extensibility 3.5, .NET Extensibility 4.7, ASP.NET 3.5, ISAPI extensions a ASP.NET 4.7
- 7) Security, konkrétně Windows Authentication
- 8) IIS 6 Management Compatibility, konkrétně IIS Management Console, IIS 6 Metabase Compatibility, WMI Compatibility a IIS Management Scripts and Tools
- 9) Windows ADK for Windows 10, konkrétně Deployment tools a User State Migration Tool (USMT)
- 10) Windows PE add-on for the ADK
- 11) SQL Server 2017, zejména Database Engine Services
- 12) WSUS roli, konkrétně WSUS Services a SQL Server Connectivity, u které je potřeba specifikovat složku, kde budou uloženy aktualizace a také název databázového serveru
- 13) SQL Server 2017 Reporting Services

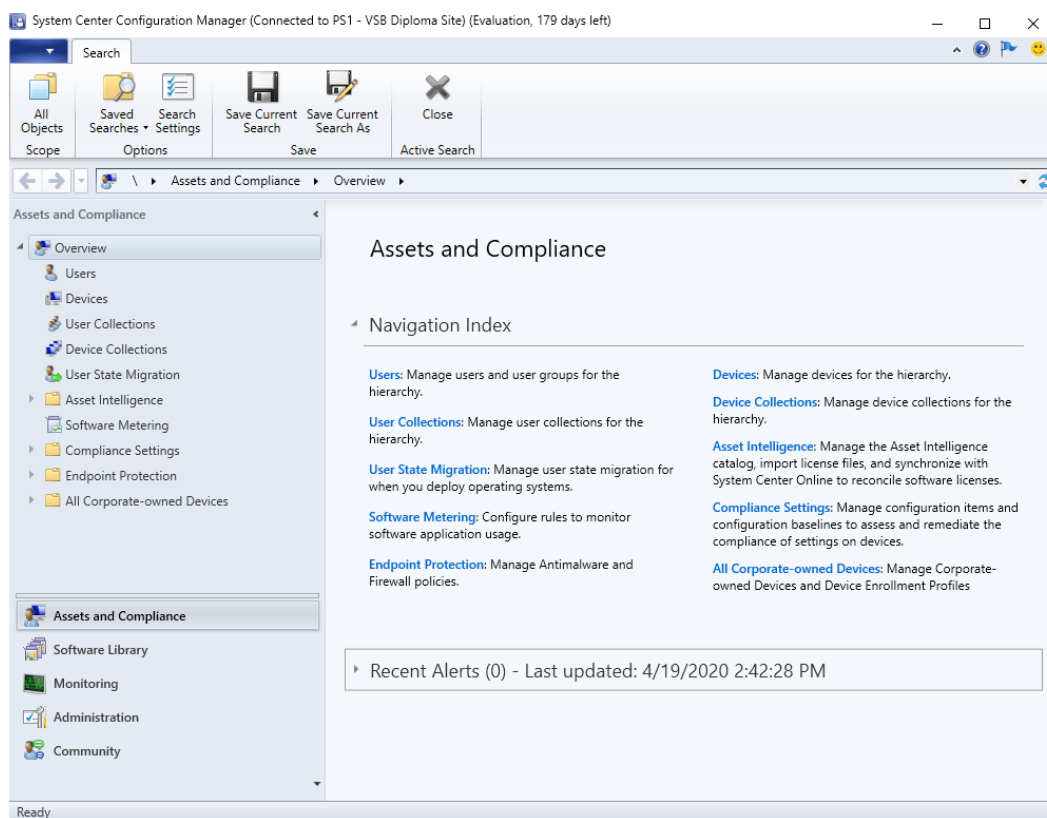
Nezbytnou součástí SCCM serveru jsou GPO, tedy politiky. Následně musí dojít k úpravě firewall pravidel na doménovém řadiči. Tyto úpravy se provedou pomocí Group Policy Managementu pro námi vytvořenou doménu. Do příchozích pravidel přidáme předdefinované pravidlo pro Windows Management Instrumentation a pro File and Printer Sharing. Do odchozích přidáme pouze File and Printer Sharing. Dále se musí povolit příchozí pravidlo pro port 1433, přes který SQL server naslouchá příchozí připojení, a 4022 pro SQL Server Service Broker. Po přidání všech výše zmíněných změn se musí provést „gpupdate /force“ pomocí příkazového řádku nebo PowerShellu a to na všech zařízeních,

⁴² MEYLER, K. – HOLT, B. – OH, M. – SANDYS, J. – RAMSEY, G. *System Center 2012 Configuration Manager: UNLEASHED*. 1. vyd. USA: Pearson Education, Inc., 2013. ISBN 978-0-672-33437-5. s. 261-265.

kteří jsou připojeni do domény.⁴³ V rámci nastavení sítě je ještě nutné změnit DNS server SCCM na IP adresu doménového řadiče. Po těchto krocích můžeme připojit SCCM do domény.

Po připojení a následném restartu se musí přes příkazový řádek provést rozšíření AD schématu a to pomocí „extadsch.exe“ souboru, který se nachází v instalační složce Configuration Manageru, konkrétně ve složce „SMSSETUP\BIN\x64“, a který musí být spuštěný pod uživatelem, který je součástí Schema Admins skupiny na doménovém řadiči. Během konfigurace doménového řadiče došlo k vytvoření System Management Container v Active Directory Users and Computers (ADUC). Když už máme vytvořený SCCM server s názvem WIN-DJTP5O5IBF9, dalším krokem bude přidání delegování kontroly. To se provede opět v ADUC, kde během delegování se musí vybrat možnost „This folder, existing objects in this folder and creation of new objects in this folder“ a musí se zadat název SCCM serveru. Po přidání delegace musíme ověřit, zda námi přidáný uživatel má tzv. „full control“ práva, pokud ne, přidáme je.⁴⁴

Když došlo ke všem nastavením výše, můžeme přejít do první konfigurací přímo v Configuration Manager Console. Zástupce pro tuto aplikaci najdeme v nabídce Start. Po kliknutí na něj se otevře hlavní přehledové okno.



Obrázek 11: SCCM – Přehledové okno

⁴³ *Configuring Firewall Settings For Configuration Manager 2012 R2*. [online]. Dostupné z: <https://www.prajwaldesai.com/configuring-firewall-settings-for-configuration-manager-2012-r2/> [cit. 2021-04-21].

⁴⁴ *SCCM 1902 Install Guide Using Baseline Media*. [online]. Dostupné z: <https://www.prajwaldesai.com/sccm-1902-install-guide-using-baseline-media/> [cit. 2021-04-21].

V levé dolní části můžeme vidět kategorie, které byly blíže popsány v teoretické části této diplomové práce. Dalším nastavením bude nastavení Boundaries a distribučních bodů, bez kterého by nedošlo k distribuci aplikací, task sequences, aktualizací ani k instalaci SCCM klienta na koncové zařízení. V Administration ve složce Hierarchy Configuration a podsložce Boundaries musíme vytvořit novou boundary, kde typem bude Active Directory site a musí dojít k nalezení tzv. Default-First-Site-Name. Tu vybereme a přidáme volitelný popis.⁴⁵

Po vytvoření boundary je nutné také vytvořit Boundary group, u které zvolíme námi vytvořenou boundary. V záložce References musíme zaškrtnout „Use this boundary group for site assignment“ a v nabídce Assigned site vybrat naši primary site. V listu pro **Site system servers** vybereme název SCCM serveru, tedy „WIN-DJTP5O5IBF9.vsb.local“. Když máme vytvořené jak boundary, tak boundary group, zkontrolujeme, zda je správně nastaven distribuční bod primary site. Ten by měl být nakonfigurován ve výchozím nastavení během samotné instalace SCCM a neměly by být nutné žádné další změny. V případě, že bychom měli více distribučních bodů, můžeme vytvořit Distribution points group, tedy skupinu, do které přidáme všechny body, které jsou pro tuto skupinu podstatné. Příkladem by mohlo být vytvoření skupiny, která by obsahovala všechny distribuční body nacházející se na území České republiky. Druhou takovou skupinou by byla skupina se všemi body na území Slovenska atp.⁴⁶

Naším cílem je správa koncového zařízení, a to není možné, pokud cílové zařízené nebude mít nainstalovaného SCCM klienta. Ve složce Site Configuration a podsložce Sites zvolíme v roletě SCCM v Settings Client Installation Settings a vybereme Client Push Installation.

Zde zvolíme automatické instalování klienta, zaškrtneme všechny typy zařízení, tedy Servers, Workstations a Configuration Manager site system servers. V poslední záložce nastavíme přepínače, které mají být použity během instalace. Jelikož nemáme žádný jiný server, který by se nám staral o prostředí, většinou se jedná o terminal site, ponecháme instalační přepínače s výchozím nastavením.

Posledním krokem před funkčním SCCM bude změna politiky, kdy má docházet k obnovení politik klientských zařízení. Tuto možnost najdeme v záložce Administration, složce Client Settings, kde po pravém kliknutí myši na Default Client Settings zvolíme Properties, při čemž dojde k zobrazení všech politik SCCM. V záložce Client Policy zjistíme, že k instalaci SCCM klienta by docházelo jednou za hodinu. Tento interval snížíme na 15 minut. V záložce Software Updates snížíme interval instalací aktualizací na jeden den. V kartě Client Cache settings nastavíme maximální velikost cache paměti koncového zařízení na 5 GB a v Computer Agent změníme název organizace na „VSB Diploma test environment“. Samozřejmě, že zde můžeme nastavit mnoho dalších politik, ale pro základní účely nám stačí pouze tyto.⁴⁷

⁴⁵ *CONFIGURE SCCM 2012 BOUNDARIES.* [online]. Dostupné z: <https://www.systemcenterdudes.com/configure-sccm-2012-boundaries/> [cit. 2021-04-21].

⁴⁶ *SCCM 2012 DISTRIBUTION POINT INSTALLATION.* [online]. Dostupné z: <https://www.systemcenterdudes.com/sccm-2012-distribution-point-installation/> [cit. 2021-04-21].

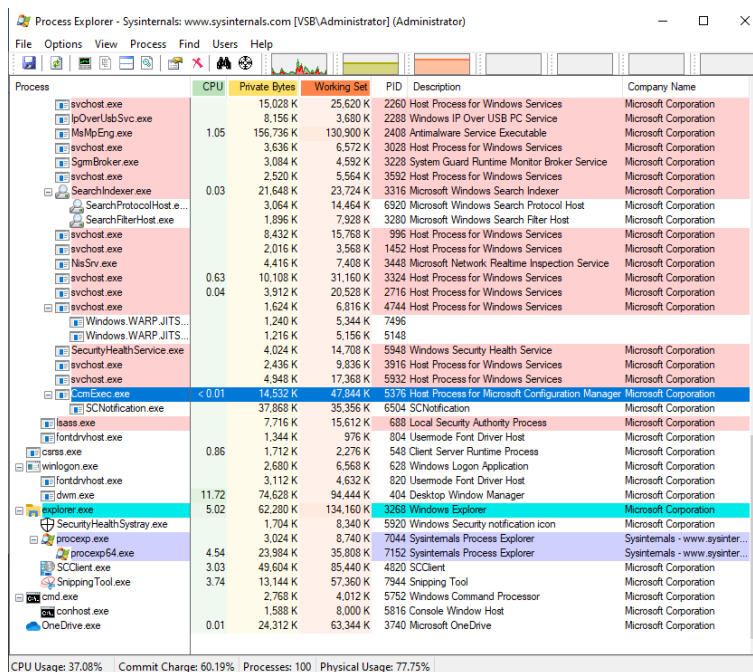
⁴⁷ *Using SCCM 2012 RC in a LAB – Part 4. Configuring Client Settings and adding roles.* [online]. Dostupné z: <https://www.niallbrady.com/2011/10/29/using-sccm-2012-rc-in-a-lab-part-4-configuring-client-settings-and-adding-roles/> [cit. 2021-04-21].

4.1.3 Testovací klienti

Po dokončení doménového řadiče a SCCM může dojít k připojení koncových zařízení do domény a následné správy pomocí našeho serveru. Pro prvního testovacího klienta s Windows 10 byl vytvořen virtuální stroj s dvěma jádry procesoru, dynamickou pamětí od 1,5 GB do 2 GB a volným místem na disku 40 GB. Po instalaci operačního systému byl stroj přiřazen do domény „vsb“ pomocí účtu s administracími právy SCCM serveru. Když došlo k připojení do domény, musel být proveden restart. Po restartu by SCCM server měl zpozorovat nově připojené zařízení a nejpozději do 15 minut by mělo dojít k instalaci SCCM klienta, a to díky naší zvolené politice. Samotná instalace trvá okolo dalších 10 minut, kdy se často stává, že instalace zhavaruje. Pro investigaci kořene problému slouží log soubor „ccmsetup.log“, který se nachází ve složce „%WINDIR%\ccmsetup\Log“. Ten obsahuje všechny kroky instalace, včetně veškerých upozornění, a pomocí CMTrace dochází ke grafickému zviditelnění chyb a dalších hlášení instalace.

V některých případech k automatické instalaci klienta nedojde a musí dojít k manuální instalaci. Prvním krokem je zkopírování všech instalačních souborů ze složky úložiště primary site na lokální disk afektovaného počítače. Tyto soubory se nachází v našem případě ve složce „\\WIN-DJTP505IBF9\SMS_PS1\Client“. Když jsou soubory zkopírovány, je nutné spustit „ccmsetup.exe“ minimálně pod právy lokálního administrátora. Pokud do této doby žádný vytvořen nebyl, je možné použít administrátorský účet SCCM. Před spuštěním instalace musíme ověřit správné instalační přepínače. Ty najdeme buďto v **Client Push Installation** nabídce, nebo si nastudujeme pomocí přepínače „/?“ další možnosti použití.

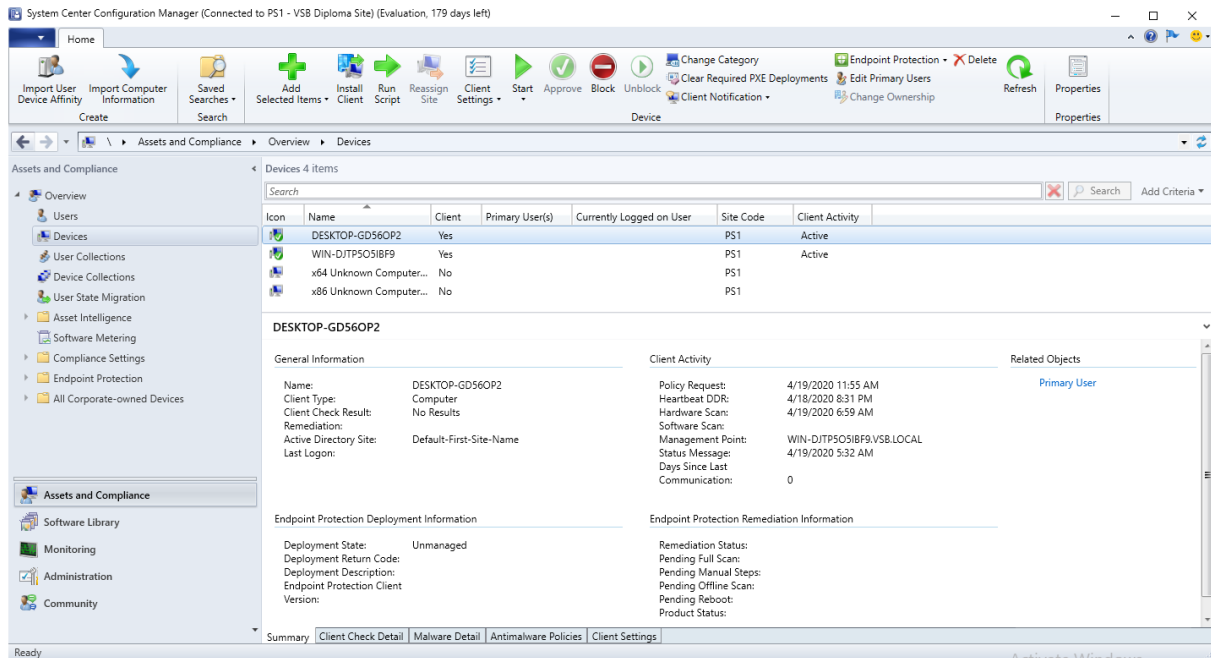
Pokud instalace proběhla úspěšně, můžeme zpozorovat, že došlo k nainstalování chybějících prerekvizit, jako jsou C++ atp. Důležitým faktorem a ověřením správné instalace je to, že v běžících procesech bude aktivní „CcmExec.exe“ a hlavně v nabídce Start se objeví zástupce na Software Center. Výborným nástrojem pro přehlednost běžících procesů, aktuálně přihlášených uživatelů atp. je Process Explorer, který je volně stažitelný.



Process	CPU	Private Bytes	Working Set	PID	Description	Company Name
svchost.exe	15,028 K	25,620 K	2,260 K	2260	Host Process for Windows Services	Microsoft Corporation
lpOverLabSvc.exe	8,156 K	3,680 K	2,288 K	2288	Windows IP Over USB PC Service	Microsoft Corporation
MsMpEng.exe	156,736 K	130,900 K	2,408 K	2408	Antimalware Service Executable	Microsoft Corporation
svchost.exe	3,636 K	6,572 K	3,028 K	3028	Host Process for Windows Services	Microsoft Corporation
SgmnBroker.exe	3,084 K	4,592 K	3,228 K	3228	System Guard Runtime Monitor Broker Service	Microsoft Corporation
svchost.exe	2,520 K	5,584 K	3,592 K	3592	Host Process for Windows Services	Microsoft Corporation
SearchIndexer.exe	21,648 K	23,724 K	3,316 K	3316	Microsoft Windows Search Indexer	Microsoft Corporation
SearchProtocolHost.exe	3,064 K	14,464 K	6,920 K	6920	Microsoft Windows Search Protocol Host	Microsoft Corporation
SearchFilterHost.exe	1,896 K	7,928 K	3,280 K	3280	Microsoft Windows Search Filter Host	Microsoft Corporation
svchost.exe	8,432 K	15,768 K	996 K	996	Host Process for Windows Services	Microsoft Corporation
svchost.exe	2,016 K	3,568 K	1,452 K	1452	Host Process for Windows Services	Microsoft Corporation
NiaSvc.exe	4,416 K	7,408 K	3,448 K	3448	Microsoft Network Realtime Inspection Service	Microsoft Corporation
svchost.exe	0,63	10,108 K	31,160 K	3324	Host Process for Windows Services	Microsoft Corporation
svchost.exe	0,04	3,912 K	20,528 K	2716	Host Process for Windows Services	Microsoft Corporation
svchost.exe		1,624 K	6,816 K	4744	Host Process for Windows Services	Microsoft Corporation
Windows WARP JITs...		1,240 K	5,344 K	7496		
Windows WARP JITs...		1,216 K	5,156 K	5148		
SecurityHealthService.exe		4,024 K	14,708 K	5948	Windows Security Health Service	Microsoft Corporation
svchost.exe		2,436 K	9,836 K	3916	Host Process for Windows Services	Microsoft Corporation
svchost.exe		4,948 K	17,368 K	5932	Host Process for Windows Services	Microsoft Corporation
CcmExec.exe	< 0,01	14,532 K	47,844 K	5376	Host Process for Microsoft Configuration Manager	Microsoft Corporation
SCNotification.exe		37,868 K	35,356 K	6504	SCNotification	Microsoft Corporation
lsass.exe		7,716 K	15,612 K	688	Local Security Authority Process	Microsoft Corporation
fontdrvhost.exe		1,344 K	976 K	804	Usemode Font Driver Host	Microsoft Corporation
carsrv.exe	0,86	1,712 K	2,276 K	548	Client Server Runtime Process	Microsoft Corporation
winlogon.exe		2,680 K	6,568 K	628	Windows Logon Application	Microsoft Corporation
fontdrvhost.exe		3,112 K	4,632 K	320	Usemode Font Driver Host	Microsoft Corporation
dmv.exe	11,72	74,628 K	94,444 K	404	Desktop Window Manager	Microsoft Corporation
explorer.exe	5,02	62,280 K	134,160 K	3252	Windows Explorer	Microsoft Corporation
SecurityHealthSystray.exe		1,704 K	8,340 K	5920	Windows Security notification icon	Microsoft Corporation
procexp.exe		3,024 K	8,740 K	7044	Sysinternals Process Explorer	Sysinternals - www.sysinter...
procexp64.exe	4,54	23,984 K	35,808 K	7152	Sysinternals Process Explorer	Sysinternals - www.sysinter...
SCClient.exe	3,03	49,604 K	85,440 K	4820	SCClient	Microsoft Corporation
SnippingTool.exe	3,74	13,144 K	57,360 K	7944	Snipping Tool	Microsoft Corporation
cmd.exe		2,768 K	4,012 K	5752	Windows Command Processor	Microsoft Corporation
conhost.exe		1,588 K	8,000 K	5816	Console Window Host	Microsoft Corporation
OneDrive.exe	0,01	24,312 K	63,344 K	3740	Microsoft OneDrive	Microsoft Corporation

Obrázek 12: Process Explorer

Po kliknutí na zástupce Software Center se načte nabídka všech dostupných instalačních balíčků a task sekvencí. Aktuálně není na tento počítač poslána žádná sekvence ani balíček. Ověření na straně SCCM, zda byl klient správně nainstalován, je možné ze záložky Assets and Compliance, složky Devices. Tady musíme nalézt název hledaného počítače, což je někdy nadlidský úkon, když prostředí obsahuje tisíce koncových zařízení. Proto musíme znát přesný název.



Obrázek 13: SCCM - Devices

Ve sloupci Client je zmíněno, jestli je klient nainstalován či nikoliv. Pokud bychom chtěli klienta nainstalovat vzdáleně ručně, je možné provést instalaci právě odsud, a to po kliknutí pravým tlačítkem myši na název počítače, kde klienta chceme nainstalovat nebo přeinstalovat, a výběrem Install Client. Během průběhu průvodce instalace nesmíme zapomenout zaškrtnout „Install the client software from a specified site“, pomocí kterého určíme, že instalační soubory se mají stáhnout z námi zvolené site – v našem případě máme jen jednu, a to primary site.

4.2 Výběr skriptovacího jazyka, nasazení a test balíčku

V ukázkách výše byly zmíněné dvě volně dostupné aplikace, a právě pro jednu z nich bude vytvořen instalační balíček. Jedná se o aplikaci s názvem Mozilla Firefox. Instalační média byla stažena z oficiálních stránek vendora „<https://www.mozilla.org/cs/firefox/all/#product-desktop-release>“. Při zjišťování přepínačů instalace ovšem došlo k tomu, že instalátor sám o sobě nenesl žádnou nápovědu právě k přepínačům. Při zjištění byly použity „-?“, „-help“, „/?“ a „/help“ a žádný z nich nezobrazil nápovědu. Tenhle scénář bývá zřídka, ale musí se počítat i s ním. V těchto případech většinou výrobce softwarů poskytuje nápovědu na svých stránkách, nebo si ostatní IT odborníci radí na fórech či specializovaných stránkách, kde si předávají zjištěné přepínače a jejich použití. Pro Mozilla Firefox byla nalezena stránka „https://wiki.mozilla.org/Installer:Command_Line_Arguments“, která právě obsahuje všechny detaily k instalaci.

Instalační balíček bude obsahovat standardy, které se běžně používají při poskytování služby kompletní správy prostředí. Mezi tyto standardy patří tichá instalace bez interakce uživatele, vypnutí

automatických aktualizací aplikace, smazání zástupce softwaru na ploše a také zástupců ve Start Menu, kteří by mohli jakkoliv modifikovat nainstalovanou aplikaci. V praxi to bývá zejména zástupce na odinstalaci, či její oprava. Z výše zmíněné stránky s instalačními přepínači byl zjištěn switch „-ms“, který provede tichou instalaci. Další parametry k aplikování námi definovaných standardů nebyly nalezeny, a proto musí dojít k tzn. zabalení instalátoru, které bude provedeno pomocí Visual Basic Scriptu. Jakákoliv instalace, spuštění skriptu, který zasahuje do systémových či společných složek pro všechny uživatele, vyžaduje minimálně práva lokálního administrátora. V našem testovacím prostředí bude využit administrátorský účet SCCM serveru.

Jelikož instalační soubor neumí sám zavřít proces „Firefox.exe“, byla vytvořena funkce CreateKillProcess(), která se stará právě o zavírání procesů. Ukázka kódu, který pracuje s pomocí VBS s WMI.

```
Function CreateKillProcess(action,process_name)

strComputer = "."
Set objWMIService = GetObject("winmgmts:{impersonationLevel=impersonate}!\\" & strComputer &
"\root\cimv2")

process_name = Replace(process_name, " ", "", 1)
arrProcesses = Split (process_name, ",")

Select Case action
    Case "create"
        Const SW_NORMAL = 1
        Set objStartup = objWMIService.Get("Win32_ProcessStartup")
        Set objConfig = objStartup.SpawnInstance_
        objConfig.ShowWindow = SW_NORMAL
        Set objProcess = objWMIService.Get("Win32_Process")

        For Each strProcess In arrProcesses
            intReturn = objProcess.Create (strProcess, Null, objConfig, intProcessID)
            Call EventLogCreateKillProcess("create",intReturn,strProcess)
        Next

    Case "kill"
        For Each strProcess In arrProcesses
            Set colProcess = objWMIService.ExecQuery("Select * from Win32_Process
            Where Name = '" & strProcess & "'")
            For Each objProcess In colProcess
                ExitCode = objProcess.Terminate()
                Call EventLogCreateKillProcess("kill",ExitCode,objProcess.Name)
            Next
        Next
    Next
End Select
End Function
```

Zdrojový kód 11: Ukázka funkce CreateKillProcess()

Tento krok je prvním krokem balíčku pro případ, že je nainstalovaná jakákoliv předchozí verze, nebo právě ta aktuální, a uživatel má tuto aplikaci zrovna v okamžiku instalace otevřenou. Dalším krokem je odinstalace předchozích verzí a té aktuální. Pozor, v tomto případě se nejedná o tzv. upgrade. Po odinstalaci zůstane několik souborů ve složce, která obsahovala nainstalovanou aplikaci, a proto je nutné ji také smazat. Konkrétně se jedná o „%ProgramFiles32%\Mozilla Firefox\“. Když máme zařízení takto čisté, může dojít ke spuštění instalace s námi nalezeným přepínačem. Po dokončení se v Ovládacích panelech v Programy a funkce vytvoří dva záznamy. Prvním je Mozilla Firefox a druhým Mozilla Maintenance Service. Pro splnění standardu vypnutí automatických aktualizací musí být druhý záznam zpátky odinstalován, protože právě tato služba je poskytuje. Bohužel to není jediné, co je potřeba udělat k jejich vypnutí. Po instalaci se v instalační složce v „%ProgramFiles32%\Mozilla Firefox\“ vytvoří soubor „Mozilla.cfg“, který obsahuje mnohem více nastavení samotné aplikace a v kterém je také potřeba upravit vypnutí aktualizací. Dalšími obdobnými soubory jsou:

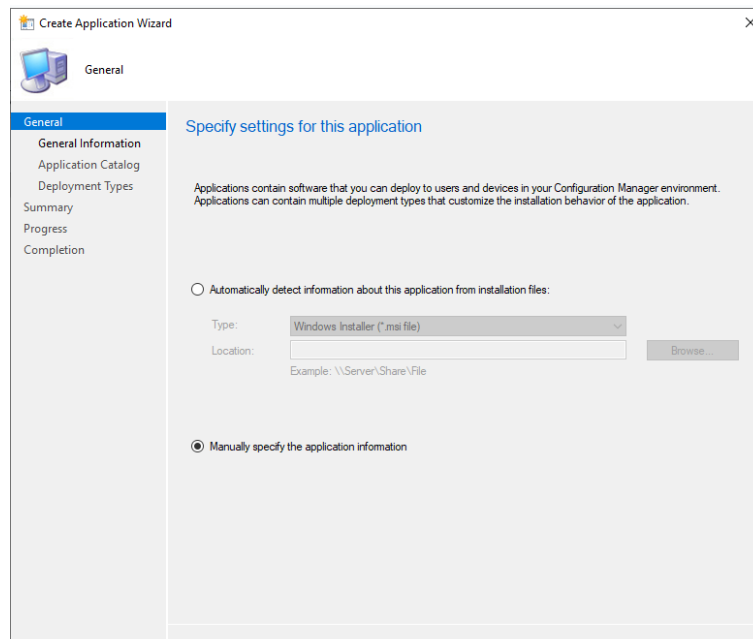
- 1) „lock-settings.js“ ve složce „%ProgramFiles32%\Mozilla Firefox\defaults\pref\“
- 2) „policies.json“ ve složce „%ProgramFiles32%\Mozilla Firefox\distribution\“

Upravené soubory si zkopírujeme do složky instalačního balíčku, protože budou použity pro další budoucí instalace na jiných zařízeních. Následuje přidání registru do větve „HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Mozilla\Firefox\Crash Reporter“ s proměnnou „SubmitCrashReport“ a její hodnotou „dword:00000000“. Toto nastavení zamezí posílání zpráv o selhání aplikace vendorovi. Pro lepší investigaci problémů, který by mohly nastat s aplikací v budoucnu, přiřadíme plnou kontrolu všem uživatelům na složku „%ProgramFiles32%\Mozilla Firefox\uninstall“, která obsahuje logy odinstalace. Posledním krokem je přidání detekčního pravidla pro SCCM, kde to bude vytvoření proměnné Comments v klíči registru této aplikace. Pro všechny tyto akce byly vytvořené vlastní funkce ve VBS a výsledný skript byl pojmenován „Install.vbs“. Druhým skriptem je skript „Uninstall.vbs“ pro odinstalaci, který obsahuje pouze odinstalaci aplikace a smazání instalační složky v %ProgramFiles32%. Oba skripty jsou součástí elektronické přílohy. Předposledním souborem je soubor ikony Mozilla Firefox, který bude použit při nasazení na SCCM, a posledním textový soubor, který obsahuje informace ohledně instalačního balíčku, včetně klíče v registrech, architektury aplikace atp. Výsledná struktura instalačního balíčku je:

- 1) Install.vbs – skript k instalaci
- 2) Uninstall.vbs – skript k odinstalaci
- 3) Mozilla_Firefox_75.0_EN_1.0.txt – textový soubor obsahující informace k balíčku
- 4) Složka _Media, která obsahuje následující soubory
 - a. CrashReporter_x64.reg
 - b. CrashReporter_x86.reg
 - c. Firefox Setup 75.0.exe – instalační soubor Mozilla Firefox
 - d. firefox_0_2.ico
 - e. lock-settings.js
 - f. Mozilla.cfg
 - g. policies.json

Po zhotovení instalačního balíčku může dojít k jeho nasazení na SCCM. Prvním krokem je zkopírování všech souborů do složky, která bude sloužit jako zdroj balíčků. Takovou složkou může být úložiště nebo v případě ukázky síťová složka SCCM „SMS_PS1\Packages\“. Scénářem nasazení byl

vybrán scénář, kde balíček bude nasazen jako aplikační model. V Configuration Manager Console ve složce Software Library v Application Management pravým kliknutím na Applications vybereme Create Application.



Obrázek 14: Scénář nasazení aplikace

V tomto okně vybereme „Manually specify the application information“, jelikož všechny možnosti budeme definovat sami a postupně. Pokračujeme kliknutím na Next. Do řádku **Name** zadáme unikátní jméno balíčku, další pole jsou volitelná, ovšem téměř každé z nich dokáže ovlivnit výsledný záznam v Software Center u koncového uživatele, a proto vyplníme **Publisher**, tedy firmu, která buďto instalační balíček nebo samotný software vytvořila, dále **Software version**, kde patří verze aplikace. Můžeme také zaškrtnout **Date Published**, které zobrazí, kdy byl balíček přidán na SCCM.

Další okno se zaměřuje hlavně na Software Center. Řádek „Localized application name“ změní hlavní nadpis balíčku a je zde možnost přidat ikonu aplikace. Pokud jsme si předem připravili nějakou uživatelskou kategorii, zde si ji vybereme, nebo vytvoříme novou. Kategorie se hodí např. k odlišení aplikací, které jsou placené a které freeware. Další viditelnou položkou je „Localized description“, do které se většinou píše popis k aplikaci, jako např. u Mozilla Firefox napíšeme, že se jedná o internetový prohlížeč, který je zdarma a který poskytuje přístup k internetu.

Pokračováním na další okno se dostaneme na vytvoření deployment type této aplikace, kde zvolíme Add. V **general** okně zvolíme typ „Script installer“ a poté je nám automaticky zvoleno „Manually specify the deployment type information“.

Pokračujeme na další okno, kde zadáme název deployment type, který musí být opět jedinečný, jelikož aplikace může obsahovat více deployment typů. Přidáme „Administrator comments“ pro zpřehlednění jednotlivých deployment type. V praxi se použití dvou a víc deployment typu používá např. když máme jeden type pro „primary user“ a druhý pro „primary device“. Když směřuje deployment type na uživatele, musí být zaškrtnuto „User must be logged in“ v **User Experience** okně. V opačném případě postačí „Whether or not user is logged in“. Ale zpátky k oknu **Content**. Zde specifikujeme, kde je balíček umístěn a také, co se má provést při akci instalace a odinstalace. Content

location zvolíme „\\WIN-DJTP5O5IBF9\SMS_PS1\Packages\Mozilla_Firefox“, do Installation program vyplníme námi vytvořený „Install.vbs“ a do Uninstall program zase „Uninstall.vbs“. Pokračujeme na další okno

Jak již bylo zmíněno výše, detekční pravidlo aplikace bude nastaveno na komentář u klíče registru, který zobrazuje informace do Programy a funkce. V okně **Detection Method** proto klikneme na Add Clause, kde zvolíme u Setting Type hodnotu „Registry“ a vybereme větev HKEY_LOCAL_MACHINE, jelikož instalace bude provedena pod systémem pro všechny uživatele a v tomto případě bude registr umístěn právě v HKLM. Kdybychom provedli instalaci pod právě přihlášeným uživatelem, bude použit registr v HKEY_CURRENT_USER, zkráceně HKCU. Registry zobrazující se v Programy a funkce jsou uloženy vždy na stejném místě na cestě:

- „SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall“: V případě 64-bitového softwaru.
- „SOFTWARE\WOW6432Node\Microsoft\Windows\CurrentVersion\Uninstall“: Pro 32-bitový software.

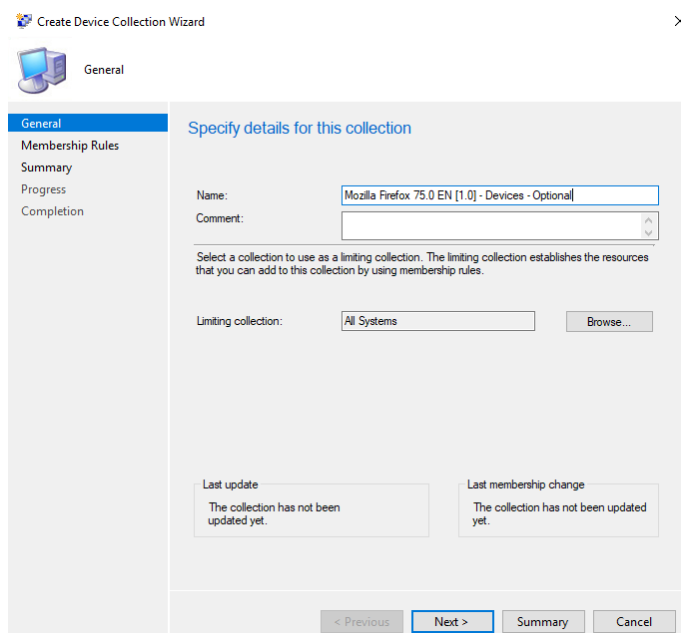
Do okna **Key** se ovšem vždy uvádí cesta 64-bitová a pro určení, že se jedná o 32-bitovou aplikaci, slouží zaškrťovací tlačítko „This registry key is associated with a 32-bit application on 64-bit systems“. Registr pro naši aplikaci se jmenuje „Mozilla Firefox 75.0 (x86 en-US)“, a proto výsledná cesta bude:

„SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\Mozilla Firefox 75.0 (x86 en-US)“

I když reálná adresa ještě obsahuje právě WOW6432Node, které nám SCCM přidá pomocí zaškrťovacího tlačítka výše. Do řádku Value patří název proměnné v onom registru a tím, že jsme se určili, že detekční pravidlo bude na proměnnou Comments, napíšeme Comments. Pozor, tento řádek je case-sensitive. U Date type vybereme String, jelikož náš komentář obsahuje jak čísla, tak písmena. V dolní části vybereme přepínač „This registry settings must satisfy the following rule to indicate the presence of this application“, protože chceme u detekčního pravidla ověřovat konkrétní řetězec. Proto zvolíme operator Equals a do Value vypíšeme název našeho balíčku „Mozilla Firefox 75.0 EN [1.0]“.

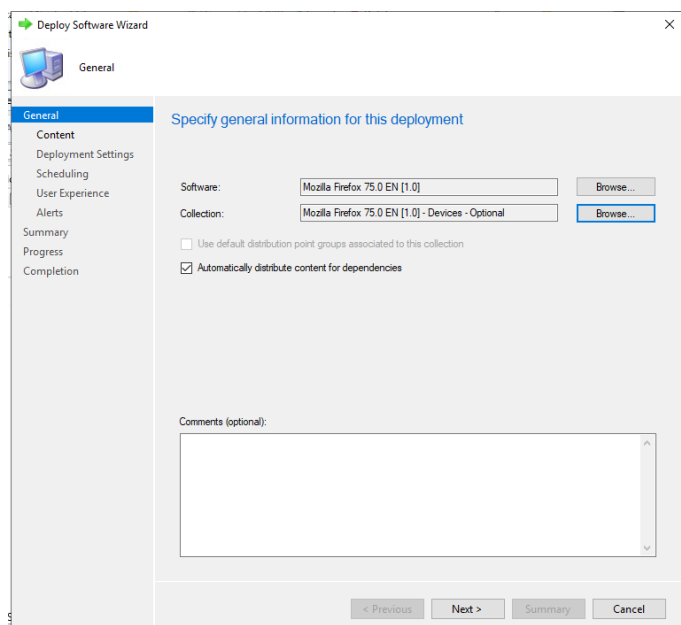
V okně **User Experience** nastavíme instalační chování na instalaci pod systémem. ovšem tímto výběrem dojde i k zvolení spuštění odinstalace pod systémem. Dále zvolíme požadavek na přihlášení, že uživatel nemusí být přihlášen během těchto akcí a umožnění uživateli manipulovat s balíčkem v rámci Software Center tzn. instalace a odinstalace přes Software Center. Maximální povolenou dobu akcí navýšíme na 240 minut, jelikož v tomto čase je kromě samotné instalace zahrnuto i stahování médií, které v některých případech trvá dlouho kvůli odlehlosti koncového zařízení od nejbližšího distribučního bodu. Odhadovanou dobu instalace zvolíme podle toho, jak dlouho reálně trvala instalace během vytváření instalačního balíčku a přičteme něco málo navíc. V našem případě instalace trvala necelé dvě minuty. K tomu jsme připočítali dobu stahování a jiné možné zpoždění a dostali jsme se na číslo patnácti minut.

V dalších oknech nebudeme provádět žádné úpravy a necháme je ve výchozím nastavení. Tím je aplikace sama o sobě vytvořena a dále musíme vytvořit kolekci, na kterou bude v budoucnu aplikace mířit. Vytvoření kolekce provedeme na SCCM ve složce Assets and Compliance pravým kliknutím na Device Collections a výběrem Create Device Collection.



Obrázek 15: Scénář nasazení aplikace – Vytvoření kolekce

Kolekci si pojmenujeme dle vlastního uvážení, ale platí stejné pravidlo její jedinečnosti. Dále je potřeba určit limitující kolekci, která v praxi slouží např. k limitaci na určitou organizační jednotku a počítače z jiné organizační jednotky do této kolekce nemohou být přidány. U kolekce můžeme zvolit, jak pravidelně dojde k obnovení této kolekce při změně členů v ní. Když máme vytvořenou aplikaci i kolekci, posledním krokem je propojení obou těchto komponent pomocí deploymentu. To se provede pravým kliknutím na námi vytvořenou aplikaci.



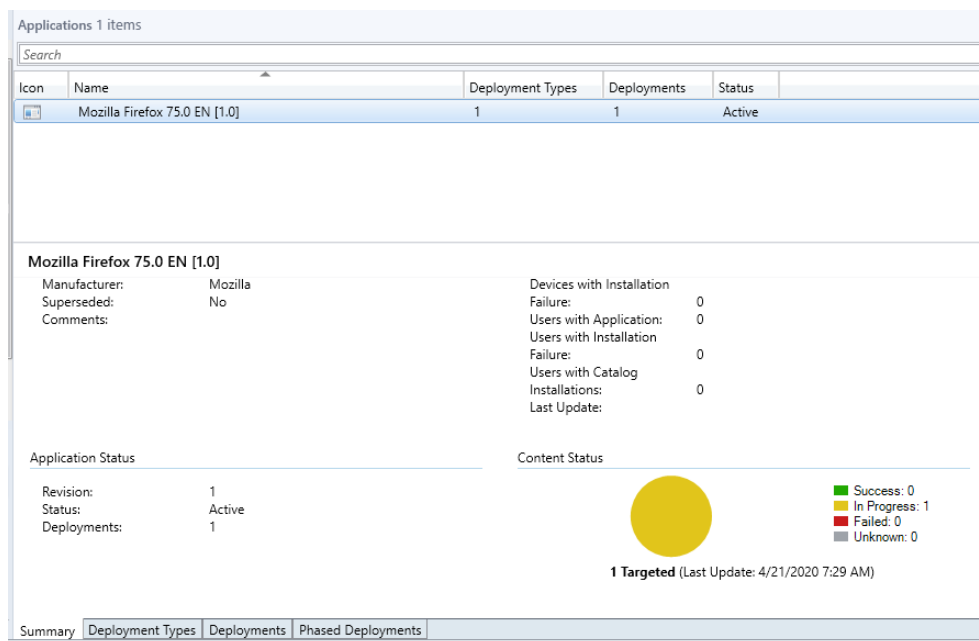
Obrázek 16: Scénář nasazení aplikace – Deployment General

Každý deployment musí jednoznačně mít aplikaci a kolekci, které propojuje. Zvolíme námi vytvořenou aplikaci i kolekci, necháme zaškrtnutou automatickou distribuci instalačních souborů na distribuční body a pokračujeme na další okno.

V okně **Content** musí dojít k zvolení distribučního bodu, kde aplikace bude nahrána. Můžeme vybrat buďto jeden nebo více bodů, nebo v našem případě zvolíme skupinu „All distribution points“, kterou jsme si předem připravili a která obsahuje všechny distribuční body našeho SCCM serveru.

V **Deployment settings** zvolíme akci, která se má provést, a způsob, jak bude balík uživateli nebo počítači doručen. Available znamená, že je balík poslán volitelně a uživatel si jej musí sám nainstalovat, popř. odinstalovat. Druhou možností je znemožnění těchto privilegií uživatele a násilná instalace ve zvoleném čase.

Čas, kdy dojde k zviditelnění balíčku, se nastavuje v okně **Scheduling**. V případě povinné instalace musíme zvýšit pozornost časům, aby nedošlo k instalaci moc brzy nebo příliš pozdě. Zbytek deploymentu zůstane v původní podobě.



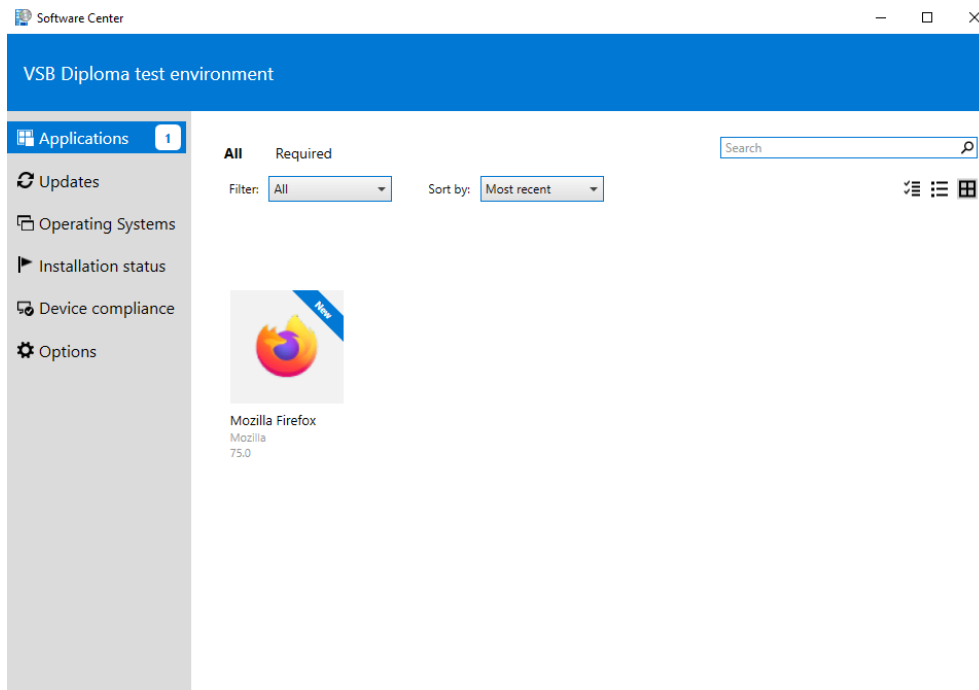
Obrázek 17: Scénář nasazení aplikace - Přehled aplikace

Po vytvoření deploymentu můžeme zpozorovat, že se v kartě Summary u naší aplikace status koláčového grafu změní z Unknown na In Progress. To znamená, že dochází ke kopírování z Content složky aplikace na distribuční body, které jsme zvolili během deploymentu. Doba trvání kopírování bude závislá na tom, jak daleko se distribuční bod nachází od primary site v kontextu rychlosti sítě.

Jedním z posledních kroků bude přidání uživatelů, nebo počítačů, do kolekce, kterou jsme si vytvořili a na kterou míří náš deployment. Přes pravé tlačítko myši vybereme Add resources, vyhledáme našeho testovacího klienta a přidáme jej do kolekce. Pro urychlení procesu přidání počítače do kolekce opět pravým tlačítkem myši zvolíme možnost Update membership, která vynutí aktualizaci kolekce.

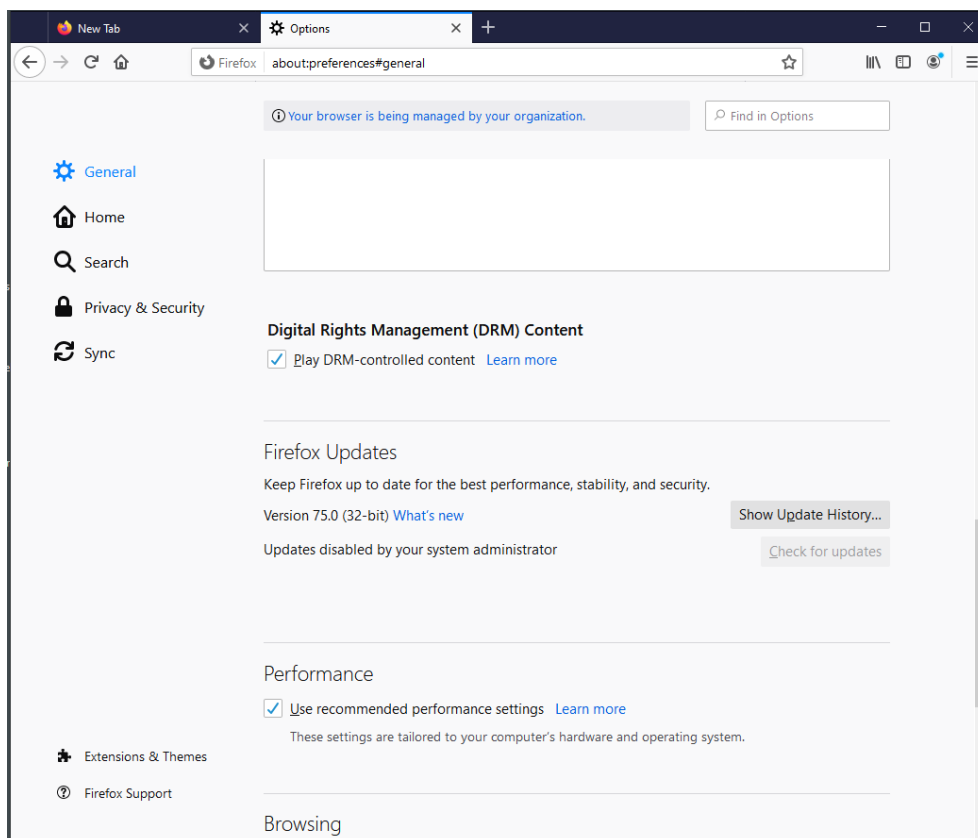
Posledním nastavením na SCCM bude úprava deployment type aplikace. Nutnost zaškrtnutí „Allow clients to use distribution points from the default site boundary group“ je to, že naše SCCM obsahuje pouze jeden distribuční bod v pouze jedné boundary group, která je defaultní. Současně zvolíme u možnosti „Deployment Download content from distribution point and run locally“. Po provedení všech úkonů můžeme v Software Center uvidět námi vytvořenou aplikaci, provést instalaci a odinstalaci instalačního balíčku. Občas se stává, že zobrazení instalačního balíčku neproběhne hned,

ale je nutné nějakou dobu vyčkat. K urychlení tohoto procesu se používá vynucení politik zařízení, na které balíček směřuje.



Obrázek 18: Software Center

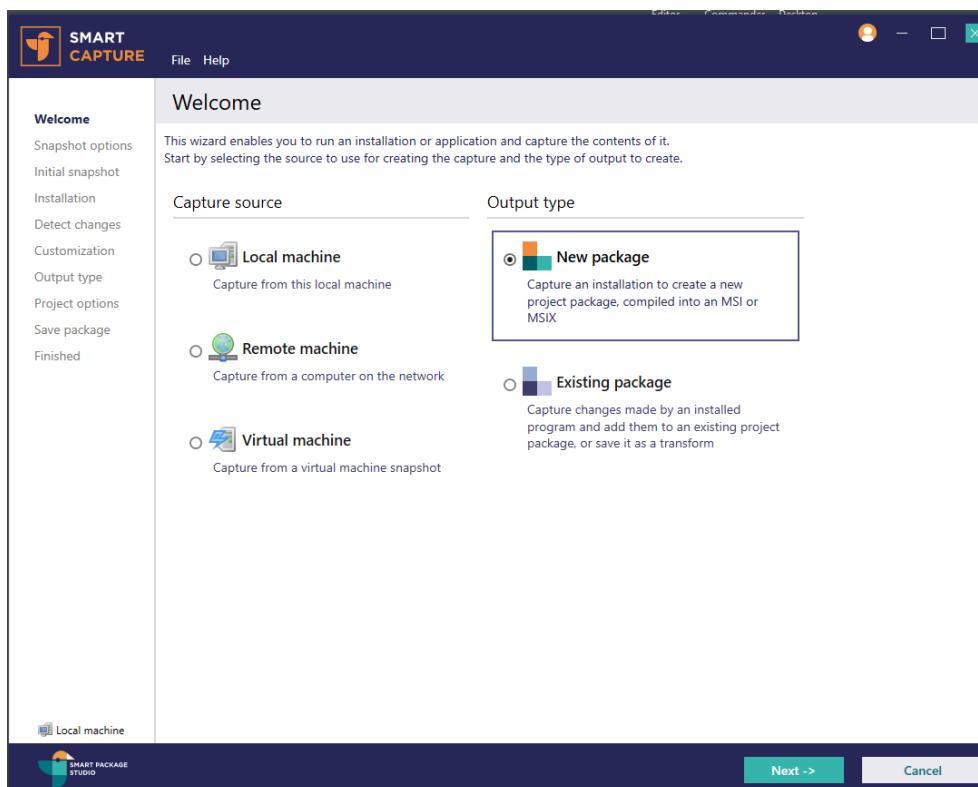
Po kliknutí na tlačítko Install uvidíme, v jaké stavu se instalace nachází. Prvně dojde ke stažení médií na lokální disk do složky „%WinDir%\ccmcache“, poté dojde ke spuštění definované akce – u instalace ke spuštění Install.vbs souboru. Poté Software Center oznámí, zda instalace proběhla v pořádku nebo vrátí chybovou hlášku. Po instalaci můžeme ověřit všechny nastavení, které by měly být aplikovány v rámci instalačního balíčku.



Obrázek 19: Kontrola zakázání automatických aktualizací

Další scénářem by bylo nasazení balíčku jako package model. Rozdíly, výhody i nevýhody tohoto řešení v porovnání s aplikačním modelem jsou detailněji popsány v kapitolách výše, ovšem za zmínku stojí to, že aplikační model je více uživatelsky přívětivý, kdežto package model je vhodný pro instalace, během kterých nemáme možnost detekčního pravidla. To zpravidla bývá, když instalační balíček obsahuje pouze např. zástupce na soubor umístěný na vzdáleném úložišti. Soubor samotný nenesou žádnou informaci, kterou bychom mohli poslat SCCM a jednoznačně tím určit, jestli se jedná o správný soubor či nikoliv. Co se stane, když chce zákazník jako standard pouze a výhradně aplikační model? Řešení vždy existuje a jmenuje se capture. Obecně se jedná o rozdíl dvou obrazů operačního systému se zaměřením na soubory a registry. Prvním krokem je vytvoření prvního obrazu, poté dojde k provedení potřebných změn, a nakonec se vytvoří druhý obraz. Ve finále se ještě upraví, které změny jsou redundantní, popř. se doplní ty, které chybí. Výsledkem capture je soubor MSI.

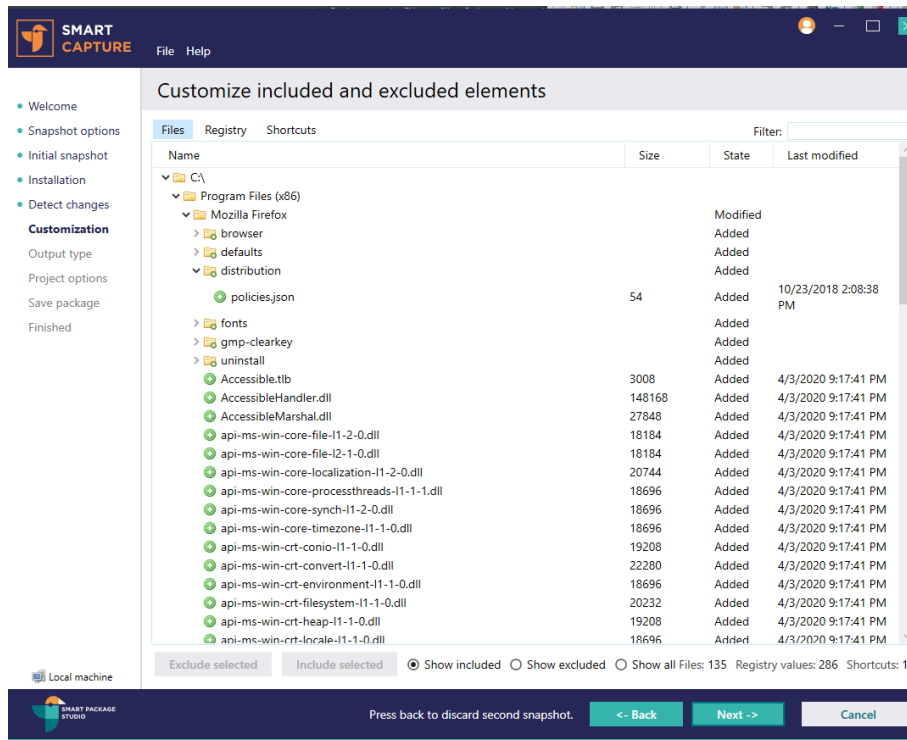
Pojďme si popsat, jak bychom capture použili pro případ Mozilly Firefox. Pro capture byla použita placená aplikace Smart Package Studio, která současně obsahuje nástroj Smart Capture. Mimo jiné nabízí nástroje pro editaci MSI, MSIX a Intune souborů, zobrazuje tipy a analýzy upravovaných souborů a jejich zlepšení a umožňuje testování našich vytvořených souborů. Po otevření Smart Capture se zobrazí uvítací okno, které nabízí vytvoření nového projektu nebo použití již existujícího. V našem případě zvolíme nový projekt a vybereme lokální zařízení, které chceme zdetekovat pro capture.



Obrázek 20: Smart Capture - Welcome

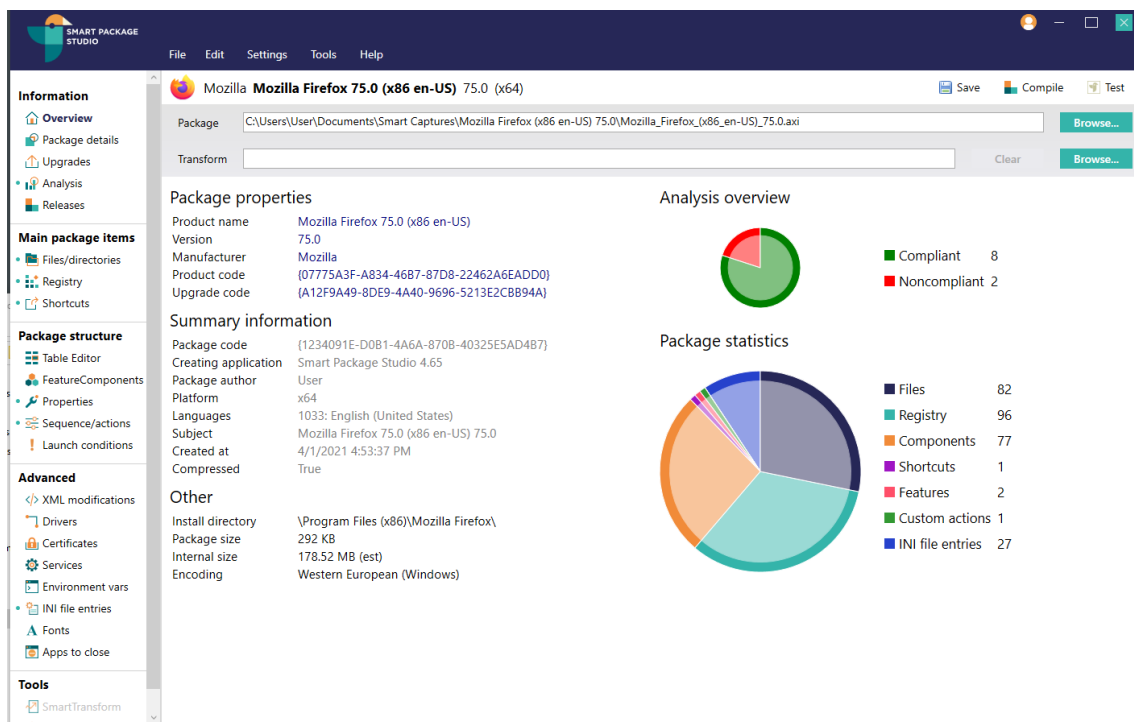
Další okno blíže specifikuje možnosti obrazu operačního systému, tzv. snapshotu, ale zde není nutné měnit nic, a tak nabídku přeskochíme a pokračujeme dále, kdy dojde k vytvoření počátečního snapshotu. Nástroj zobrazuje, které soubory a registry se aktuálně zpracovávají, a zobrazuje jejich celkový počet, který ovlivní to, jak dlouho se snapshot bude vytvářet. Okno pro provedení instalace nabízí prozkoumání lokálního disku a vybrání souboru k instalaci včetně definice instalačních přepínačů, které chceme použít během této simulované instalace. Po vložení cesty k souboru a přepínačů klikneme na tlačítko Execute nebo stlačíme klávesu F10 k simulaci instalace. Po dokončení instalace je nutné ručně provést všechny změny, které chceme promítnout do nově vytvořeného MSI souboru. V našem případě tedy postupně odebereme Mozilla Maintenance Service přes Přidat nebo odebrat programy, dále modifikujeme všechny tři soubory jako jsme to udělali pro aplikační model, přidáme registr pro Crash Reporter a klíč Comments ve větvi Uninstall a přidáme práva na složku Uninstall kvůli logům odinstalace. Pokud jsme si jistí, že všechny změny byly provedeny, můžeme pokračovat na další okno. V tom opět dojde ke skenování systémů a k vytvoření druhého snapshotu. Po dokončení uvidíme všechny rozdílnosti, které během těchto dvou obrazů nastaly. U tohoto okna pravděpodobně strávíme nejvíce času, protože musíme projít všechny změny. Začneme u změn souborů, kde nám Smart Capture již vyfiltruje nepodstatné soubory a složky, jako jsou dočasné soubory, soubory z paměti systému apod., což nám zmenší práci, ale i tak zůstane spousta souborů, které musíme jeden po druhém ověřit. Většina běžných instalací se chová stejně, a proto časem zjistíme, že dochází ke kopírování souborů do podobných složek, do podobných registrů, a právě tyto složky a větve ověříme. Mezi ně patří jednoznačně Program Files, AppData a třeba Start Menu složky. Pro registry to jsou větve Uninstall a pro uložení např. licenčních klíčů „HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\Microsoft“. Nástroj viditelně rozlišuje, co bylo

přidáno, a naopak, co bylo odebráno. Ověříme, jestli naše změny byly korektně rozpoznány a v opačném případě je opravíme.



Obrázek 21: Smart Capture – Customization

Nakonec dojde k vytvoření Smart Capture projektu s příponou axi, pro který ještě nastavíme parametry. Zvolíme název aplikace, která bude poté zobrazena po instalaci v Přidat nebo odebrat programy, její verzi, výrobce, výchozí instalační složku, zda se jedná o 64-bitovou aplikaci nebo jen 32-bitovou a kde se má projekt uložit.



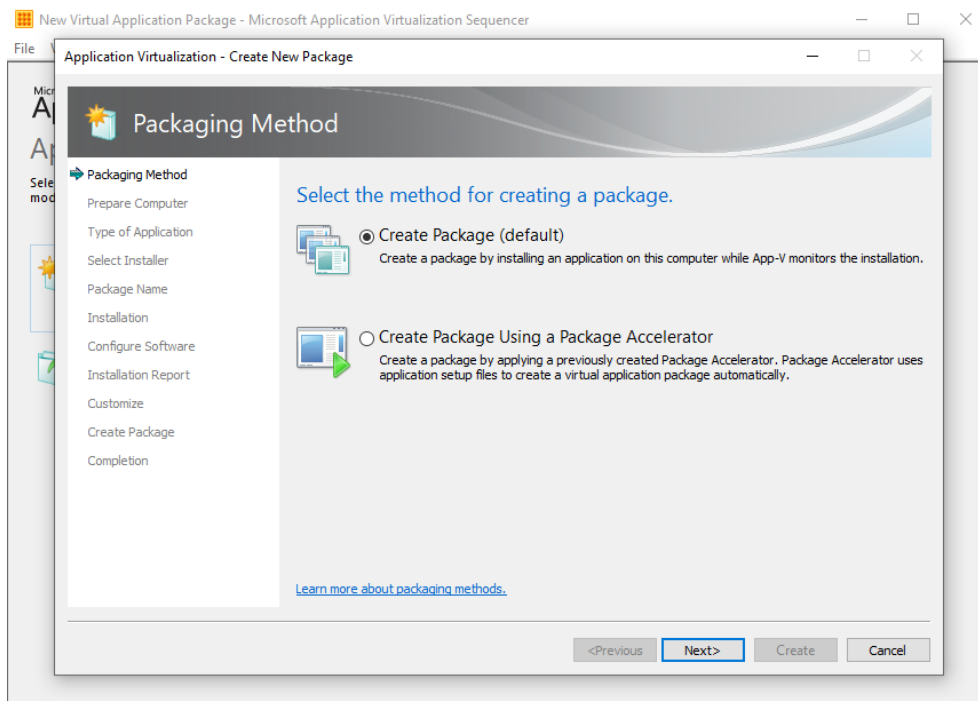
Obrázek 22: Smart Capture – Okno projektu

Smart Package Studio připraví vše potřebné za nás. Vytvoří nový Product code, zkomprimuje soubory a také poradí, co by bylo vhodné změnit podle jejich analýz a jestli vůbec projekt půjde zkompileovat do finálního MSI souboru. Pokud je vše v pořádku, můžeme zmáčknout tlačítko Compile, které vytvoří náš MSI soubor v projektové složce. Ta mimo jiné obsahuje:

- 1) Mozilla_Firefox_(x86_en-US)_75.0.axi – projektový soubor Smart Package Studio
- 2) Mozilla_Firefox_(x86_en-US)_75.0.log – soubor s logy kompilace
- 3) Mozilla_Firefox_(x86_en-US)_75.0.msi – výsledný MSI soubor
- 4) Mozilla_Firefox_(x86_en-US)_75.0.base.axi - projektový soubor Smart Capture
- 5) Složku ProgramFilesFolder – obsahující všechny soubory, které jsou v projektu kopírovány do složky Program Files

Capture je hotový, a proto si zkopírujeme výše zmíněný „Mozilla_Firefox_(x86_en-US)_75.0.msi“, nasadíme jej jako package model na SCCM a otestujeme instalaci. Původní myšlenkou ovšem bylo z obyčejného zástupce vytvořit MSI soubor, abychom získali záznam v Přidat nebo odebrat programy, abychom poté mohli zástupce detekovat pomocí SCCM a jeho aplikačního modelu. Ukázali jsme si, že tímto způsobem – capture - lze zpracovat i komplikovanější a komplexnější aplikace než jen jednoduchého zástupce.

Mezi další možnosti zpracování instalace je App-V. Pro ukázkou opět použijeme námi již velice známou Mozillu Firefox. Prvním krokem pro úspěšné vytvoření App-V balíčku je bezchybně nastavené prostředí na virtuálním stroji, na kterém je nainstalován App-V sekvencer. Nastavení je již probráno v teoretické části této diplomové práce, takže tento krok přeskočíme a přejdeme rovnou k vytváření pomocí aplikace Microsoft Application Virtualization Sequencer.



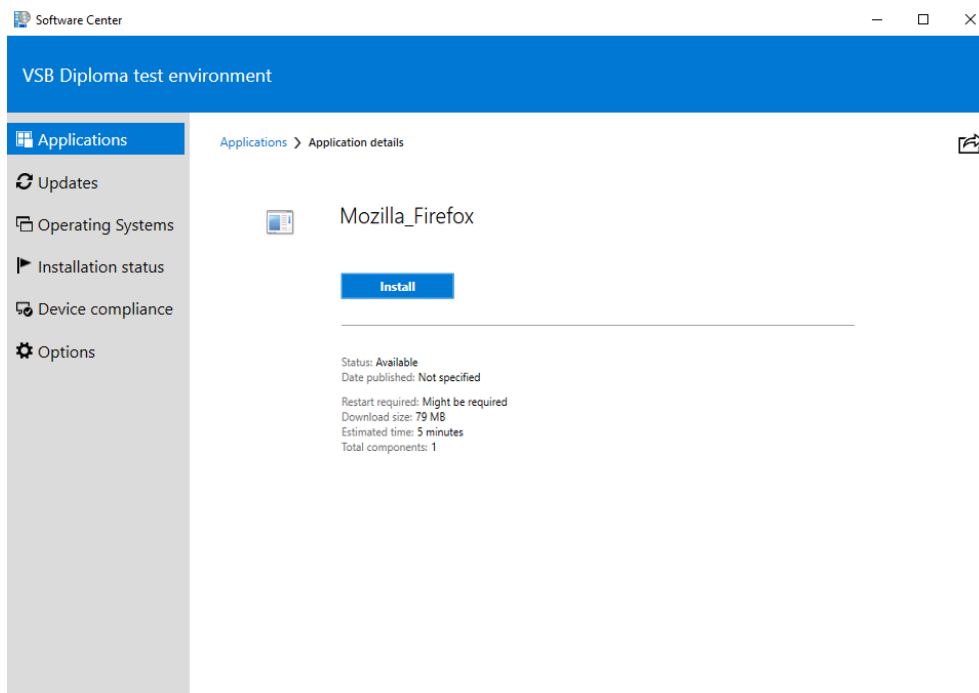
Obrázek 23: Microsoft Application Virtualization Sequencer

Obecně je App-V velmi podobné capture, takže se bude zdát, že provádíme stejné kroky. Zásadní rozdíly jsou až po vytvoření nových souborů a také to, jak s nimi lze pracovat. Prvním oknem zvolíme pouze vytvoření nového App-V balíčku a pokračujeme až k vytvoření vlastní instalace. Během

okna Installation dochází ke skenování systému a zaznamenávají se všechny změny souborů a registrů. Tady vidíme první rozdíl oproti capture, kde se nejdříve udělal první snapshot, pak se provedly změny a na konec se udělal druhý snapshot, kde náš soubor obsahoval právě změny mezi první a druhým snapshotem. Zde dochází k živému zaznamenávání změn. Pomocí tlačítka Run spustíme instalaci Mozilly Firefox vybráním EXE instalačního souboru, provedeme potřebné konfigurace kopírování souborů, přidání registrů apod. Když máme všechny změny dokončeny, zaškrtneme „I am finished installing.“ a pokračujeme na další okno. V tom máme možnost otevřít aplikaci a popřípadě vypnout vyskakovací okna typu souhlas s licenčními podmínkami, nastavení výchozího prohlížeče, odesílání anonymních dat, aj., což je opět rozdíl oproti capture, kde tato funkcionality nebyla. Během vytváření projektu dochází k ověřování, zda jsou nainstalovány všechny prerekvizity a jestli snapshot je v pořádku. Po dokončení průvodce dojde k vytvoření projektu a otevře se Microsoft Application Virtualization Sequencer Editor. Projekt si pojmenujeme Mozilla_Firefox a v editoru zkontrolujeme, které operační systémy jsou povoleny jako kompatibilní, opět projdeme a smažeme redundantní soubory a registry a v záložce Advanced zaškrtneme všechny zaškrťovací tlačítka „Allow all named objects to interact with the local system“, „Allow all COM objects to interact with the local system“ a v poslední řadě „Allow virtual applications full write permissions to the virtual file system“. Ještě jednou zkontrolujeme všechny záložky a uložíme projekt jako appv soubor. Výsledná složka po „kompilaci“ bude obsahovat:

- 1) Všechny soubory, které jsme použili během konfigurace během okna Installation
- 2) Mozilla_Firefox.appv
- 3) Mozilla_Firefox_DeploymentConfig.xml
- 4) Mozilla_Firefox_UserConfig.xml
- 5) report.xml
- 6) Mozilla_Firefox.msi

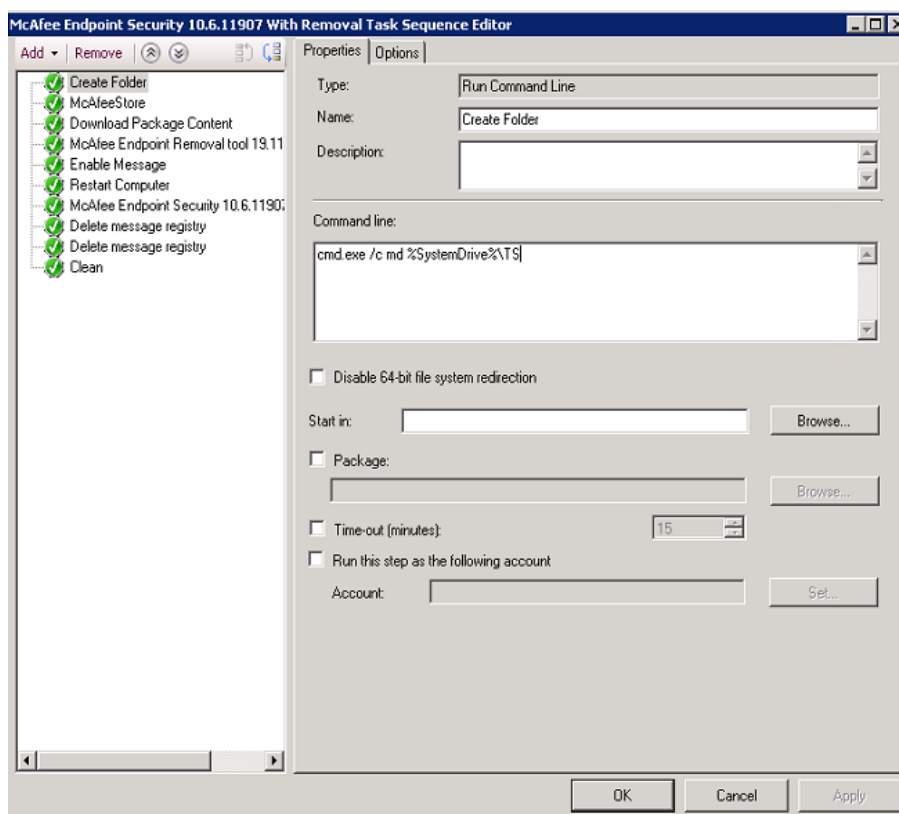
Pro nasazení na SCCM budou potřebné pouze soubory s příponou appv, dále XML soubor DeploymentConfig a UserConfig. Opět viditelná změna, kde výstupem capture byl MSI soubor, zde to jsou tři soubory – jeden APPV a dva XML. Na SCCM vytvoříme nový aplikační model, ale v okně General nebudeme volit „Manually specify the application information“, ovšem zvolíme jeden z předpřipravených typů Microsoft Application Virtualization 5. Pokud jsme cokoliv na App-V sekvenceru udělali špatně, SCCM má velice propracované ověřování souborů appv a v dalším kroku by došlo k chybě. Pokud je soubor appv správně vytvořen a balík bude funkční, SCCM pouze informuje, že je vše v pořádku. Balíček otestujeme vytvořením deploymentu na kolekci, která již obsahuje našeho testovacího klienta. Po aktualizaci politik se App-V Mozilla Firefox objeví v Software Center, přes který spustíme instalaci v případě, že je deployment nastaven jako optional. Záměrně jsme změnili název a nechali prázdná pole v konfiguraci Software Center, ať jde App-V balík rozlišit od prvního VBS balíku.



Obrázek 24: Software Center - App-V package

Posledním scénářem by bylo vytvoření vlastní task sequence. Pro task sequence použijeme příklad z praxe, a to na antivirus McAfee. Při vytvoření balíčku pro McAfee narazíme na problém, že jej nelze odinstalovat pomocí skriptů, ani přes Přidat nebo odebrat programy. Proto vendor poskytuje mimo instalační soubor také soubor pro odinstalaci, který musíme spustit před samotnou instalací. Na výběr máme z několika možností. První je nasazení zvlášť aplikace McAfee a zvlášť McAfee removal toolu, kde bychom u hlavní aplikace nastavili dependency aplikace pro removal tool. Tím by nejdříve proběhla odinstalace a následně proběhla instalace. Druhou možností je provedení totožné činnosti, jen u package modelu. První package model by byl pro hlavní aplikaci a druhý pro removal tool, kde bychom u hlavního package modelu nastavili „Run another program first“. Obě řešení vypadají obdobně, nabízí podobné možnosti, jenže posléze bychom natrefili na problémy, a proto si pojdme vysvětlit výhodu využití task sequence.

Prvním problémem je to, že removal tool během odinstalace odebere také některé komponenty firewallu a uživateli, který je připojen pomocí VPN, se toto VPN spojení přeručí a s tím souvisí i spojení s SCCM serverem, tudíž nedojde ke stažení druhé komponenty – instalace McAfee. V tomto případě by se uživatel ocitl bez antivirusu a neměl možnost jej zpět nainstalovat. Ano, po restartování počítače a následného zpětného připojení na VPN by k instalaci opět mohlo dojít, jenže tato akce potrvá minuty, a právě tyto minuty vytváří nebezpečí pro zařízení bez antivirusu, potažmo pro celou infrastrukturu, jelikož pokud by došlo k zavirování v okamžiku, kdy je počítač bez antivirusu a poté by došlo k připojení do firemní sítě, ohrožení jsou všichni. Bohužel tato teorie se několikrát promítla i ve skutečnosti, a proto na tuto problematiku autor poukazuje. Proto si pojdme ukázat jednotlivé kroky task sequence a vyhnout se problémům s použitím zřetězení ať už pomocí aplikačního modelu nebo package modelu.



Obrázek 25: Task Sequence - McAfee

Prvním krokem je vytvoření složky, do které se stáhnou všechna instalační média. Dodáme podmínku, že k vytvoření složky dojde pouze a tehdy, pokud neexistuje. Abychom nemuseli stále používat absolutní cestu k této složce, vytvoříme si proměnnou, která tuto cestu bude nést. V dalším kroku zvolíme package modely, které se stáhnou, tzn. McAfee hlavní aplikaci a McAfee removal tool. Tím zajistíme to, že i když dojde k přerušení spojení VPN, média budou stále lokálně stažena na cílovém stroji a instalace bude moci pokračovat. Nastavíme přepínač pro removal tool na „—accepteula —ALL -noreboot“ a přidáme exit kód 1. Ten EXE soubor vrátí v případě, že žádné McAfee nebylo nainstalováno. Po odinstalaci dojde k přerušení spojení, a proto přidáme do registru zprávu, která se objeví po znovu přihlášení uživatele a která jej po restartu uklidní a navede, co má dále dělat. Pokud infrastruktura používá BitLocker, dočasně jej vypneme pomocí příkazového řádku „manage-bde -protectors -disable C:“. Informaci o aktivitě BitLockeru zjistíme přes SCCM. V případě, že cílový stroj je notebook, zjistíme, jestli je jeho baterie napájena pomocí query na WMI „select * from Win32_Battery where Batterystatus > 0“. Po těchto krocích můžeme restartovat cílový stroj. Po zpětném naběhnutí se zobrazí námi vytvořená hláška a dojde ke spuštění instalace McAfee i když stroj není na VPN. Po instalaci odebereme hlášku, aby se nezobrazila znovu po restartování, a odebereme dočasnou složku s instalačními médii. Nakonec znovu aktivujeme BitLocker, pokud to je možné. Aby package nebo aplikační model mohly být použity v task sekvenci, je nutné zaškrtnout u každého modelu patřičné tlačítko. Přímou u task sequence ještě nastavíme, ať se dopředu stáhne všechn obsah před samotným spuštěním.

Doposud jsme prováděli tzv. upgrade softwaru, jenže u něj dochází k tomu, že všechna uživatelská data jsou smazána a uživatel si musí zpětně znovu nastavit danou aplikaci dle svých požadavků. Samozřejmě záleží na tom, jak chytrý je skript, doprovázející upgrade. Zpravidla se nejdříve

tato uživatelská data zálohují a po upgradu opět přepokopávají zpět na své místo. Abychom nemuseli řešit tento problém, někteří vendori nabízejí média pro aktualizaci. U MSI technologie hovoříme o MSP souborech. Tyto soubory nemažou data, jen aktualizují data, která jsou nová. Příkladem je Adobe Reader.

Během jednotlivých odstavců docházelo na poukazování chyb v rámci použití jedno či druhého způsobu užití modelu a technologie a bylo upozorňováno na vyhýbaní se omylům. Výhodou použití VBS je to, že pomocí něj lze modifikovat instalační balíček jakkoliv chceme. Můžeme zavírat procesy, zacházet se soubory a registry, přidávat oprávnění a jeví se jako jednoznačně nejlepší cesta ke zpracování instalačního balíčku, když máme funkční instalační soubor. Capture způsob je obecně výhodný, když chceme z obyčejného souboru vytvořit instalační soubor, tzn. nechceme řešit kopírování, ale chceme mít záznam v Přidat nebo odebrat programy a přes SCCM detekovat tento soubor. Při použití capture se obvykle použije i VBS, či jiný skriptovací jazyk, protože stejně je nutné zavírat procesy, popř. dělat jiné akce, které nejdou ošetřit v rámci MSI souboru. Poslední App-V balíček má výhodu v tom, že si žije vlastním životem v bublině a nic jej nemůže ovlivnit. To stejné platí i naopak, že App-V balíčkem nedojde k ovlivnění operačního systému. Když nedochází k vyrušení uživatele ani systému, není nutné zavírat procesy a tím se vyhýbáme užití skriptovacích jazyků.

Aplikační model je vhodný pro aplikace, které jsou detekovatelné, a měl by být použit v případech, kdy jsou balíčky posílány jako volitelné, tudíž uživatel vchází do Software center a vidí jednotlivé balíčky a nějakým způsobem s nimi interaguje. Výhodou je funkce schvalování balíčků administrátorem, detailnější předávání chybových kódů do Monitoringu, možnost poslání balíčků do jiných operačních systémů, než je Windows, automatické vyplňování informací o aplikačním modelu pro některé instalační typy a v poslední řadě také funkce supersedence pro případy, kdy dochází ke kolizím různých verzí jednoho softwaru. Package model je vhodné použít pro jednoduché příkazy PowerShellu, příkazového řádku, nebo třeba seskupení více příkazů v bat nebo cmd souboru, tedy kdy nemůžeme jednoznačně detekovat pomocí souborů nebo registrů, jestli došlo k žádané změně. Výhodou je např. možnost instalace balíčku rovnou z distribučního bodu, aniž by došlo k jeho stažení lokálně na úložiště klienta, ale také větší jednoduchost nastavení, není zde tolik informací, které je nutné vyplnit. Nevýhodou je jednoznačně méně detailní předávání chybových hlášek pro SCCM. Oba modely mají své výhody i nevýhody a jejich použití závisí na očekávání instalačního balíčku. Task sekvence je nadřazená oběma modelům, protože je rozšiřuje a přidává k nim další možnosti užití. Na druhou stranu je komplexnější a o to složitější na vytvoření.

4.3 Automatické vytváření balíčků pro aktualizace

Prvním krokem pro vytváření automatických balíčků aktualizací je samotné povolení aktualizací pomocí SCCM. Ve složce Administration a podsložce Site Configuration kliknutím na Sites se zobrazí v Configuration Manageru roleta, kde se nachází Add Site System Roles. Pomocí průvodce se proklikáme až na kartu System Role Selection, kde přidáme naši primary site další roli, kterou bude zprostředkovávat. Onou rolí je právě Software update point. V dalším okně zaměníme použití portů z 80 a 443 na porty 8530 a 8531. První možnost je výchozí pro WSUS 3.0 SP2 a druhá zase pro WSUS na Windows Serveru. Pokračujeme na další okno, kde se vybírá účet, pomocí kterého dojde ke stažení aktualizací. V našem případě nemáme dedikovaný účet pro tyto účely, a proto došlo k vybrání Administrator účtu. Nastavení synchronizace necháme výchozí, a to synchronizace z Microsoft Update.

Interval těchto synchronizací nastavíme na pevně na provedení každé čtyři hodiny a zapneme upozornění, které je vyvoláno při chybě synchronizace. V Supersedence Rules zapneme expiraci aktualizací, které jsou supersedovány jinou aktualizací. Velice důležitým bodem tohoto nastavení je klasifikace, která určí, jaké aktualizace budou vyhledávány na Microsoft Update. Základními jsou Critical updates, Definition updates, Security updates a updates. V následujícím okně zvolíme, na které produkty chceme aktualizace synchronizovat a dalším oknem vybereme, na které jazyky operačních systémů budou aktualizace vyhledány.

Přidání role Software update point je hotovo a dále je nutné provést úvodní synchronizaci ručně. Ve složce Software Library a podsložce Software Updates klikneme na All Software Updates a po kliknutí se nahoře v Configuration Manageru objeví roleta, ve které je Synchronize Software Updates. Tímto tlačítkem se spustí synchronizace, během které dochází k zapisování do logů „wsyncmgr.log“ a „WCM.log“, které se nachází v instalační složce SCCM. Ta ve výchozí instalaci bývá „C:\Program Files\Microsoft Configuration Manager\Logs“.

Pro automatické vytváření těchto aktualizací je nutné vytvořit Automatic Deployment Rule. Tyto pravidla je možné vytvořit ve složce Software Library a podsložce Software Updates, kde se nachází Automatic Deployment Rules. Pravým kliknutím myši na tuto položku vytvoříme nové pravidlo. V General napíšeme název tohoto pravidla a vybereme Template pomocí tlačítka Manage Templates, kde se nachází Patch Tuesday. Poté zvolíme kolekci, na kterou budou aktualizace mířit. V dalším okně zvolíme zasílání všech zpráv od klientů, ne pouze ty chybové a úspěšné. Dalším krokem je v Software Updates vybrat, které aktualizace budou vyhledávány. Zvolíme tedy Date Released or Revised s parametrem Last 2 months, což vyhledá aktualizace za poslední dva měsíce. Dále zvolíme jazyk angličtinu, produktu vybereme relevantní a Update Classification opět Critical Updates, Definition Updates, Security Updates, Updates a Tools. V Evaluation Schedule nastavíme interval, jak často má docházet k vytvoření balíčku s aktualizacemi. Nedává smysl dávat interval menší, než je samotná synchronizace s Microsoft Updates, takže zvolíme stejnou dobu, a to čtyři hodiny. Čas, kdy má dojít k samotné instalaci takto vytvořeného balíčku, se definuje v záložce Deployment Schedule. Zde zvolíme Client local time, aby dostali všichni uživatelé aktualizace ve stejný čas, viditelnost a možnost instalace volitelně dáme As soon as possible a nejzazší termín instalace dva dny po jejich zviditelnění. V okně User Experience nastavíme notifikace tak, že budou všechny viditelné v Software Center a zvolíme chování instalace nejzazšího termínu stylem, že po uplynutí této doby dojde k instalaci i restartu i mimo maintenance window. Pokud nedošlo k vytvoření deployment package, v dalším kroku jej musíme vytvořit. Nazveme ho např. Windows updates, zvolíme složku, kde se aktualizace nachází a zvýšíme prioritu na High. Dále zvolíme distribuční body, na které budou aktualizace nahrány, nastavíme stažení aktualizací z internetu a vybereme pouze anglický jazyk. V případě, že si chceme toto nastavení uložit jako Template pro další budoucí pravidlo, v Summary okně k tomu slouží tlačítko. Troubleshooting je možný pomocí SCCM ve složce Monitoring, kde vidíme jak status synchronizačních bodů, tak deployment balíčků aktualizací.

4.4 Linuxové zařízení pod správou SCCM

Posledním zařízením, které bude připojeno do domény, bude virtuální stroj s operačním systémem Ubuntu 12.04 LTR. Ubuntu není nikterak hardwarově náročné, a proto stroji bylo přiděleno dynamicky od 1 GB do 2000 MB RAM paměti a jedno procesorové jádro hostu.

Během instalace byla zvolena jako lokace Praha a jazyk klávesnice en-US. Dále byl vytvořen root uživatel sku0064 a počítač byl nazván linux-client. Po dokončení instalace operačního systému bylo nutné stáhnout klienta SCCM pro další operační systémy z oficiálních stránek Microsoftu, konkrétně „Config Mgr Clients for Linux.EXE“. Problémem bylo to, že Microsoft tento instalátor poskytuje s příponou exe, která není podporována Linuxem. Z tohoto důvodu se instalátor musí extrahovat na klientovi s OS Windows a poté se vyextrahované soubory musí nahrát na Linuxové zařízení. V našem případě došlo k získání souborů přímo na SCCM serveru a poté byly zkopírovány na Google drive, odkud byly staženy na linux-client počítači. Když byly všechny tři soubory „install“, „license.rtf“ a „ccm-Universalx64.tar“ na lokálním disku, mohlo dojít k ověření prerekvizit. První je, že jsme schopni z Linuxového zařízení se pingnout na SCCM server, který má IP adresu 192.168.1.105. V případě úspěchu můžeme pokračit k bodu dvě, kterým je pingnutí na název našeho SCCM serveru, konkrétně na „win-djtp5o5ibf9.vsb.local“. Aby tohle bylo možné, musí dojít k modifikaci dvou souborů. Prvním je „/etc/resolv.conf“, kde k výchozím hodnotám je nutné přidat následující:

```
nameserver 192.168.1.105
nameserver 127.0.0.1
search win-djtp5o5ibf9.vsb.local localdomain
```

Zdrojový kód 12: Obsah souboru /etc/resolv.conf

Druhým souborem je „/etc/hosts“, kde se přidala podobná hodnota jako do prvního souboru:

```
127.0.0.1 localhost
127.0.1.1 Linux-Client
192.168.1.105 win-djtp5o5ibf9.vsb.local
```

Zdrojový kód 13: Obsah souboru /etc/hosts

Takovou úpravou docílíme toho, že nám stále bude fungovat internetové připojení do celého světa a současně do naší interní sítě s možností pingnutí SCCM serveru pomocí hostname. Tento krok je nezbytný pro instalaci SCCM klienta. Druhou prerekvizitou je upravení práv souboru „install“ a to tak, že z něj uděláme spustitelný soubor příkazem „chmod +x install“. Třetí prerekvizitou je přístup k internetu, jelikož během instalace dojde ke stažení chybějících modulů potřebných k instalaci. Když jsou všechny prerekvizity splněny, musí dojít k přihlášení root uživatele v terminál okně, kterým je sku0064 s námi zvoleným heslem. Po přesunu do složky, kde se nachází stažené soubory k instalaci klienta SCCM, spustíme instalaci s následujícími parametry, kde „-mp“ určuje management point, což je náš SCCM server, a „-sitecode“ pouze specifikuje název naší site:

```
./install -mp win-djtp5o5ibf9.vsb.local -sitecode PS1 ccm-Universalx64.tar
```

Zdrojový kód 14: Instalační parametry SCCM klienta

Pokud byla instalace úspěšná, dojde k vytvoření složky „/var/opt/microsoft/cm/“, která obsahuje log soubory „scxcm.log“ a „scxcmprovider.log“, které nesou informace ohledně konektivity, problémů s připojením atp., a proto jsou vhodné k případnému troubleshootingu. Současně dojde k zviditelnění linux-client záznamu ve složce Assets and Compliance a podložce Devices na našem SCCM serveru v Configuration Manageru. V případě linuxových zařízení musí dojít k jejich schválení, a

to tak, že na daný záznam klikneme pravým tlačítkem myši a vybereme Approve. Od této chvíle je zařízení pod správou SCCM serveru.

5 Závěr

Autor této práce je zaměstnán na pozici balíčkáře, který se stará o vytváření instalačních balíčků a následně je nasazuje na SCCM servery k různým zákazníkům. Pomocí této diplomové práce mu byl rozšířen přehled o tom, jakým způsobem jsou v dnešní době vytvářeny servery starající se o správu aplikací, operačního systému a aktualizací systému v podnicích, firmách a dokonce mezistátních korporacích.

Seznámení se s SCCM bylo relativně prosté díky mnohaletým zkušenostem z oboru, ovšem největší potíže přišly během kooperace virtuálního prostředí VMM s instalacemi jak doménového řadiče, tak System center configuration manageru, jelikož byla potřeba oddělit interní síť firmy od testovací tak, aby nedošlo k narušení či nechtěnému a neoprávněnému vniknutí do ní. K tomuto pomohl PfSense virtuální stroj, který slouží jako softwarový router.

Po vytvoření prostředí bylo dalším problémem zprovoznění jak viditelnosti jednotlivých serverů, tak jejich spolupráce, ať už na úrovni AD, tak sebe samotných. SCCM server totiž obsahuje mnoho prerekvizit, bez kterých není ani možné spustit instalaci Configuration Manageru samotného. Výhrou je, když se po dvou hodinách objeví okno říkající, že instalace proběhla úspěšně. Tím ovšem začíná zahájení vytváření politik a GPO, vytvoření nových uživatelů s různými právy a rozjetí distribučních bodů. Následovalo připojení testovacího klienta do domény a test doručení alespoň primitivního příkazového řádku přes Software Center k otestování správné konektivity mezi koncovými zařízeními a SCCM.

Po otestování a ověření funkčnosti prostředí mohlo dojít k vytvoření instalačního balíčku z aplikace Mozilla Firefox, která je jedním z nejnámějších internetových prohlížečů. Do balíčku byly přidány konfigurace jako vypnutí automatických aktualizací, smazání zástupce na ploše a přidání registrů k vypnutí některých funkcionalit této aplikace. Tyto konfigurace byly provedeny ve skriptu stavěném na skriptovacím jazyku Visual Basic Script, ve kterém autor vymyslel vlastní funkce pro jednotlivé modifikace. Ke stejné modifikaci autor použil druhou variantu pomocí funkce capture aplikace Smart Package Studio a také využil technologii App-V od společnosti Microsoft.

Když bylo připraveno prostředí i instalační balíček, mohlo dojít k jeho nasazení na SCCM. Tento proces byl detailně zmapován a zdokumentován v poslední kapitole této práce. Jednalo se také o test instalace balíčku pomocí Software Center s ohledem na co nejlepší uživatelsky přívětivý vzhled. Proto byla přidána ikona balíčku a také popisky definující bližší informace ohledně aplikace. Balíček byl na testovací zařízení poslán jako volitelný, proto si mohl uživatel sám zvolit, kdy dojde k instalaci a také mohl balík odinstalovat.

V rámci práce se také otestovala funkcionálnost task sekvencí, konkrétně pro případ užití antivirusu McAfee, kde dochází k problémům během upgradu. Ty se pomocí task sekvence odstranily a na závěr došlo k porovnání jednotlivých příkladů aplikačního modelu, package modelu a zmíněných sekvencí včetně jejich výhod a nevýhod.

Dalším bodem bylo vytvoření role SCCM, která měla na starosti automaticky vytvářet balíčky sloužící pro aktualizace jak klientských zařízení, tak i serverových. Pro zprovoznění této služby došlo k přidání nové ADR, která se synchronizovala s Microsoft Updates každé čtyři hodiny. Po této synchronizaci došlo ve stejnou dobu k vytvoření nové skupiny aktualizací, které byly nalezeny a které

byly distribuovány dále. Distribuce proběhla na zařízení, které byly součástí kolekce „All desktop and Server Clients“ a na kterou mířil automaticky vytvořený deployment.

Posledním řešením praktické části byla instalace Systém center configuration manager klienta na Linuxové zařízení, které obsahovalo OS Ubuntu 12.04 a byl virtualizován jako všechny ostatní stroje v našem testovacím prostředí. Během instalace docházelo k upravování zejména síťových souborů a jiných síťových nastavení. Poté proběhla instalace pod root uživatelem a bylo otestování připojení z SCCM serveru na linux-client počítač.

6 Seznam použité literatury a zdrojů

1. *Co je Configuration Manager? - Configuration Manager.* [online]. Dostupné z: <https://docs.microsoft.com/cs-cz/mem/configmgr/core/understand/introduction/> [cit. 2021-04-24].
2. *Packages and programs in Configuration Manager.* [online]. Dostupné z: <https://docs.microsoft.com/en-us/configmgr/apps/deploy-use/packages-and-programs/> [cit. 2021-04-21].
3. *Create applications in Configuration Manager.* [online]. Dostupné z: <https://docs.microsoft.com/en-us/configmgr/apps/deploy-use/create-applications/> [cit. 2021-04-21].
4. *Introduction to software updates in Configuration Manager.* [online]. Dostupné z: <https://docs.microsoft.com/en-us/configmgr/sum/understand/software-updates-introduction/> [cit. 2021-04-21].
5. *Introduction to reporting in Configuration Manager.* [online]. Dostupné z: <https://docs.microsoft.com/en-us/configmgr/core/servers/manage/introduction-to-reporting/> [cit. 2021-04-21].
6. *SCCM 1902 Install Guide Using Baseline Media.* [online]. Dostupné z: <https://www.prajwaldesai.com/sccm-1902-install-guide-using-baseline-media/> [cit. 2021-04-21].
7. *Configuring Firewall Settings For Configuration Manager 2012 R2.* [online]. Dostupné z: <https://www.prajwaldesai.com/configuring-firewall-settings-for-configuration-manager-2012-r2/> [cit. 2021-04-21].
8. *CONFIGURE SCCM 2012 BOUNDARIES.* [online]. Dostupné z: <https://www.systemcenterdudes.com/configure-sccm-2012-boundaries/> [cit. 2021-04-21].
9. *SCCM 2012 DISTRIBUTION POINT INSTALLATION.* [online]. Dostupné z: <https://www.systemcenterdudes.com/sccm-2012-distribution-point-installation/> [cit. 2021-04-21].
10. *Using SCCM 2012 RC in a LAB – Part 4. Configuring Client Settings and adding roles.* [online]. Dostupné z: <https://www.niallbrady.com/2011/10/29/using-sccm-2012-rc-in-a-lab-part-4-configuring-client-settings-and-adding-roles/> [cit. 2021-04-21].
11. *Windows Installer - Win32 apps.* [online]. Dostupné z: <https://docs.microsoft.com/en-us/windows/win32/msi/windows-installer-portal/> [cit. 2021-04-24].
12. *NSIS Users Manual.* [online]. Dostupné z: <https://nsis.sourceforge.io/Docs/> [cit. 2021-04-24].
13. *InstallShield Download Your Free Trial Today.* [online]. Dostupné z: <https://www.reverera.com/install/products/installshield.html/> [cit. 2021-04-24].
14. *Article: InstallShield Setup Silent Installation Switches.* [online]. Dostupné z: <http://www.itninja.com/blog/view/installshield-setup-silent-installation-switches/> [cit. 2021-04-21].

15. *Inno Setup*. [online]. Dostupné z: <https://jrsoftware.org/isinfo.php> [cit. 2021-04-24].
16. *What is VBScript? Introduction & Examples*. [online]. Dostupné z: <https://www.guru99.com/introduction-to-vbscript.html> [cit. 2021-04-24].
17. SKULINA, J. *Absolvování individuální odborné praxe*. Ostrava, 2018. 30 stran. Bakalářská práce. VŠB-TUO, Fakulta elektrotechnicky a informatiky. Vedoucí práce Ing. Dohnálek Pavel.
18. *AutoIt Scripting Language*. [online]. Dostupné z: <https://www.autoitscript.com/site/autoit/> [cit. 2021-04-24].
19. *Enable the App-V in-box client (Windows 10) - Windows Application Management*. [online]. Dostupné z: <https://docs.microsoft.com/en-us/windows/application-management/app-v/appv-enable-the-app-v-desktop-client/> [cit. 2021-04-24].
20. ALVAREZ, A. *Microsoft Application Virtualization Advanced Guide*. 1. vyd. Birmingham: Packt Publishing Ltd, 2012. ISBN 978-1-84968-448-4.
21. DESHEV, H. *Pro Windows PowerShell*. 1. vyd. New York: Springer-Verlag, 2008. ISBN 978-1-4302-0546-3.
22. MARTINEZ, S. - DAALMANS, P. - BENNETT, B. *Mastering System Center 2012 R2 Configuration Manager*. 1. vyd. Indianapolis: John Wiley & Sons, Inc., 2014. ISBN 978-1-118-82170-1.
23. TULLOCH, M. *Understanding Microsoft Virtualization Solutions: From the Desktop to the Datacenter*. 2. vyd. Redmond: Microsoft Press, 2010.
24. BOSANAC, D. *Scripting in Java™: Languages, Frameworks, and Patterns*. 1. vyd. Boston: Pearson Education, Inc., 2008. ISBN 978-0-321-32193-0.
25. TULLOCH, M. - PERRIMAN, S. *Introducing Microsoft System Center 2012 R2: Technical Overview*. 1. vyd. Redmond: Microsoft Press, 2013. ISBN 978-0-7356-8283-2.
26. TULLOCH, M. *Introducing Windows Server 2012 R2*. 1. vyd. Redmond: Microsoft Press, 2013. ISBN 978-0-7356-8278-8.
27. MEYLER, K. – HOLT, B. – OH, M. – SANDYS, J. – RAMSEY, G. *System Center 2012 Configuration Manager: UNLEASHED*. 1. vyd. USA: Pearson Education, Inc., 2013. ISBN 978-0-672-33437-5.
28. TMURGENT TEAM *Packaging for App-V & MSIX: Packaging Concepts*, 1. vyd., neznámé: TMurgent Technologies. ISBN neznámé.

7 Seznam příloh

- A. Příloha v IS EDISON, zdrojový kód skriptu „Install.vbs“
- B. Příloha v IS EDISON, zdrojový kód skriptu „Uninstall.vbs“