

Česká Zemědělská Univerzita v Praze

Provozně ekonomická fakulta

Katedra informačních technologií



**Automatizace pomocí VMware
vRealize Automation**

Diplomová práce

Vedoucí diplomové práce: **Ing. Jiří Vaněk, Ph.D.**

Autor: **Bc. Tibor Marchyn**

© Praha 2020

ČESKÁ ZEMĚDĚLSKÁ UNIVERZITA V PRAZE

Provozně ekonomická fakulta

ZADÁNÍ DIPLOMOVÉ PRÁCE

Bc. Tibor Marchyn

Systemové inženýrství a informatika

Informatika

Název práce

Automatizace pomocí VMware vRealize Automation

Název anglicky

Automation with VMware vRealize Automation

Cíle práce

Cílem práce je implementace produktu VMware vRealize Automation v prostředí vybrané společnosti za účelem zefektivnění každodenních úkolů vývojářů a testerů pomocí automatizace ve virtuálním prostředí a zjednodušení interních procesů IT oddělení.

Díličí cíle práce:

- Charakteristika virtualizační technologie
- Možnosti VMware vRealize Automation a další možné alternativy
- Analýza stávajícího stavu v organizaci
- Implementace VMware vRealize Automation
- Analýza a komparace finálního a původního stavu včetně vyčíslení celkových nákladů spojených s vlastnictvím (TCO)
- Závěry a zhodnocení

Metodika

Teoretická část vychází z bakalářské práce autora a charakteristiky virtualizační technologie, studiu a analýze dostupných zdrojů a literatury o produktu. Přestože vybraná společnost už daný produkt zakoupila, stručně se bude také věnovat alternativním produktům nebo metodám jak dosáhnout podobného cíle.

Praktická část se zaměří na analýzu stávajícího stavu ve vybrané společnosti, definování a charakteristice procesů stanovených IT oddělením a činnostem vykonávaným zaměstnanci, které jsou ve spojení s procesy zdlouhavé a je potřeba je zjednodušit. Po kompletizaci informací z analýzy se naváže konkrétní implementací produktu (instalace, konfigurace), přípravou prostředí, úpravou dílčích procesů a zhodnocení celkových nákladů spojených s vlastnictvím (Total Cost of Ownership – TCO).

Závěr práce bude patřit shrnutí a zhodnocení implementace.

Doporučený rozsah práce

60 – 80 stran

Klíčová slova

virtualizace, VMware, vRealize Automation, automatizace, linux, CentOS, Windows, datacentrum, DevOps

Doporučené zdroje informací

EISNER, Matthias, SANDFORT, Yves. Learning vRealize Automation 7.2 [Online]. Birmingham: Packt Publishing, 2017. Dostupné z: https://learning.oreilly.com/videos/learning-vrealize-automation/9781787281882/9781787281882-video1_1

MARCHYN, Tibor. Analýza a porovnání virtualizačních nástrojů. Praha, 2014. Bakalářská práce. Česká zemědělská univerzita v Praze. Vedoucí práce Ing. Zdeněk Votruba.

NOVAK, Karel, VALSECCHI, Paolo, MAURO, Andrea, GAVANDA, Martin. Mastering VMware vSphere 6.7 – Second Edition. Birmingham: Packt Publishing, 2019. ISBN 978-1-78961-337-7.

Powell, J. Mastering vRealize Automation 6.2. Birmingham: Packt Publishing, 2015. ISBN 978-1-78217-377-9.

RAJENDRAN, Sriram. Learning VMware vRealize Automation. Birmingham: Packt Publishing, 2016. ISBN 978-1-78588-583-9.

ROBERTS, Trevor, DOORN, Yvo van, SIGLER, Egle a ATWELL, Josh. DevOps for VMware administrators. Upper Saddle River, NJ: VMWare Press, 2015. ISBN 978-0-13384-647-8.

Předběžný termín obhajoby

2019/20 LS – PEF

Vedoucí práce

Ing. Jiří Vaněk, Ph.D.

Garantující pracoviště

Katedra informačních technologií

Elektronicky schváleno dne 8. 5. 2019

Ing. Jiří Vaněk, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 14. 10. 2019

Ing. Martin Pelikán, Ph.D.

Děkan

V Praze dne 31. 03. 2020

Čestné prohlášení:

Prohlašuji, že svou diplomovou práci „Automatizace pomocí VMware vRealize Automation“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce. Jako autor uvedené diplomové práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne:

Podpis:

Poděkování:

Děkuji vedoucímu diplomové práce Ing. Jiřímu Vaňkovi, Ph.D. za věcné připomínky, za možnost psát práci na toto téma a za vedení mé práce.

Automatizace pomocí VMware vRealize Automation

Abstrakt

Diplomová práce se zabývá technologií virtualizace a její automatizace pomocí platformy VMware vRealize Automation. Teoretická část stručně popisuje historii a typy virtualizace, samotnou platformu VMware vRealize Automation a alternativní produkty, kterými je možné dosáhnout podobného cíle. Praktická část popisuje současný stav v organizaci před implementací platformy, samotnou implementaci a reálné případy použití. Poslední část se věnuje ekonomickému pohledu a zhodnocení celkových nákladů spojených s vlastnictvím (Total Cost of Ownership – TCO).

Klíčová slova

Virtualizace, VMware, vRealize Automation, Automatizace, Linux, CentOS, Windows, Datacentrum, DevOps

Automation with VMware vRealize Automation

Abstract

This thesis is focused on virtualization technology and its automation using VMware vRealize Automation platform. First theoretical part of the thesis describes history and types of virtualization technology, VMware vRealize Automation platform itself and alternative products, which could achieve similar goals. Practical part is about current situation in company before implementation of product, implementation itself and showing real use cases. Last part belongs to economy point of view and evaluation of total costs of ownership (TCO).

Keywords

Virtualization, VMware, vRealize Automation, Automation, Linux, CentOS, Windows, Datacenter, DevOps

Obsah

1	ÚVOD	12
2	CÍL PRÁCE A METODIKA	14
2.1	CÍL PRÁCE	14
2.2	METODIKA	14
3	TEORETICKÁ VÝCHODISKA	15
3.1	HISTORIE VIRTUALIZACE	15
3.2	SOUČASNOST VIRTUALIZACE	16
3.3	HISTORIE A SOUČASNOST VMWARE VREALIZE AUTOMATION	16
3.4	DEFINICE VIRTUALIZACE	17
3.4.1	<i>Sumarizace výhod a nevýhod virtualizace</i>	19
3.4.2	<i>Příklady vhodné a nevhodné virtualizace</i>	20
3.4.3	<i>Základní pojmy a terminologie</i>	20
3.5	AUTOMATIZACE VE VIRTUALIZACI	22
3.5.1	<i>Produkty pro automatizaci IT procesů</i>	22
3.6	VMWARE VREALIZE AUTOMATION (VRA)	27
3.6.1	<i>vRA obecně</i>	27
3.6.2	<i>Benefity vRA</i>	28
3.6.3	<i>Běžné případy použití vRA</i>	28
3.6.4	<i>Dostupné edice a licencování</i>	29
3.7	ARCHITEKTURA VRA 7.X	30
3.7.1	<i>Základní služby vRealize Automation Appliance</i>	31
3.7.2	<i>Služby Infrastructure as a Service (IaaS) serveru</i>	33
3.7.3	<i>vRealize Orchestrator (vRO)</i>	38
3.8	TYPY NAsAZENÍ VRA (DEPLOYMENT TYPES / DEPLOYMENT DESIGN)	43
3.8.1	<i>Přístup před nasazením</i>	43
3.8.2	<i>Části návrhu</i>	47
3.8.3	<i>Návrh prostředí vRealize Automation</i>	50
4	VLASTNÍ PRÁCE	60
4.1	CHARAKTERISTIKA SPOLEČNOSTI	60
4.2	ANALÝZA STAVU PŘED IMPLEMENTACÍ (SOUČASNÝ STAV)	60
4.2.1	<i>Charakteristika prostředí virtualizace</i>	61
4.2.2	<i>Charakteristika hardware a software</i>	63
4.2.3	<i>Charakteristika procesů</i>	67
4.2.4	<i>Analýza a kalkulace provozních nákladů před automatizací</i>	74
4.3	NÁVRH ŘEŠENÍ AUTOMATIZACE	76
4.3.1	<i>Požadavky společnosti</i>	76
4.3.2	<i>Hodnotící kritéria při výběru řešení</i>	77
4.3.3	<i>Vícekritériální hodnocení variant</i>	77
4.3.4	<i>Návrh nového prostředí a architektury</i>	78
4.3.5	<i>Instalace a konfigurace VMware vRealize Automation</i>	83
4.3.6	<i>Nasazení automatizace</i>	92
5	VÝSLEDKY A DISKUSE	104
5.1	ÚPRAVA PROCESŮ	104
5.2	KOMPARACE PŮVODNÍHO A NOVÉHO STAVU	106
5.3	KALKULACE TCO	108
5.4	DOPORUČENÍ	109
6	ZÁVĚR	110
7	SEZNAM POUŽITÝCH ZDROJŮ	112

Seznam obrázků

Obrázek 1 - Typ 1 (vlevo).....	19
Obrázek 2 - Typ 2 (vpravo)	19
Obrázek 3 - High-level schéma OpenStack	23
Obrázek 4 - High-level schéma Terraform	25
Obrázek 5 - Přehled vRA.....	27
Obrázek 6 - Možnosti integrací vRA	29
Obrázek 7 - Portable License Unit.....	29
Obrázek 8 - Interakce mezi Model Manager a DEM.....	36
Obrázek 9 - Schéma interakcí komponent IaaS.....	38
Obrázek 10 - vRO endpoint-y.....	39
Obrázek 11 - Event Broker – Svázání (Subscribe) nového workflow s událostí.....	40
Obrázek 12 - Příklad použití podmínek pro spuštění workflow	41
Obrázek 13 - VMPS Master Workflow	41
Obrázek 14 - Editace workflow	42
Obrázek 15 - Editace skriptu jednoho z úkolů.....	42
Obrázek 16 - Etapy návrhu prostředí	43
Obrázek 17 - Typu clusterů	50
Obrázek 18 - Minimální typ nasazení.....	54
Obrázek 19 - Enterprise deployment	56
Obrázek 20 - Clustery	63
Obrázek 21 - High-level infrastruktura před implementací.....	64
Obrázek 22 - Příklad vPC	66
Obrázek 23 - VMware cluster.....	81
Obrázek 24 - Konfigurační průvodce – Volba typu nasazení vRA	82
Obrázek 25 - Konfigurační průvodce – Volba umístění rolí	82
Obrázek 26 - High-level architektura vRA.....	83
Obrázek 27 - Výběr souboru OVA	84
Obrázek 28 - přizpůsobení operačního systému	84
Obrázek 29 - Stažení Management Agent-a a nastavení NTP serveru.....	85
Obrázek 30 - poskytnutí detailů připojení	86
Obrázek 31 - Uživatel, pod kterým bude běžet Management Agent.....	86

Obrázek 32 - Konfigurace SQL Server databáze.....	87
Obrázek 33 - Konfigurace agenta pro vSphere.....	88
Obrázek 34 - Instalace komponent a rolí	88
Obrázek 35 - Dashboard vRA.....	89
Obrázek 36 - Přihlášení k výchozímu tenantu	89
Obrázek 37 - Nastavení práv pro první administrátorský účet	90
Obrázek 38 - Definice Fabric Group	91
Obrázek 39 - Vytvoření rezervace	91
Obrázek 40 - Vytvoření rezervace	92
Obrázek 41 - Ukázka dat z tabulky správy IP adres	93
Obrázek 42 - Základní parametry blueprintu.....	95
Obrázek 43 - Rozšiřující parametry blueprintu	95
Obrázek 44 - Návrhář blueprintu	96
Obrázek 45 - Definice zdrojové šablony a způsob nasazení	96
Obrázek 46 - Definice profilů v blueprintu	97
Obrázek 47 - Vlastnosti instance v návrhu	97
Obrázek 48 - Editor formuláře žádosti	98
Obrázek 49 - Publikování blueprintu.....	98
Obrázek 50 - Definice Entitlement katalogu	99
Obrázek 51 - Definice Entitlement katalogu	99
Obrázek 52 - Katalog pro uživatele	100
Obrázek 53 - Editor workflow	101
Obrázek 54 - Přiřazení workflow k události.....	102
Obrázek 55 - Vytvoření požadavku na VM.....	103
Obrázek 56 - Průběh zpracování požadavku na VM	103

Seznam tabulek

Tabulka 1 - dostupné edice VMware vRA	29
Tabulka 2 - Porovnání funkcionality v edicích.....	30
Tabulka 3 - HW požadavky pro jednotlivé role	53
Tabulka 4 - Rozdíl v počtu instancí jednotlivých nasazení	55
Tabulka 5 - Maximální počet položek	55
Tabulka 6 - Mzdový přehled – IT a Telekomunikace	75

Tabulka 7 - Přehled nákladů měsíčně a sumárně za 3 roky.....	75
Tabulka 8 - Vícekriteriální hodnocení variant.....	78
Tabulka 9 - Přehled nákladu měsíčně a sumárně za 3 roky po implementaci.....	107

Seznam diagramů

Diagram 1 - Využití zdrojů.....	62
Diagram 2 - Proces vytvoření nové VM.....	71
Diagram 3 - Proces změny stávající VM.....	73
Diagram 4 - Proces vytvoření nové VM s implementovanou automatizací.....	105
Diagram 5 - Proces změny VM s implementovanou automatizací	106

1 Úvod

V dnešní době je virtualizace (viz kapitola 3.4) brána jako standardní součást každé IT infrastruktury, protože šetří čas správců sítě, náklady a umožňuje optimalizovat využití kapacit zdrojů tak, aby byl hardware co nejlépe využit. S vyvíjejícími se technologiemi, trendem využívání cloudu, softwaru jako služby (SaaS¹ – Software As A Service) vznikají nároky na rychlejší a jednodušší nasazování systému a aplikací interně (tzv. on-premise) nebo v už zmíněném cloudu. Proto se do popředí nad samotnou virtualizaci staví její automatizace a takzvaná infrastruktura jako služba (IaaS² – Infrastructure As A Service). Automatizaci v oblasti virtualizace lze chápat jako soubor činností od nejjednoduššího vytvoření virtuálního stroje pouze na základě požadavku, např. přes samoobslužný webový portál, až po nasazení složitých prostředí, kde se automatizuje i vytváření síťové infrastruktury nebo úložišť, odtud název IaaS. Toho je možné dosáhnout mnoha způsoby a produkty, jako například OpenStack, Terraform, nebo VMware vRealize Automation, který je hlavním tématem diplomové práce v rámci případové studie v reálném nasazení ve firemním prostředí.

Virtualizace už není pouhým prostředkem pro konsolidaci hardwarových serverů, ale primárně platformou pro SaaS a IaaS služby. Tyto služby našly své uplatnění primárně v cloudu, který díky globalizaci nabýval na významu, zpřístupnil služby celosvětově a upozadil standardní nasazení např. webových aplikací pouze na jednom nebo více serverech z vlastního datového centra dané firmy. Tohle by ale nebylo možné, kdyby většina operací nebyla automatizovaná. Ve většině společností za tuto automatizaci odpovídá oddělení DevOps³ (spojení slov software development (Dev) a IT operations (Ops)).

Diplomová práce se zaměřuje na charakteristiku a implementaci produktu VMware vRealize Automation, který byl zvolen jako automatizační nástroj na základě konkrétních požadavků ve vybrané společnosti. Její teoretická část shrnuje produkt samotný, jeho architekturu a možnosti. Přestože vybraná společnost už daný produkt zakoupila, stručně se věnuje alternativním produktům nebo metodám, jak dosáhnout podobného cíle.

¹ SaaS – Software as a Service

² IaaS – Infrastructure as a service

³ DevOps – spojení anglických výrazů Development a (IT) Operations

Praktická část analyzuje konkrétní prostředí před nasazením a uvádí veškeré požadavky, které vznikly, a které bylo nutné dodržet při implementaci. Implementace samotná zahrnuje instalaci, konfiguraci prostředí a nasazení do provozu.

Poslední část shrnuje implementaci po stránce finanční, srovnává původní a finální stav procesů včetně celkových nákladů spojených s vlastnictvím (TCO⁴).

V diplomové práci jsou používány standardní odborné termíny. V případě neexistence rozumného českého ekvivalentu jsou uvedeny v anglickém jazyce – originále, pokud existuje rozumný a používaný český výraz, potom je tento použit přednostně.

⁴ TCO – Total Cost of Ownership

2 Cíl práce a metodika

2.1 Cíl práce

Hlavním cílem práce je implementace produktu VMware vRealize Automation v prostředí vybrané společnosti, za účelem zefektivnění každodenních úkolů vývojářů a testerů pomocí automatizace ve virtuálním prostředí a zjednodušení interních procesů IT oddělení.

Dílčí cíle práce:

- Charakteristika virtualizační technologie
- Popis produktu a možností VMware vRealize Automation a stručný popis alternativ
- Analýza stávajícího stavu v organizaci
- Implementace produktu
- Analýza a komparace finálního a původního stavu včetně celkových nákladů spojených s vlastnictvím (TCO)
- Závěry a zhodnocení

2.2 Metodika

Teoretická část vychází z bakalářské práce (1) autora a charakteristiky virtualizační technologie, studia a analýzy dostupných zdrojů a literatury o produktu. Přestože vybraná společnost už daný produkt zakoupila, stručně se bude taky věnovat alternativním produktům nebo metodám, jak dosáhnout podobného cíle.

Praktická část se zaměřuje na analýzu stávajícího stavu ve vybrané společnosti, popisu procesů stanovených IT oddělením a činností vykonávaným zaměstnanci, které jsou ve spojení s procesy zdlouhavé a je potřeba je zjednodušit. Po kompletizaci informací z analýzy je navržena architektura řešení, která je také graficky zdokumentována. Na to navazuje konkrétní implementace produktu (instalace, konfigurace), příprava prostředí, šablony pro konkrétní operační systémy a služby a nasazení workflow pro konkrétní operace na části infrastruktury jako DHCP nebo DNS. Celá implementace bude zakončena úpravou dílčích procesů a zhodnocením celkových nákladů spojených s vlastnictvím (Total Cost of Ownership – TCO).

Závěru práce patří shrnutí a zhodnocení implementace.

3 Teoretická východiska

3.1 Historie virtualizace

První zmínka o virtualizaci pochází už z roku 1960, kdy vznikl operační systém CP-40, který byl vyvinut pro System/360 Mainframe od firmy IBM. Byl to první operační systém, který měl implementovanou plnou virtualizaci, tj. dokázal poskytnout virtuální prostředí, které podporovalo všechny aspekty a funkce tak, jako kdyby byl operační systém instalován přímo na hostitelský server. OS CP-40 umožňoval spustit až 14 virtuálních strojů ve stejném čase. (2) V roce 1967 byl vyvinut první hypervizor a jeho druhá verze pod označením CP-67 v roce 1968. Tato verze, jež se považuje za předchůdce stávající virtualizace, umožňovala sdílení paměti napříč virtuálními stroji a poskytovala každému uživateli jeho vlastní prostor v paměti. (3) V roce 1997 byla společností Connectix vydána aplikace s názvem Virtual PC pro počítače Apple. Následně byla v roce 2001 vydána první verze pro Windows. Bohužel tato aplikace nezaznamenala větší komerční úspěch.

O dva roky dříve, v roce 1999, byl společností VMware Inc. uveden první produkt, který definoval pojem virtualizace pro veřejnost, a to VMware Workstation, který umožnil spouštět další operační systém nad hostitelským OS. Stal se synonymem pro virtualizaci. Společnost VMware získala mnoho zákazníků z oblasti správy IT a testovacích laboratoří a díky tomuto úspěchu vydala v roce 2001 první verzi svého hypervizoru ESX (dnes ESXi, nebo vSphere). Tím, že technologie virtualizace začala nabírat hodně rychlý vývoj, začali se o ni zajímat i další hráči na trhu. Jedním z nich byla společnost Microsoft, která v roce 2003 oznámila a převzala software Virtual PC od společnosti Connectix a začala vydávat její nové verze pod označením Microsoft Virtual PC 2004. Ve stejném roce, o něco málo později, byla vydána také první veřejná verze softwaru Xen. V dnešní době patří společnost VMware Inc. mezi průkopníky a je jedničkou na trhu s virtualizačními technologiemi. Její náskok v technologii virtualizace se snaží doběhnout jak Microsoft, tak i Red Hat nebo Citrix. (1)

3.2 Současnost virtualizace

Aktuální vývoj virtualizace se přesouvá od otázky „jestli virtualizovat“ spíše k otázce „co všechno můžeme virtualizovat?“. Trend rozšiřování privátního a veřejného cloudu má za následek, že je potřeba virtualizovat ne jenom serverový hardware, ale i další infrastrukturní části, jako je síťová infrastruktura (SDN – Software-defined networking) a datová úložiště (SDS – Software-defined Storage), oboje zahrnuto pod společným názvem Infrastruktura jako služba (IaaS). Všechny tyto technologie umožňují přechod k takzvané hyperkonvergované infrastruktuře. Základním stavebním kamenem celé takové infrastruktury je server, který díky propracované, integrované a odzkoušené kombinaci hardwarových a softwarových komponent, zajišťuje prostředí pro provoz virtuálních serverů, i pro uložení jejich dat. Rozšiřitelnost takové infrastruktury je snadná – pouze se přidá další server. Pokud je nutný výpočetní výkon, bude to server s výkonnými CPU. Pokud je naopak potřebná disková kapacita, server bude obsahovat kapacitně velké lokální datové úložiště. (4)

Software jako služba (SaaS) sice není přímo virtualizací, ale ve spojení se vším výše uvedeným, se začíná mluvit o souhrnném pojmu „cokoliv jako služba“ (XaaS – Anything As a Service). Tady se začínají objevovat prvky automatizace a kontejnerů⁵. Do popředí se dostávají technologie jako Kubernetes, Terraform a Docker, které mají největší uplatnění v cloudu.

3.3 Historie a současnost VMware vRealize Automation

Historie VMware vRealize Automation sahá až do roku 2005, kdy byl ve společnosti Credit Suisse's Global Research and Development Group vyvinut původní software, aby vyřešil provozní a správní výzvy při nasazování virtualizace. Poté co bylo v roce 2008 nasazení virtualizace kompletní a software byl používán pro správu tisíců virtuálních strojů, se společnost Credit Suisse rozhodla vytvořit společnost založenou na této technologii s názvem DynamicOps.

Virtualizace operací je základní technologií cloudového portfolia DynamicOps. Virtualizace operací je abstraktní vrstvou mezi několika systémy řízení, které formují

⁵ Kontejner – struktura, ve které je zabalen zdrojový kód nebo jiný software a všechny jeho závislosti

cloudovou infrastrukturu a jejich odběrateli. To umožňuje zaměstnancům IT oddělení spravovat vrstvy níže (např. síť, úložiště apod.) tak, aby vrstvy nad nimi nevěděli, jak nebo proč.

V červenci 2012 byla společnost DynamicOps (5) koupená společností VMware a tento produkt byl přejmenován na vCenter Automation Center (vCAC). S příchodem verze 6.2 pak následovalo další přejmenování a začlenění do rodiny vRealize pod názvem vRealize Automation (vRA) tak, aby byla sladěna jejich nová strategie.

Je důležité vědět, že vRA jednoduše umožní zákazníkům s různými znalostmi využít přesně ty cloudové zdroje, které jsou jim přidělené. Systém je postaven tak, aby zákazník na konci dne neřešil, kde byl jeho virtuální stroj spuštěn, pokud to bude rychlé, a bude to plnit přesně ten účel, pro který jej potřebují. To znamená, že i když bude v procesu potřeba nějaké schválení, hned poté se veškeré požadavky aplikují automaticky na hypervizorech, nebo cloudových službách. Umožní to každodenní operace bez zásahu administrátorů, kteří už nemusí vytvářet virtuální stroje. Zároveň jsou tyto stroje dle nastavených podmínek automaticky mazány, čímž není potřeba nakupovat hardware tak často. To je hlavním cílem – snadné používání uživateli, šetření nákladů organizace.

V dnešní době spočívá hlavní hodnota vRA v tom, že umožňuje spravovat různé hypervizory a cloudové služby (jako např. vSphere, Red Hat KVM, AWS⁶, Microsoft Hyper-V, Microsoft Azure apod.) tak jako fyzický hardware (přes Cisco UCS, Dell iDRAC a HP iLO karty) pro stavbu spravovaných hybridních cloudů, privátních cloudů, virtuálních desktopů a prostředí platform jako službu (anglicky: platform as a service environments). Jde o rozsáhlou škálu možností a funkcí, které budou i nadále přibývat a poskytovat větší přidanou hodnotu pro uživatele i administrátory.

3.4 Definice virtualizace

Hned v úvodu je potřeba zdůraznit, že virtualizace není simulace ani emulace, protože když virtualizujeme, tak nic nesimulujeme, ani nevytváříme žádné fiktivní zdroje. Mnoho lidí si tyto dva termíny plete. Pojem virtualizace vztahený k výpočetní technice se dá jednoduše vysvětlit jako sdílení jednoho zdroje více systémy ve stejném čase. Záměrně je zmíněno slovo „systémy“, protože existuje vícero typů virtualizací. (6) (7) (40)

⁶ AWS – Amazon Web Services

Patří mezi ně:

- virtualizace serverů (více operačních systémů na jednom hardware),
- sítě (nahrazení klasické sítě softwarovou vrstvou, která urychluje síťové operace probíhající mezi virtuálními stroji tím, že probíhají v paměti RAM),
- pracovní plochy (minimalizace počtu PC a běžících desktopových operačních systémů využitím tzv. thin-client systémů, které se připojují vzdáleně na terminálové služby),
- aplikace (aplikace běžící ve virtuálním stroji, kdy se virtualizují jen zdroje, ne celý OS, například Java Virtual Machine, Net Framework),
- úložiště (součást softwarově definovaných úložišť, která musí nabízet vylepšení výkonu a efektivní využití místa, aniž by bylo potřeba zakoupit další hardware úložiště).

Při nasazení virtualizovaných serverů se z pohledu uživatele nic nemění, protože se chovají stejně, jako fyzické servery. Každý virtuální stroj si při spuštění alokuje svoje vlastní zdroje a má svůj vlastní virtuální hardware, což umožňuje používat různé verze operačních systémů. Komunikace virtualizovaných strojů je poskytována standardní síťovou vrstvou, např. TCP/IP, kterou umožňuje daný operační systém.

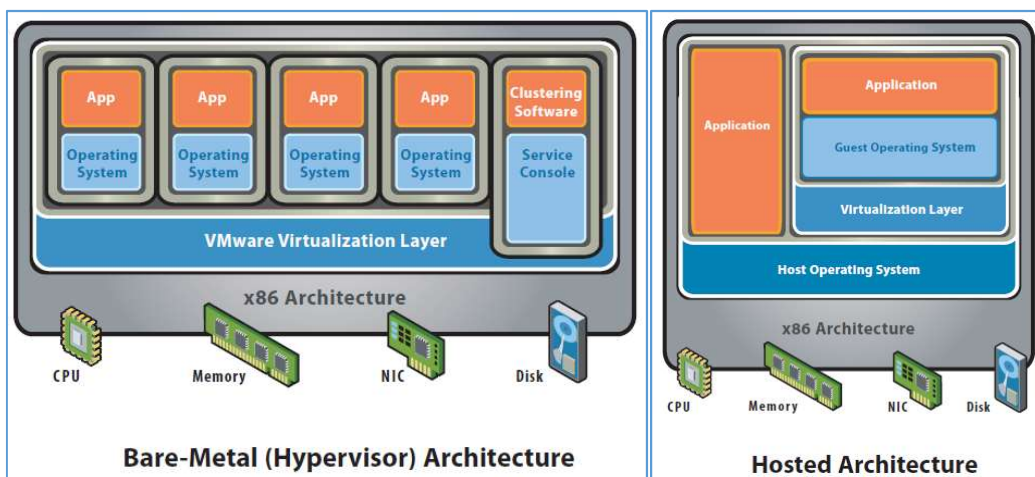
Před samotnou virtualizací je nutné si zvolit její typ, resp. typ softwaru – hypervizoru, který použijeme. Typy hypervizorů popsal Robert. P. Goldberg již v roce 1973 a do dnešní doby se nezměnily. (8)

Typy hypervizorů:

- Typ 1 – Nativní (v anglickém originálu – Bare Metal – Goldberg definoval jako Bare Machine) (9)
- Typ 2 – Hostovaný (v anglickém originálu – Hosted) (9)

Typ 1 (obr. 1) je hypervizor, který je spuštěn přímo na hostitelském hardware a zároveň tento hardware kontroluje a spravuje hostované virtuální stroje. Virtuální stroje jsou spuštěné v další vrstvě nad hypervizorem.

Typ 2 (obr. 2) je hypervizor, který běží v rámci běžného prostředí operačního systému. Je na něm závislý a nedokáže plně kontrolovat hardware, ale jen tolik, kolik mu umožní OS.



Obrázek 1 - Typ 1 (vlevo) (zdroj: www.vmware.com)

Obrázek 2 - Typ 2 (vpravo) (zdroj: www.vmware.com)

3.4.1 Sumarizace výhod a nevýhod virtualizace

Postupným vývojem této technologie bylo uvedeno několik produktů zaměřených na serverovou virtualizaci, které umožňovaly na jednom serveru spustit více operačních systémů současně, bez nutnosti instalovat hostitelský operační systém proti klasickému scénáři, kdy na jednom serveru je možné spustit jen jeden operační systém.

Z toho vyplývá několik výhod:

- úspora kapitálových a provozních nákladů,
- až o 80 % lepší využití zdrojů v závislosti na použité technologii,
- konsolidace serverů,
- dynamická a jednoduchá alokace zdrojů v případě okamžité potřeby,
- jednoduché testování a možnost izolace jednotlivých prostředí,
- vysoká dostupnost aplikací.

Nevýhody se hledají hůře, ale jelikož žádná technologie není dokonalá, jsou jimi následující:

- prvotní náklady jsou vyšší jak na hardware, tak na software pro virtualizaci,
- ne všechno je vhodné virtualizovat (viz kapitola 3.4.2),
- vyšší závislost na funkcionalitě hardware – chyba postihne všechny virtuální stroje na něm běžící, tzv. „Single point of failure“,
- chyby v software pro virtualizaci – daný operační systém nebo aplikace se může ve virtualizovaném prostředí chovat jinak než na fyzickém hardware,
- částečné vyhrazení zdrojů pro samotný hypervisor.

3.4.2 Příklady vhodné a nevhodné virtualizace

Virtualizace je v dnešní době horké téma a využívaná technologie ve mnoha firmách i u domácích uživatelů. Ne vždy je možné ale virtualizovat. Zde jsou uvedeny příklady, co je vhodné a mělo by se virtualizovat, a co je naopak nevhodné k virtualizaci.

Vhodné použití je tam, kde se dá ušetřit na HW zdrojích (CPU, RAM), protože je daný systém nedokáže využít na 100 %. Například:

- Webové servery,
- Mailové servery,
- Doménové kontroléry,
- CRM / ERP aplikace,
- DNS servery,
- Souborové servery,
- Proxy servery.

Naopak existují služby, které je nevhodné virtualizovat, i když spousta výrobců hypervizorů tvrdí opak, ale praxe říká něco jiného:

- Systémy, které potřebují extrémní výkon, a využívají CPU, RAM a diskové I/O operace na maximum. Příkladem jsou databázové servery, nebo renderovací grafické servery,
- Cokoli, co ke své funkčnosti vyžaduje specifický HW, jako jsou např. kontrolní systémy, měřicí systémy, které jsou připojené přes USB, RS-232, PCI. Některé hypervizory umožňují přímý přístup k tomuto HW, avšak nedoporučuje se to,
- Aplikace, které mají ve smluvních podmínkách zákaz použití virtualizace,
- Systémy vyžadující fyzické zabezpečení, nebo to vyžadují bezpečnostní regulace, protože hypervizor zvyšuje bezpečnostní riziko.

3.4.3 Základní pojmy a terminologie

Z předchozí části je zřejmé, že pro virtualizaci je potřeba software, který spravuje, obsluhuje a poskytuje fyzické zdroje virtuálním strojům. Tento software tvoří poměrně malou mezivrstvu mezi fyzickým a virtuálním hardwarem a jmenuje se hypervizor. Jeho hlavní úlohou je umožnit současný běh několik virtuálních strojů současně, přidělovat procesorový výkon, paměť, umožnit přístup k diskům, síti a dalšímu hardware jako Host

Bus Adapter (HBA) karty, GPU karty atd. Hypervizorů je několik typů. Typ 1 je nativní hypervizor, běžící přímo na hostitelském hardwaru. Tento typ je využíván produkty VMware vSphere, Citrix XenServer, Microsoft Hyper-V Server nebo KVM. Typ 2 je hypervizor instalovaný na operační systém a až nad ním běží jednotlivé virtuální stroje. Rozdíl mezi oběma typy je zásadní, protože ve druhém případě je možné na daném hostitelském operačním systému provozovat aplikace a služby, které také využívají zdroje fyzického stroje. Mezi tento typ patří, například VMware Workstation, VMware Fusion (virtualizace pro Mac OS), Parallels (virtualizace pro Mac OS), Microsoft Hyper-V Server for Windows desktop / Windows Server, Oracle VirtualBox. Několikrát zmiňovaný virtuální stroj je samotný virtuální hardware běžící nad hypervizorem.

Dalším hodně zmiňovaným termínem v oblasti virtualizace je virtuální switch, nebo vSwitch. Jedná se o softwarovou náhradu hardwarového switchu. Každá z technologií obsahuje jeho určitou variantu a každá ji má implementovanou jinak, ale činnost je velmi podobná. Virtuální síťové karty se místo připojování kabelem do switchu připojí k virtuálnímu switchi a provoz mezi jednotlivými virtuálními stroji tak neprobíhá na linkové vrstvě síťové karty serveru, ale v paměti RAM. Tím je možné dosáhnout většího výkonu a přenosových rychlostí – běžně 10Gbps.

Hodně využívanou funkcionalitou, kterou nelze bez virtualizace prakticky udělat, jsou snapshoty. Zjednodušeně řečeno, uložení aktuálního stavu virtuálního stroje. Při vytvoření snapshotu se dějí dvě věci. První je zamknutí stávajícího virtuálního disku pro zápis a vytvoření nového rozdílového disku. Druhá je obdobou první, ale pro RAM. Hypervizor stávající obsazenou RAM zamkne pro zápis, začne zapisovat rozdíly mimo tuto oblast a zamknutou část uloží do souboru. Následně spojí původní a rozdílovou část.

U nativních hypervizorů se ve firemním prostředí pracuje s pojmem cluster. (10) Jde o sloučení několika hypervizorů do skupin, které pracují společně tak, že navenek působí jako jeden hypervizor. V prostředí virtualizace se cluster využívá hlavně pro zajištění vysoké dostupnosti aplikací (z anglického high availability).

3.5 Automatizace ve virtualizaci

Trendem (11) (12) v současné době je v organizacích trendem opouštět model tradičních serverů, adoptovat virtualizaci a používat ji ve více kritických aplikacích, čímž se ale infrastruktura zesložituje. Datová centra se začínají přestavovat na privátní cloud, bohužel s větší složitostí se lehkost a rychlost proměny snižuje. Řešením k překonání těchto limitací by měla být automatizace. Automatizaci v oblasti virtualizace lze chápat jako soubor činností od nejjednoduššího vytvoření virtuálního stroje pouze na základě požadavku, např. přes samoobslužný webový portál, až po nasazení složitých prostředí, v nichž se automatizuje i vytváření síťové infrastruktury nebo úložišť (Infrastructure As A Service – IaaS), tedy ve všeobecnosti jde o automatizaci IT procesů. Díky automatizaci IT procesů založených na firemních zásadách (politikách) mohou IT týmy/oddělení využít předem definovaných workflow a efektivněji využít jejich administrativní možnosti, což jim umožňuje rychleji reagovat a vyřešit servisní požadavky.

Automatizace IT procesů může zahrnovat například:

- Plánování kapacit
- Automatické poskytování zdrojů (Automated provisioning)
- Správu životního cyklu VM
- Samo obslužnou automatizaci (Self-service automation)
- Správa změn a konfigurací
- Optimalizace zdrojů
- Automatizace řízená zásadami (Policy-based)

To je možné dosáhnout mnoha způsoby a produkty.

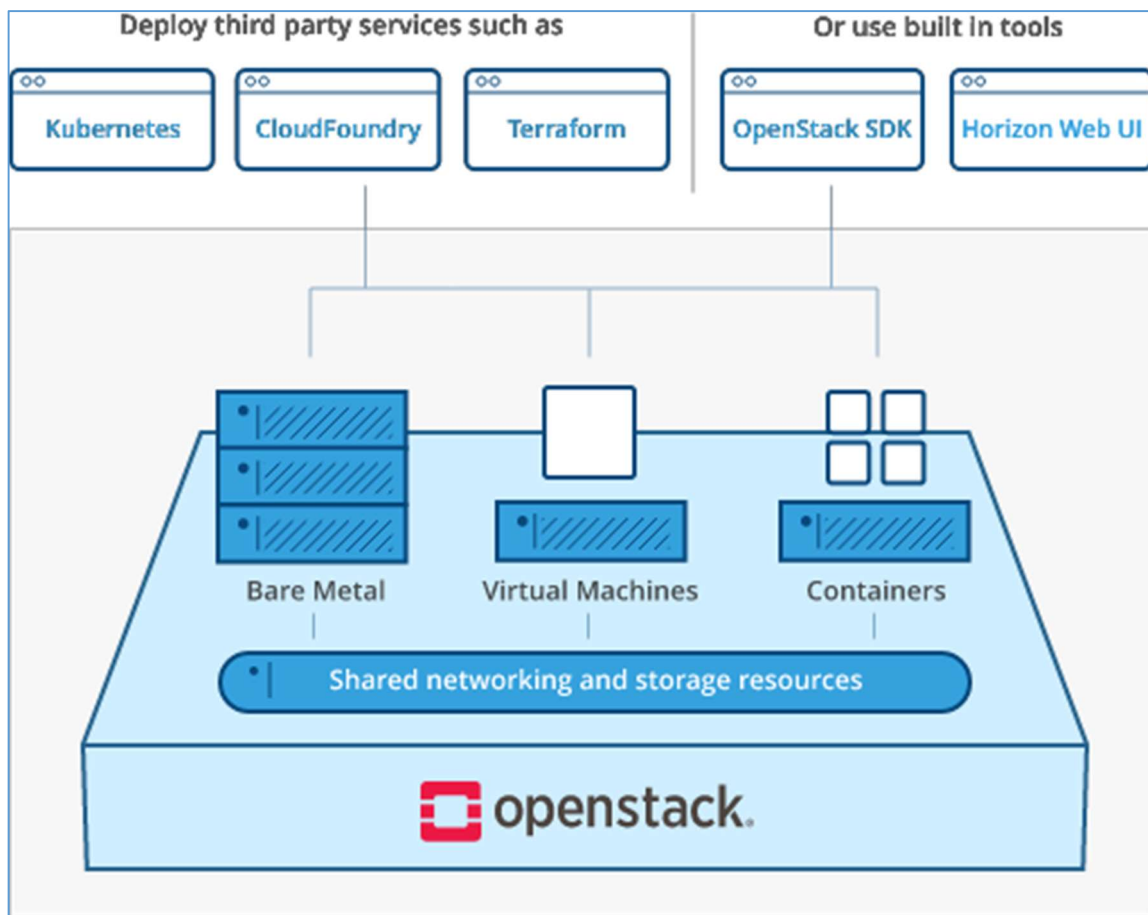
3.5.1 Produkty pro automatizaci IT procesů

Není možné uvést všechny existující produkty, proto byli vybrány tři nejdůležitější z pohledu současného IT, z toho jeden bude charakterizován podrobně. Vybrané produkty jsou zahrnuty ve vícekritériální analýze v rámci případové studie. (39)

3.5.1.1 OpenStack

OpenStack (13) je open source platformou, která využívá virtuálních a HW zdrojů pro výstavbu a správu privátních a veřejných cloud prostředí, většinou je nasazena jako IaaS infrastruktura. Nástroje tvořící platformu OpenStack, nazývané „projekty“, se zabývají

základními cloudovými službami výpočetního výkonu, sítě, úložišť, identit a služeb obrazů operačních systémů (image services). OpenStack využívá konzistentní sadu API, aby abstrahovalo zdroje, jako CPU a RAM, a také hypervizory od různých výrobců. Zároveň nabízí API pro administrátory a vývojáře, kteří mohou automatizovat a integrovat OpenStack do jiných systémů.



Obrázek 3 - High-level schéma OpenStack (zdroj: <https://www.openstack.org/software/>)

Jak bylo výše zmíněno, architektura OpenStack se skládá z mnoha open source projektů. Tyto projekty se používají k nastavení „undercloud“ a „overcloud“. Undercloud obsahuje základní komponenty, které musí administrátoři systému nastavit. Overcloud znamená správu prostředí koncových uživatelů.

K dispozici je šest stabilních základních služeb, které zpracovávají výpočetní výkon, síť, úložiště, identitu a obrazy operačních systémů, zatímco více než tucet dalších volitelných služeb se liší vývojovou fází. Těchto šest základních služeb tvoří infrastrukturu, která umožňuje ostatním projektům poskytovat dashboardy, orchestraci, nasazení bare-metal serverů, komunikaci, kontejnery a správu.

Základní komponenty: (14)

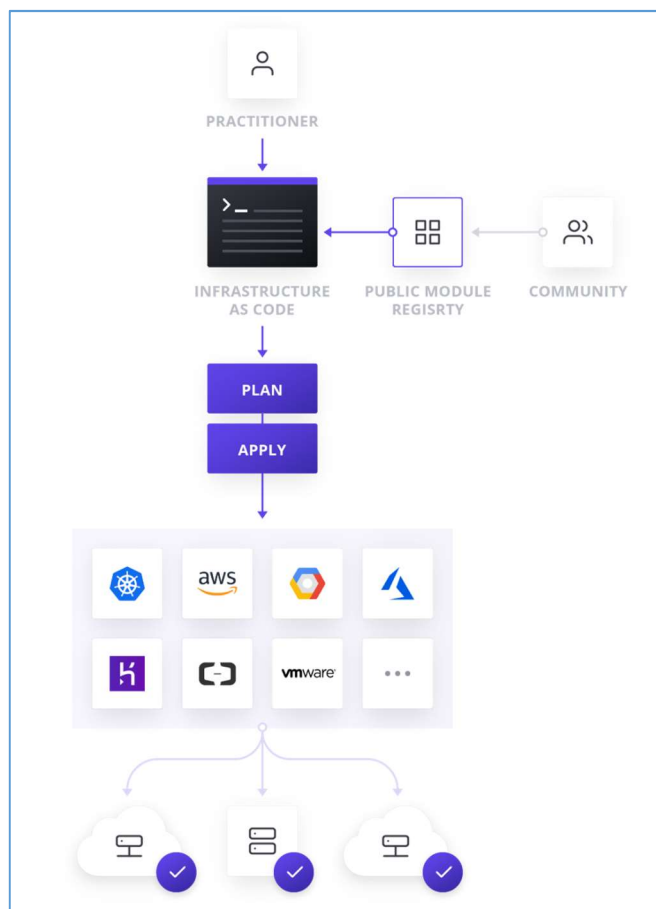
- **Nova** – projekt, který poskytuje způsob, jak zajistit výpočetní instance (virtuální servery). Nova podporuje vytváření virtuálních strojů, fyzických (baremetal) serverů (pomocí projektu Ironic) a má omezenou podporu pro systémové kontejnery. Funguje jako sada démonů na existujících linuxových serverech, které tuto službu poskytují.
- **Neutron** – projekt, který poskytuje „síťové připojení jako službu“ (NaaS) mezi zařízeními rozhraní (např. vNIC) spravovanými jinými službami (např. Nova). Implementuje rozhraní OpenStack Networking API.
- **Swift** – vysoce dostupný, distribuovaný, případně konzistentní objekt/blob úložiště. Organizace jej mohou používat efektivně, bezpečně a levně k ukládání dat.
- **Cinder** – je služba poskytující blokové úložiště (Block Storage) jako svazky (volumes) virtuálním strojům vytvořeným ve službě Nova, fyzickým serverům spravovaných službou Ironic, kontejnerům a dalším.
- **Keystone** – poskytuje autentizaci klientům rozhraní API, service discovery a distribuovanou multi-tenant autorizaci
- **Glance** – služba, ve které mohou uživatelé nahrávat obrazy systémů, které mají být použity s jinými službami.

3.5.1.2 Terraform

Terraform (15) je nástroj pro bezpečné a efektivní budování, provádění změn a verzování infrastruktury. Terraform může spravovat stávající a populární poskytovatele služeb, i vlastní interní řešení.

Konfigurační soubory popisují komponenty potřebné pro spuštění jedné aplikace, nebo celého datového centra. Terraform generuje akční plán popisující, co udělá pro dosažení požadovaného stavu, a poté ho provede, aby vybudoval popsanou infrastrukturu. Se změnou konfigurace je Terraform schopen určit, co se změnilo, a vytvořit přírůstkové akční plány, které lze použít.

Infrastruktura, kterou Terraform může spravovat, zahrnuje komponenty na nízké úrovni, jako jsou výpočetní instance, úložiště a síť, jakož i komponenty na vysoké úrovni, jako jsou položky DNS, funkce Software as a Service (SaaS) atd.



Obrázek 4 - High-level schéma Terraform (zdroj: <https://www.terraform.io/>)

Klíčové vlastnosti:

- **Infrastruktura jako kód** – Infrastruktura je popsána pomocí konfigurační syntaxe na vysoké úrovni. To umožňuje, aby byl plán datového centra verzován a bylo s ním zacházeno jako s jakýmkoli jiným kódem. Kromě toho lze infrastrukturu sdílet a znovu použít.
- **Akční plány** – Terraform má stav „plánování“, ve kterém vygeneruje akční plán. Akční plán ukazuje, co Terraform vykoná. To umožňuje provádět kontroly změn dříve, než Terraform začne manipulovat s infrastrukturou.
- **Graf zdrojů** – Terraform vytváří graf všech zdrojů a paralelizuje tvorbu a úpravu jakýchkoli nezávislých zdrojů. Z tohoto důvodu Terraform buduje infrastrukturu co nejefektivněji a operátoři získají přehled o závislostech v infrastruktuře.
- **Automatizace změn** – Komplexní změny lze aplikovat na infrastrukturu s minimální lidskou interakcí. S výše uvedeným akčním plánem a grafem zdrojů uživatel přesně

ví, co Terraform změní a v jakém pořadí, čímž je možné vyhnout se mnoha lidským chybám.

3.5.1.3 VMware vRealize Automation

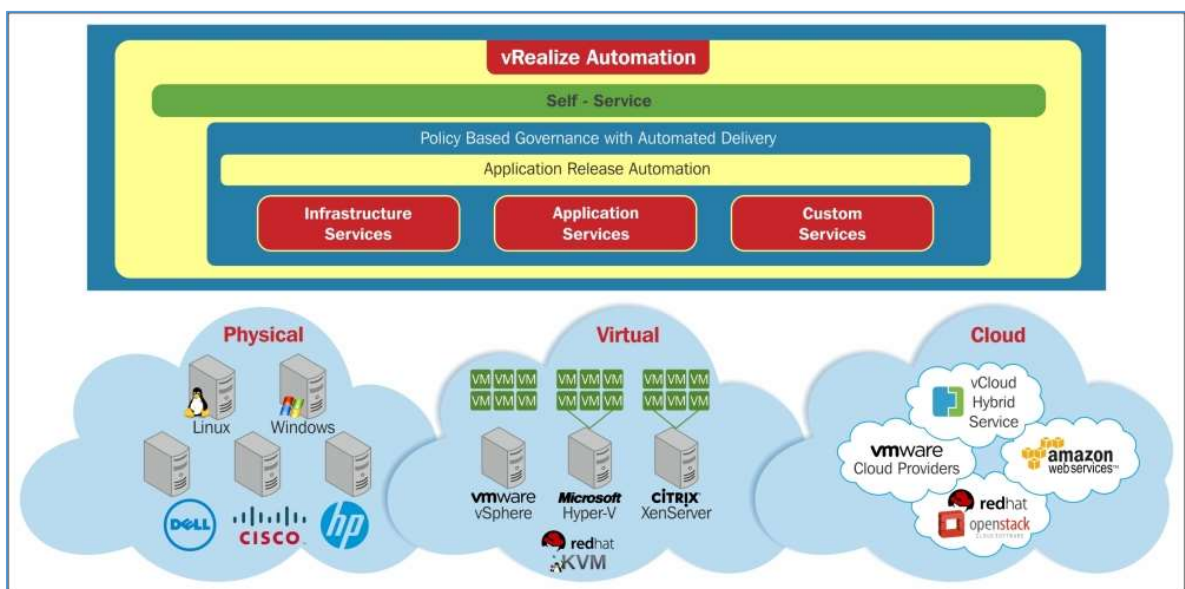
vRA (16) by se dalo zjednodušeně definovat jako samoobslužný portál, který poskytuje „všechno jako službu“ - XaaS (Everything as a service). Přesnější definice by se dala formulovat tak, že vRA je kompletní platforma pro správu cloudu (CMP – Cloud Management Platform). vRA je detailně charakterizováno v následující kapitole 3.6.

3.6 VMware vRealize Automation (vRA)

Táto časť charakterizuje software, architekturu jeho hlavných častí a možnosti, ktoré nabízí.

3.6.1 vRA obecně

vRA lze zjednodušeně definovat jako samoobslužný portál, který poskytuje „všechno jako službu“ - XaaS (Everything as a service). Přesnější definice by zněla tak, že vRA je kompletní platforma pro správu cloudu (CMP – Cloud Management Platform). Definice XaaS zahrnuje mimo jiné IaaS (Infrastructure as a Service), DBaaS (Database as a Service) a také ITaaS (IT as a Service). Další charakteristikou vRA je řízení procesů na základě politik a možnost zadávání uživatelských požadavků skrze jeden webový portál. Toto umožňuje flexibilitu tím, že umožňuje uživatelům poskytnout infrastrukturu a aplikace od různých dodavatelů. Zároveň vkládá do firemních procesů agilnost automatickým zpracováním uživatelských požadavků, na které za jiných, resp. normálních okolností závisí a čekají na jejich systémové administrátory, než je zpracují. Uživatelé už nemusí čekat na to, než jejich IT oddělení postaví a zprovozní fyzické nebo virtuální servery, nainstaluje na ně software a zabezpečí dle politik dodržování předpisů.



Obrázek 5 - Přehled vRA - zdroj: Mastering vRealize Automation 6.2

3.6.2 Benefit vRA

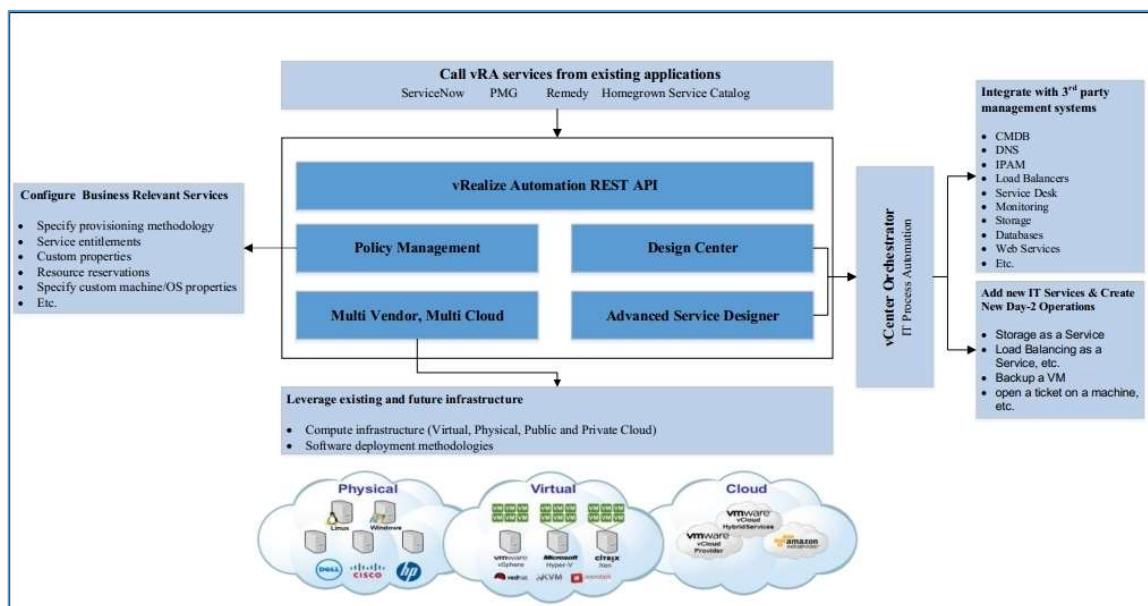
Mezi klíčové benefity platformy vRA patří: (17)

- **Agilnost** – umožňuje automatizovat proces doručování IT služeb (infrastruktura, kontejnery, aplikace a další jiné IT služby) tak, aby rychle reagovaly na obchodní potřeby
- **Rozšiřitelnost** – schopnost jednoduše integrovat nástroje třetích stran skrze celý IT ekosystém a ochránit investice do stávajících a budoucích technologií
- **Kontrolovatelnost** – Začlenit politiky založené na řízení do IT služeb v prostředí hybridního cloudu a zajistit dodržování politik, výkon a finanční výsledky
- **Platforma pro vývojáře** – samoobslužný portál dává možnost vývojářům rychle získat stavební bloky pro své projekty
- **Kompletní správa životního cyklu** – dosažení optimálního řízení vytížení od počátečního nasazení, průběžného balancování až po expiraci a navrácení použitých zdrojů zpátky pro použití v dalším nasazení
- **Bezpečnost** – veškeré akce, které jsou prováděné uživateli v rámci samoobslužného portálu nebo API jsou standardně ve výchozím nastavení logovány a dedikovaný manažer dostává emaily v případě, že je vyžadováno schválení konkrétní akce

3.6.3 Běžné případy použití vRA

vRA nabízí nespočetně možností při implementaci nebo integraci s produkty třetích stran buď vestavěnými nástroji, nebo zásuvnými moduly. Mezi nejvíce používané scénáře můžeme zařadit:

- Vytvoření katalogu standardních operačních systému nebo vzorů virtuálních strojů, které mohou být použity na jedno kliknutí
- Nabízet jiné služby nad samotnou infrastrukturom, například PaaS, XaaS
- Integrovat se s CMDB nebo ITSM nástroji pro sledování aktivit o virtuálních nebo fyzických strojích v době, kdy jsou nasazovány
- Integrace se systémem IPAM pro správu IP adres
- Nasazování hybridních cloudů



Obrázek 6 – Možnosti integrací vRA (zdroj: Rajendran, Sriram - Learning VMware vRealize Automation - Packt 2016)

3.6.4 Dostupné edice a licencování

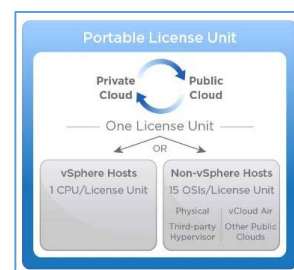
Dostupné edice a licencování lze rozdělit do dvou kategorií podle toho, jestli je produkt nakupován samostatně, nebo v rámci sady vRealize Suite.

	Standard	Advanced	Enterprise
vRealize Automation (samostatný produkt)	Není dostupné	Ano / licencování OSI	Ano / licencování OSI
vRealize Suite (sada)	Ano / licencování PLU	Ano / licencování PLU	Ano / licencování PLU

Tabulka 1 - dostupné edice VMware vRA

Typy licencí: (18) (19)

- OSI – Dle počtu instancí operačních systémů (Operating System Instances) – Licencuje se počet nasazených a spravovaných strojů, není omezena počtem hypervizorů nebo cloudových prostředí
- PLU – Přenosná licence (Portable License Unit) – Licencuje procesory CPU na konkrétním hypervizoru v případě vSphere, nebo obsahuje 15 OSI, pokud se použije jiný typ hostitelského systému (např. AWS, fyzický server, ...)



Obrázek 7 - Portable License Unit - Zdroj:

<https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/vcloud-suite-pricing-packaging-whitepaper.pdf>

Dostupné edice (původní nepřeložený text):

Feature	Advanced	Enterprise
Automated self-service with unified service catalog and API functions	✓	✓
Multi-vendor hypervisor, physical endpoint and public cloud support	✓	✓
Governance and compliance policies	✓	✓
Resource lifecycle management	✓	✓
Resource right-sizing and reclamation	✓	✓
Extensibility options for ecosystem partner tools integration	✓	✓
Marketplace of vendor and partner provided blueprints, plug-ins and other contents	✓	✓
Automatic service catalog pricing (integrated with vRealize Business for Cloud)	✓	✓
Infrastructure as a Service (IaaS)	✓	✓
Anything as a Service (XaaS)	✓	✓
Application stack provisioning and management	✗	✓
Out-of-the-box integration with Configuration Management tools	✗	✓
Container management	✗	✓

Tabulka 2 - Porovnání funkcionality v edicích – Zdroj:

<https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/products/vrealize/vmware-whats-new-vrealize-automation.pdf>

Obě edice nabízejí téměř totožnou funkcionality s tím rozdílem, že verze Enterprise je více zaměřena na nasazování aplikací (Puppet, Chef, SaltStack), kontejnerů a integrování s nástroji pro správu konfigurací.

3.7 Architektura vRA 7.x

Architekturu platformy vRA je nutné charakterizovat ze dvou pohledů. Prvním jsou základní stavební kameny architektury a druhým jsou samotné typy nasazení a specifikací možností distribuce jednotlivých komponent.

Základními kameny se myslí samotné servery, a to *vRealize Automation Appliance* a *IaaS Server*. V rámci verze 7.x byl kladen důraz na zlepšení uživatelského zážitku (user experience – UX), počínaje tou nekritičtější částí – nasazením řešení, a zadruhé jeho konfigurací. Po splnění příslibu efektivnějšího nasazování přináší vRA 7.x významný skok v rámci UX a jeho průvodcem pro zcela automatizovanou instalaci celé platformy a úvodní konfigurace. To vše sebou přineslo i výrazné zjednodušení architektury nasazení a počtu nutných serverů pro běh celé platformy. Ve srovnání s verzí 6.x, která byla nasazená ve vysoce dostupné variantě (high-available – HA), bylo nutné mít minimálně 8 serverů pouze pro chod základních služeb (nepočítaje IaaS/komponentu ve Windows a servery externích aplikací). V porovnání s nasazením vRA 7.x, se vše vešlo pouze do jednoho páru serverů základních služeb. Tento pár poskytuje Framework pro služby vRA, Identity Manager (SSO/vIDM), vPostgres databázi, vRealize Orchestrator (vRO) a RabbitMQ. To vše v módu cluster a konfigurovatelné a schované za jednou VIP (virtuální IP adresa) load

balancerů a jedním SSL certifikátem. I když se od verze 6.x množství služeb přesunulo do vRA Appliance, komponenta IaaS (která je programována v .Net) stále zůstává jako externí služba a je na ní potřeba další servery. To by se mělo změnit v už oznámené verzi vRA 8 a Windows Server by neměl být potřeba.

3.7.1 Základní služby vRealize Automation Appliance

Jak již bylo dříve zmíněno, vRA Appliance (dříve také Cloud Automation Framework for Extensibility – CAFÉ) poskytuje Framework pro služby vRA, Identity Manager (SSO/vIDM), vPostgres databázi, vRealize Orchestrator (vRO) a RabbitMQ.

vRA Appliance je předem připravený systém ve formě šablony, Virtual Appliance, který se jednoduše importuje do prostředí VMware vSphere. Je postavena na operačním systému SUSE Linux Enterprise Server 11 (SLES). Primárním určením celé vRA Appliance je zaměření na byznys logiku a workflow, přičemž poskytuje samoobslužný portál.

VMware Identity Manager (vIDM)

vIDM představuje obrovské zlepšení na poli autentizace v porovnání s předešlou verzí. vIDM je ve verzi vRA 7.x novinkou a vestavěnou součástí vRA Appliance. Kromě zlepšení autorizace a zjednodušení komplexity nasazení taky výrazně zjednoduší proces údržby a upgradů. (20)

Funkcionalita poskytována vIDM:

- Ověřování (uživatelské jméno/heslo, Kerberos, SAML, Smart Card / Certifikáty, RSA SecurID, RADIUS, RSA Adaptive Authentication)
- Multifaktorové ověřování
- Auditování autentizace
- Vylepšení škálovatelnosti
- Podpora vysoké dostupnosti (HA)
- Podpora multi-tenant prostředí (více domén v rámci jednoho tenantu / jedna doména pro více tenantů)
- Podpora lokálních účtů
- Synchronizace s Active Directory
- Podpora přizpůsobení uživatelského webového rozhraní (branding) při přihlášení pro každý tenant

vPostgres

Pro ukládání veškeré konfigurace, politik, definic a katalogů slouží interní vestavěná databáze vPostgres. Databáze je připravena pro vysokou dostupnost a umožňuje replikace mezi členy vRA clusteru. (21)

RabbitMQ

Služba RabbitMQ je typu message broker a využívá protokol Advanced Message Queuing Protocol (AMQP). Pro správný chod celé vRA Appliance je nutné, aby se služba startovala ještě před samotnou službou „vcac-server“, v opačném případě by docházelo k pádům a nefunkčnosti některých komponent platformy. Message broker je možno charakterizovat jako jednoduchou poštovní službu mezi službami. I když většina služeb v rámci vRA používá pro komunikaci REST API, RabbitMQ obsluhuje následovně: (22)

- Pracovní fronty (Work queues)
- Zásobníkové a dávkové operace (Buffer and batch operations)
- Doručení požadavků (Request offloading)
- Distribuce zátěže (Workload distribution)

Pro kontrolu, jestli služba RabbitMQ běží v pořádku a je připojena, slouží příkaz

```
rabbitmqctl list_queues
```

Výstup by se měl podobat příkladu níže.

```
[master] vmware-vra-01:~ # rabbitmqctl list_queues
Listing queues
vmware.vcac.core.ipam-service.workflowCompleted 0
vmware.vcac.core.ipam-service.requestCompleted 0
vmware.vcac.core.composition-service.error      0
vmware.vcac.core.approval-service.approvalProcessor 0
vmware.vcac.core.catalog-service.error 0
vmware.vcac.core.network-service.requestCompleted 0
...
```


vRealize Orchestrator (vRO)

vRO je nástroj pro automatizaci prostředí a orchestraci byznys procesů, a to nejen v rámci vRA, ale je možné ho nasadit i samostatně a integrovat přímo s vSphere vCenter Server. Vzhledem k tomu, že vRO je rozsáhlým nástrojem, který lze charakterizovat a rozebrat v samostatné knize a zároveň je to stěžejní část vlastní práce, je mu níže věnována samostatná část, která více přibližuje jeho funkcionalitu.

3.7.2 Služby Infrastructure as a Service (IaaS) serveru

IaaS server nemá přesnou definici, veškerá dokumentace a literatura popisuje pouze jeho komponenty a jejich definice, nebo účel. V jednoduchosti lze ale říct, že IaaS server je motor pohánějící všechny akce spojené s akcemi uživatelů ve webovém portálu, REST API a akcemi spojenými s privátním nebo veřejným cloud prostředím. IaaS se instaluje na Windows Server a proto je někdy označován jako „legacy“ komponenta. IaaS server zahrnuje několik komponent, které můžou, ale nemusí být instalované na jeden server současně. Je zde nativně podporováno distribuované nasazení. (23) (24)

3.7.2.1 Komponenty IaaS

Web Server

Webový server, který využívá *Internet Information Server (IIS)* a hostuje Model Manager a zároveň portál pro administraci infrastruktury, modelování šablon a vytváření katalogu a prostředí pro samotné koncové uživatele. I když část uživatelského portálu běží na samotné vRA Appliance (linux), mandatorní část portálu je nadále generovaná na Windows Serveru. Web běžící na vRA Appliance zobrazuje obsah z IaaS pomocí frames (rámů). Tato komponenta komunikuje se službou Manager Service, která poskytuje aktualizace z Distributed Execution Manager (DEM), SQL server databáze a agentů. (23) (25)

Model Manager

Model Manager je srdce IaaS serveru. Běží v rámci IIS a jeho základním úkolem je zapouzdřit data ze SQL Server databáze jako modely a poskytnout k nim přístup skrze webové služby. Navíc jsou v rámci Model Manageru uloženy veškeré informace, logika a jiné artefakty, které jsou potřebné pro spuštění workflow na externích systémech (jako

například i samotný vSphere, Microsoft System Center Virtual Machine Manager, apod.). Tyto informace jsou následně využity službami DEM při nasazování. (23) (25)

Data model a rozhraní REST

Jak již bylo zmíněno, veškerá data jsou uložena v SQL Server databázi, webový server, jehož součástí je Model Manager poskytující webovou službu REST pro přístup k těmto datům. Tyto data jsou poskytována jako modely. Interně je rozhraní REST implementováno technologií .Net Framework.

Informace o zabezpečení

Model Manager je úložištěm i pro přístupová práva, jako například kdo je schopen vidět konkrétní data a které akce mohou být použity v průběhu nasazování.

Workflows

V předešlé části se nacházela zmínka o vRO a službě Event Broker. Model Manager s tím přímo souvisí. vRA poskytuje možnost nasadit/poskytnout zdroje na různé platformy – fyzické, virtuální nebo cloudové. Nicméně, v závislosti na platformě a technice nasazování, je potřeba mít v Model Manageru uložena různá vestavěná workflow pro nasazení (např. PXE, SCCM, VM Template). Tyto workflow, jako i vlastní, je možné rozšířit výše zmíněnou službou Event Broker.

Events and triggers (události a spouště)

Neméně důležitou funkcionalitu, související s Workflow a Event Broker, je možnost registrovat vlastní události v rámci datového modelu. Typické použití je například aktualizace CMDB v případě, že uživatel změní vlastnosti vytvořeného stroje skrze webový portál.

Manager Service

Manager Service běží jako služba ve Windows a koordinuje komunikaci mezi Model Manager, databází, agendami, Active Directory a SMTP. Musí být spuštěná pod doménovým účtem s lokálními administrátorskými právy na všech IaaS Windows serverech. Tato služba nemá skoro žádné konfigurační možnosti, ale musí být zajištěno, aby neustále běžela. (23) (25)

Microsoft SQL Server

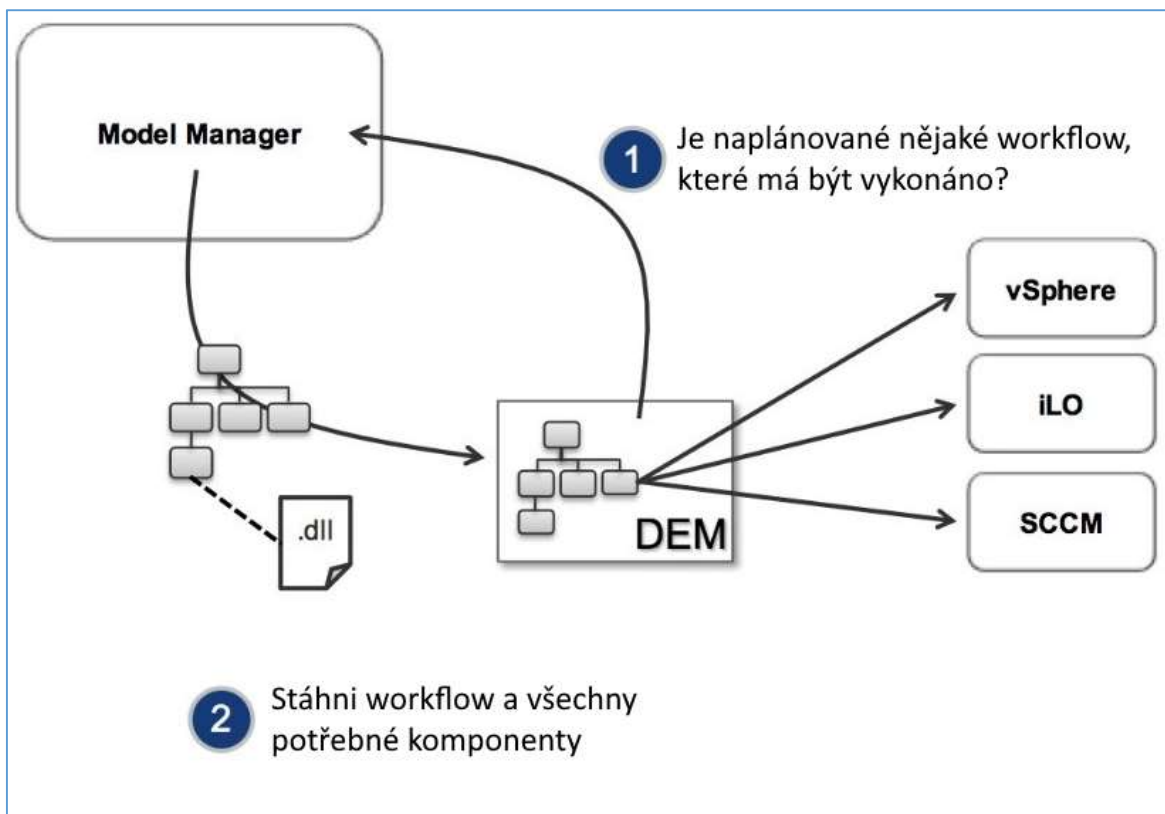
IaaS využívá Microsoft SQL Server databázi pro ukládání veškerých dat, za které je IaaS odpovědný. IaaS ve verzi 7 nepodporuje žádnou jinou databázi. V případě, že se zvažuje nasazení vRA ve vysoce dostupné variantě (high-availability), je potřeba zvážit implementaci Microsoft SQL Server failover cluster.

Distributed Execution Manager (DEM)

Zatímco workflow logika je uložena v Model Manager-u, o její spuštění a chod se starají služby DEM. Odpovědnost DEM je komunikovat se službou Model Manager, sesbírat požadavky na vykonání nějaké akce nebo spuštění workflow a vykonat je. Z technického hlediska rozdělujeme DEM na 2 druhy: (23) (25)

- *DEM Orchestrator* – komunikuje se službou Model Manager-u a plánuje spuštění workflow. DEM Orchestrator pouze monitoruje a plánuje, ale workflow nevykonává. Protože mezi DEM Orchestratorem a Model Managerem probíhá velké množství komunikace a interakcí, je doporučeno ho instalovat „co nejbližší“ nebo na Windows Server, který hostuje Model Manager. V konkrétní chvíli může být aktivní pouze jeden DEM Orchestrator v rámci instalace vRA. Všechny ostatní instance jsou pasivní a jedna z nich se stává aktivní pouze v případě pádu aktuálně aktivní instance.
- *DEM Worker* – je zodpovědný za vykonání workflow. V kontrastu se službou DEM Orchestrator, všechny instance DEM Worker jsou aktivní ve stejný čas, co napomáhá při škálovatelnosti a vysoké dostupnosti. Protože DEM worker interaguje s externími zdroji, měl by být nainstalován „co nejbližší to jde“ k systémům, na které nasazuje nebo s nimi spolupracuje. V některých případech je nutné definovat, který DEM Worker bude použit pro konkrétní workflow. Pro tento účel slouží skills (schopnosti). Ty je možné nastavit v konzoli vRealize Automation Designer. Skill (schopnost) je vztah mezi DEM Worker a workflow.

Interakce mezi službou Model Manager a DEM je znázorněná na obrázku níže.



Obrázek 8 - Interakce mezi Model Manager a DEM (zdroj: <https://cloudadvisors.net/vra-buch/kap-2/>)

Agents (Agenti)

vCloud Automation Center Management Agent

Tento typ agenta musí být nainstalován dříve, než administrátor spustí instalačního průvodce v rámci vRA. Musí být na každém Windows Server-u, na kterém běží IaaS služby. Hlavním úkolem agenta je sbírat telemetrii a registrovat IaaS servery k vRA Appliance. (25)

Proxy agents (Proxy agenti)

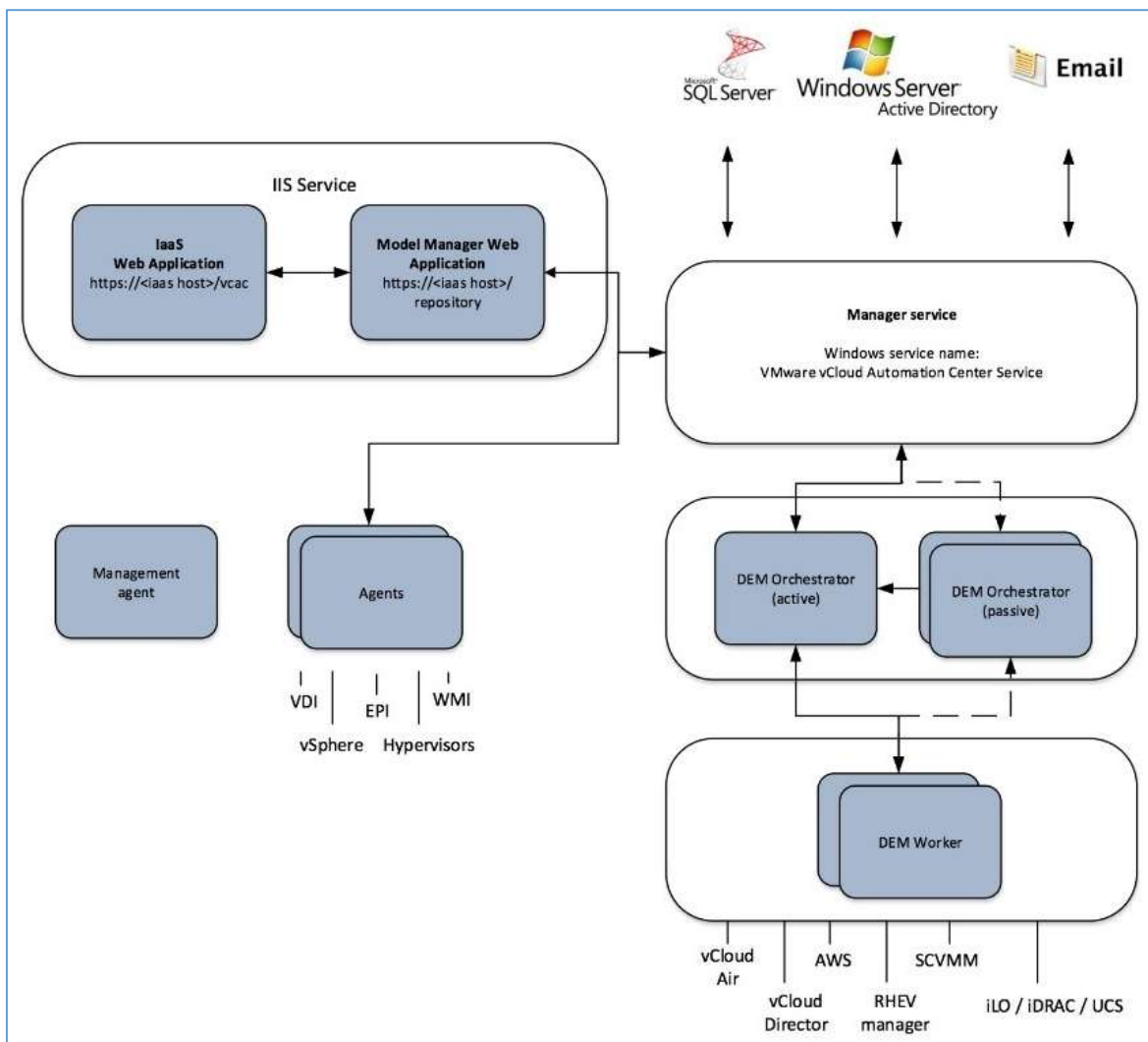
V některých případech DEM nemá implementovanou možnost spolupracovat s některými externími systémy napřímo, a proto je nutné využívat agentů, hlavně pro hypervizory vSphere a Hyper-V, které byli implementované v prvních verzích vRA. Podpora pro jiné hypervizory (např. KVM, Red Hat Enterprise Linux OpenStack, Amazon Web Services, Dell iDRAC nebo Cisco UCS) byla implementována přímo do DEM. Ve verzi vRA 7 existuje několik typů agentů: (25)

- *Virtualization proxy agents* – využívají se pro interakci s hypervizory. Jsou odpovědní za nasazení a vytváření virtuálních strojů, jako i synchronizaci dat

z hypervizoru do vRA databáze (např. importování dat o virtuálních strojích). Agent se instaluje jako služba ve Windows a musí být nainstalována a nakonfigurována pro každé jedno prostředí samostatně. To znamená, že v případě třech různých vCenter Server prostředí, je nutné instalovat tři agenty. Agenti tohoto typu jsou pro VMware vSphere, Microsoft Hyper-V a Citrix XenServer.

- *Virtual Desktop Integration (VDI) agents* – jsou odpovědní za registrování virtuálních strojů v externích systémech správy desktopů. Nejpopulárnější Virtual Desktop Infrastructure (VDI) systém Citrix XenDesktop. Po nasazení a zaregistrování poskytne vRA majiteli registrovaného desktopu přímé spojení s webovým rozhraním XenDesktop. Jeden agent může komunikovat s jedním nebo více Desktop Delivery Controller (DDC).
- *External Provisioning Integration (EPI)* – PowerShell agenti pomáhají se streamováním Citrix obrazů disků (Citrix disk images) na vyžádání. Z nich virtuální desktop bootuje a běží. Visual Basic agenti spouští skripty jako další krok v nasazování nebo rušení desktopů.
- *Windows Management Instrumentation (WMI)* – umožňuje sbírat informace ze serverů, které jsou pod správou vRA. Tento typ je vyžadován v případě, že se používá nasazení serverů pomocí Windows Image File (WIM).

Schéma interakcí komponent IaaS



Obrázek 9 - Schéma interakcí komponent IaaS (zdroj: <https://cloudadvisors.net/vra-buch/kap-2/>)

3.7.3 vRealize Orchestrator (vRO)

vRO je nástroj pro automatizaci prostředí a orchestraci byznys procesů, a to nejen v rámci vRA, ale je možné ho nasadit i samostatně a integrovat přímo s vSphere vCenter Server. Od verze vRA 7.x je vestavěný do vRA Appliance a zjednodušuje celou instalaci a správu, a toto vestavění patří k nejzásadnějším novinkám vRA verze 7.x a VMware doporučuje primárně tuto variantu místo původního externího serveru. (25)

Zatímco workflow lze také implementovat pomocí tradičních programovacích technik, vRO usnadňuje vývoj workflow díky integrovanému vývojovému prostředí a dalším vestavěným funkcím. Designéři šablon nebo administrátoři rádi využívají již

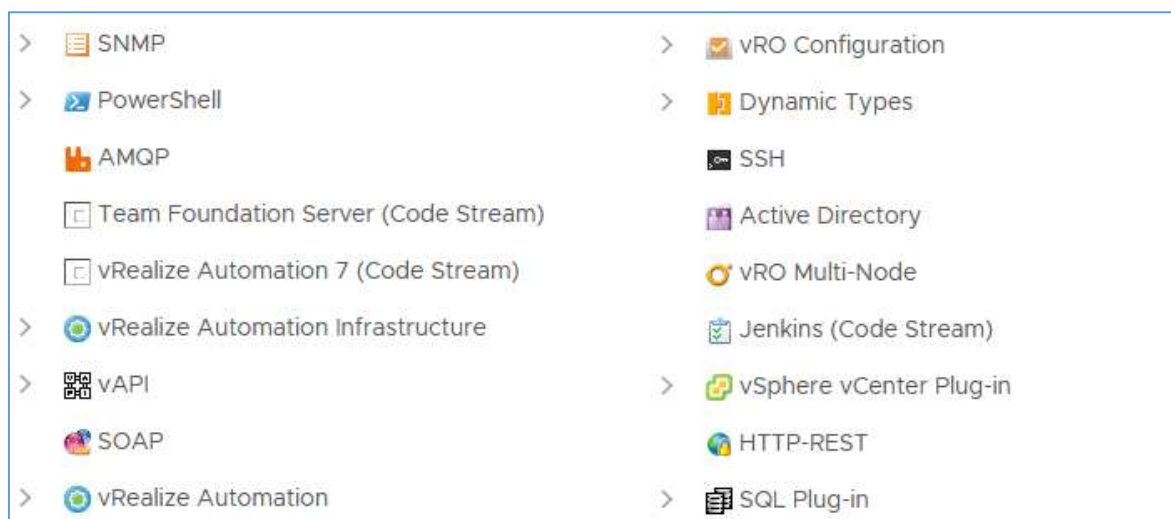
hotových skriptů, než aby je psali znova, proto vRO umožňuje exportovat a importovat workflow jako balíčky a tím usnadňuje přenositelnost a rozšiřitelnost. vRO přináší už v základu přes 500 workflow, které mají administrátorům a designerům šablon ulehčit práci bez programování. Samozřejmostí je rozšiřitelnost těchto workflow, které je možné programovat pomocí JavaScript-u a je zmíněná v části „Event Broker (Life Cycle Extensibility)“.

Licence pro vRO je automaticky obsažena v rámci VMware vSphere Standard a vyšší nebo vRA.

vRO může být nápomocné v různých případech užití:

- Životní cyklus infrastrukturních služeb může být přizpůsobené na míru (například je možné registrovat virtuální stroj v rámci CMDB nebo přiřadit vlastní hostname)
- Vývoj vlastních akcí přiřazených na blueprint (šablonu)
- Automatizace celé vRealize infrastruktury
- vRO je skvělý nástroj pro dodavatele třetích stran. V případě, kdy chtějí integrovat své řešení do vRA, mohou vyvinout vlastní workflow, dle něhož mohou být řešení publikovaná do katalogů služeb ve vRA. Takto publikované služby se nazývají taky Anything-as-a-Service (XaaS)

Pro co největší zjednodušení práce, vRO už v základu podporuje následující endpointy (koncové body) s kterými je dále možné ve workflow pracovat:



Obrázek 10 - vRO endpoint-y (zdroj: vlastní)

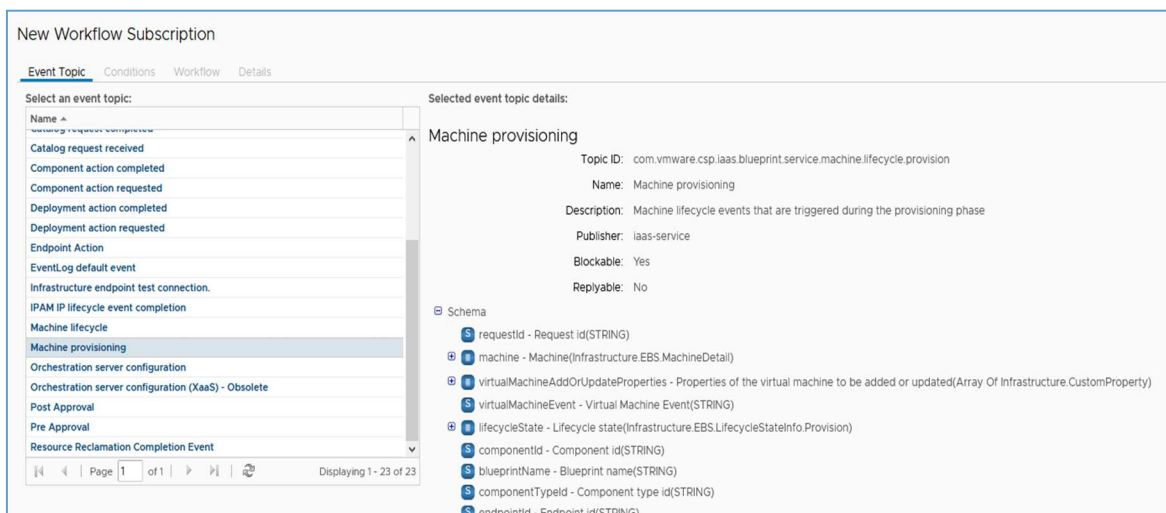
Event Broker (Life Cycle Extensibility)

Představuje novinku, která nahrazuje Workflow Stubs z verze 6.x a přidává nové možnosti rozšiřitelnosti a reakcí na různé události týkající se nasazování a životního cyklu virtuálních strojů ve spolupráci s vRealize Orchestrator, ale taky požadavků na schválení, nebo systémových logů. (26) (27)

Ve verzi 6.x bylo možné vložit vlastní logiku do předem definovaných fází životního cyklu IaaS použitím workflow pro změnu stavu, známých jako workflow stubs. Tím bylo možné uskutečnit volání vRealize Orchestratoru pro obousměrnou komunikaci s externími systémy. Předem definované fáze, které bylo možné použít:

- BuildingMachine (WFStubBuildingMachine).
- RegisterMachine (WFStubMachineRegistered).
- MachineProvisioned (WFStubMachineProvisioned).
- Expired (WFStubMachineExpired).
- UnprovisionMachine (WFStubUnprovisionMachine).
- Disposing (WFStubMachineDisposing).

Místo těchto šesti možností nabízí Event Broker ve verzi 7.x intuitivní uživatelské rozhraní pro administrátory, aby mohli svázat jednotlivé workflow (subscribe) k událostem přijímaným v RabbitMQ sběrnici. To buď na všechny události, nebo určené podmínkou (například spustit workflow, pokud všechny virtuální stroje s názvem začínajícím slovem „TEST_“ změni svůj stav na „zapnuto“). To nabízí mnoho nových možností a různých kombinací, a to včetně navázání „před“, „v průběhu“ a „po“ (pre-, during-, post-).



Obrázek 11 - Event Broker – Svázání (Subscribe) nového workflow s událostí (zdroj: vlastní)

Edit Workflow Subscription

Event Topic **Conditions** Workflow Details

Specify whether the workflow should be run for all events or based on conditions.

Run for all events
 Run based on conditions

All of the following

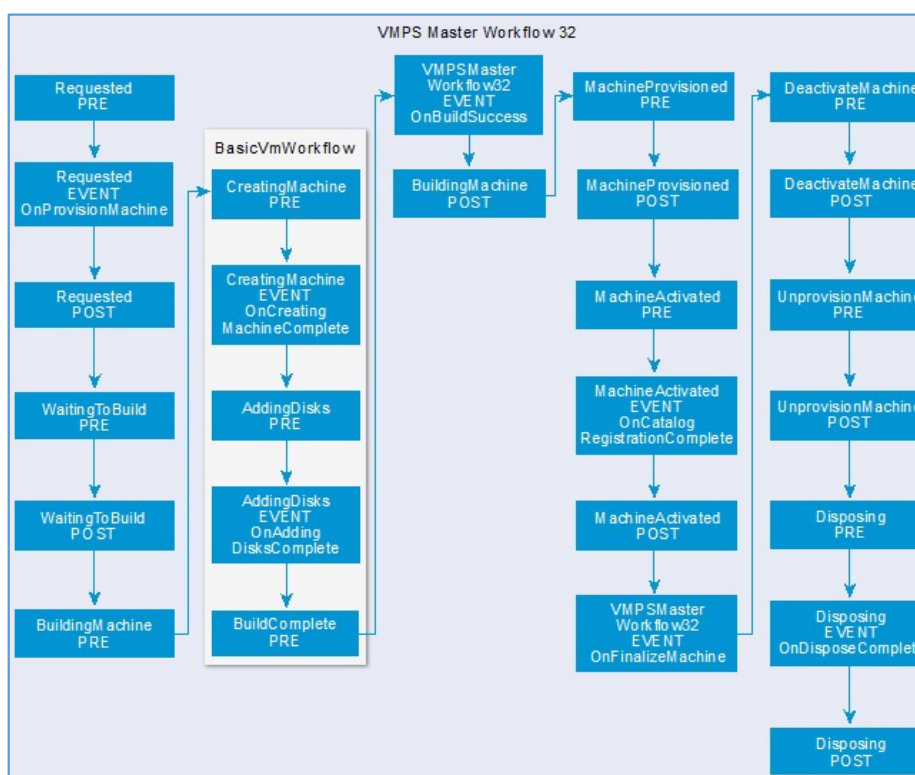
Data > Lifecycle state > Lifecycle state name	Equals	VMPSMasterWorkflow32.Requested	✘
Data > Lifecycle state > State phase	Equals	PRE	✘
Data > Machine > Properties > NetType	Equals	DHCP	✘

+ Add expression

Obrázek 12 - Příklad použití podmínek pro spuštění workflow (zdroj: vlastní)

Typickým případem užití (Use Case) je rozšíření workflow možností zaregistrovat vytvořený stroj (fyzický nebo virtuální) v konfigurační databázi (Configuration Management Database – CMDB), integrovat ho do prostředí Puppet, nebo Ansible.

Další významnou změnou oproti verzi 6.x je přidání VMPS Master Workflow, které zahrnuje všechny nasazovací workflow (Basic, Clone, Linked Clone, Flex Clone, Linux Kickstart, Virtual SCCM, WIM Image). Obrázek 9 znázorňuje VMPS Master Workflow a zdůrazňuje část BasicVmWorkflow, která je zodpovědná za vytvoření prázdného virtuálního stroje.

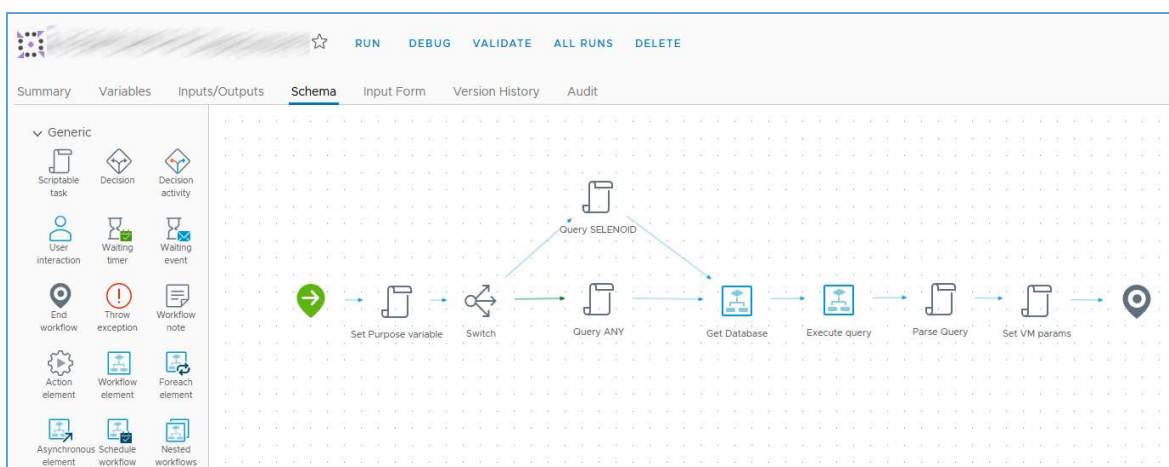


Obrázek 13 - VMPS Master Workflow (zdroj: <https://docs.vmware.com/en/vRealize-Automation/7.0/com.vmware.vra.extensibility.doc/GUID-D7B946C0-F7CB-4A95-AB48-5351800D59BB.html>)

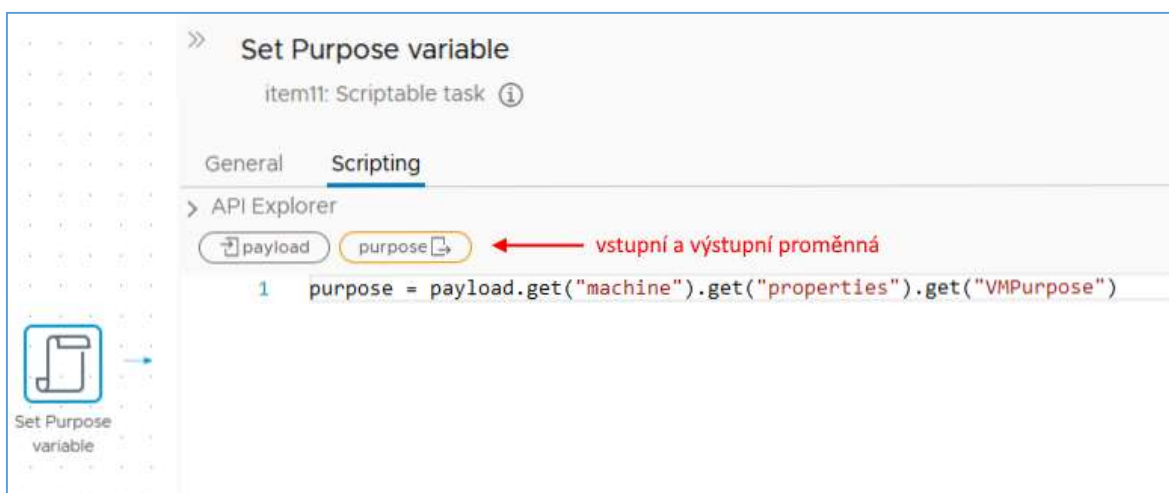
Workflow prochází čtyřmi různými fázemi:

- Request (Požadavek) – zahrnuje i schvalování
- Provision (poskytnutí/nasazení) – podporuje různé typy nasazení jako vytvoření prázdného virtuálního stroje, klonování, Kickstart, nebo WIM)
- Manage (Správa) – akce jako Power On, Power Off nebo práce se snapshoty
- Destroy (Zničení) – všechny akce související s deaktivací a likvidací strojů

Editace vRO workflow ve verzi 7.x probíhá v novém webovém rozhraní. Níže je ukázka, jak workflow vypadá. Jeho přesnější popis bude obsažen v praktické části této práce.



Obrázek 14 - Editace workflow (zdroj: vlastní)



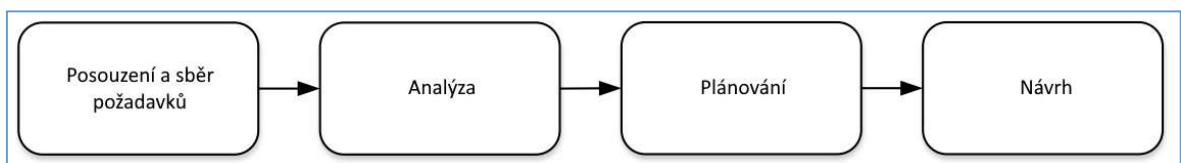
Obrázek 15 - Editace skriptu jednoho z úkolů (zdroj: vlastní)

3.8 Typy nasazení vRA (Deployment types / Deployment Design)

Navrhování prostředí vRA není jednoduchý úkol. Je to způsobeno základní architekturou, ale také požadavkem škálovat řešení tak, aby vyhovovalo potřebám velkých prostředí s tisíci virtuálních strojů. Kromě toho musí být splněny různé obchodní požadavky. Mnoho požadavků lze splnit hned tzv. out-of-the-box, ale jiné vyžadují promyšlený návrh před zahájením implementace. Tato kapitola se zabývá krátkým přehledem možností a způsobů nasazení prostředí vRA a tím, jak přistupovat před samotným nasazením. (28)

3.8.1 Přístup před nasazením

Na počátku nasazení prostředí vRA hledá mnoho odborníků ten správný způsob, jak k němu přistupovat. Ve všeobecnosti se dá říct, že každý projekt by měl začít sběrem požadavků a jejich analýzou. Správně identifikovat byznys cíle a technické požadavky pomůže identifikovat přesné ty komponenty vRA, které se mají nasadit. Proto je vhodné, aby odborník provádějící nasazení měl základy řízení IT projektů. Podobně jako u ostatních projektů, i zde může být proces návrhu rozdělen do čtyř etap. (28)



Obrázek 16 - Etapy návrhu prostředí (zdroj: <https://cloudadvisors.net/vra-buch/kap-3/>)

3.8.1.1 Posouzení a sběr požadavků

Je opravdu obtížné vybudovat infrastrukturu, pokud nejsou ujasněny obchodní potřeby a odpovídající požadavky. Proto je potřeba se v úvodní fázi návrhu zapojit do obchodního procesu a zjistit požadavky prostřednictvím zapojení různých zúčastněných stran. Zúčastněné strany zapojené do projektu mají někdy jen vágní představu o platformě pro správu, v tomto případě vRA. Zúčastněné strany se liší, firma od firmy. V takovém případě je dobré nejprve začít s tzv. proof of concept (PoC) (ověření konceptu). Ověření konceptu pomáhá pochopit technické problémy a je nesmírně cenným nástrojem při shromažďování další zpětné vazby, aby bylo zajištěno, že máme spolehlivé informace o plánování.

Další přístupy k získání zpětné vazby zahrnují:

- Individuální schůzky s vedoucími IT a zúčastněnými stranami top managementu.
- Schůzky s týmem, který je zodpovědný za řízení technických operací / správu sítě.
- Prototypování, dotazníky, případy užití nebo brainstorming.

V mnoha společnostech najdeme skupinu lidí, kteří se snaží zvýšit technický rozvoj společnosti. Většinou jsou to testéři nebo vývojáři, kteří potřebují rychle nasadit svoje produkty nebo systémy v rámci procesu vývoje nebo testování. Takové systémy mají často krátkou životnost a měly by být izolovány od produkčního prostředí. Pokud ale budou čekat, než tradiční systém IT podpory poskytne požadované zdroje, stráví značné množství času čekáním, což vede ke zpomalení vývoje a samozřejmě zvýšení nákladů na vývoj. I když koncoví uživatelé mohou požadovat zdroje od veřejných poskytovatelů sami, často to není žádoucí, protože by to často obcházel stávající pravidla správy a dodržování předpisů. Použití funkce katalogu služeb vRealize Automation pomáhá splnit tyto cíle a zároveň je schopen řídit náklady a využití cloudové infrastruktury. V neposlední řadě se mnoho společností snaží vybudovat vlastní privátní cloud, aby bylo možné dodávat IT zdroje stejným způsobem, jaký poskytuje veřejný cloudový poskytovatel. Tato poslední varianta, společně s nutností rychle vyvíjet, testovat a nečekat na IT oddělení v případě potřeby poskytnutí virtuálního stroje, je hlavním motivem této práce. (28)

3.8.1.2 Analýza

Jakmile jsou shromážděné podrobnosti o požadavcích od všech zúčastněných stran, je potřeba je analyzovat předtím, než se začne plánovat návrh. To zahrnuje identifikaci služeb, které mají být nasazeny. Řekněme například, že produkční prostředí bude v první fázi nabízet možnost nasadit operační systém Linux a všechny takto nasazené virtuální stroje musí být integrované do stávajícího prostředí (například, mají být registrované v serveru Puppet nebo mají mít zapnuty automatické aktualizace). Dalším důležitým aspektem je plánování kapacit. Je nezbytné vědět, kolik virtuálních strojů (a jaký druh virtuálních strojů) bude nasazen, aby bylo možné přesně odhadnout hardwarové požadavky nasazení vRealize Automation a taky správně vybrat typ licence. (28)

Další otázky zahrnují:

- Existuje potřeba různých platforem?
- Je potřeba splnit určité požadavky na dodržování předpisů (compliance)?
- Jaký druh koncových uživatelů používá samoobslužný portál? Jak je chcete seskupit?
- Je potřeba multi-tenant? Mohou různé tenanty sdílet stejný hardware nebo musí být od sebe zcela izolované?
- Existuje nějaký nástroj, který musí být integrován do procesu nasazení? Mnoho společností například používá nástroje pro správu konfigurace, jako např. Puppet, Chef nebo Ansible.
- Jaký druh serveru DNS společnost používá?
- Jaké další nástroje musí být integrovány (například antivirus, CMDB)?

3.8.1.3 Plánování

Po shromáždění všech požadovaných informací můžeme začít s etapou plánování. Plánování často nezahrnuje pouze problémy s vRA, ale také úložiště, síť nebo virtuální prostředí. (28)

Stručně řečeno, plánování by mělo zahrnovat také následující aspekty:

- Plánování sítě
- Plánování úložiště
- Plánování ESXi serverů
- Plánování clusterů
- Plánování vysoké dostupnosti
- Plánování kapacity
- Konfigurace VM
- Geolokační plánování

3.8.1.4 Návrh

Jakmile jsou zpracovány předešlé etapy, je možné začít s vytvářením dokumentu návrhu. Je to fáze, která bezpochyby vyžaduje nejvíce času a je rozhodně nejtěžší. Návrh lze rozdělit na základní návrh cloudové infrastruktury (veřejný/privátní cloud) a návrh samotné vRA instalace. Nejdůležitějšími výstupy fáze návrhu jsou koncepční návrh, logický design a fyzický design. Mnohdy se stává, že těmito dokumentům ne všichni porozumí, takže je vhodné tyto dokumenty jednotlivě charakterizovat. (28)

Koncepční návrh by měl být vytvořen jako první. Vychází pouze z obchodních požadavků a neobsahuje žádné informace o implementaci. Koncepční návrh lze poté použít k poskytnutí základního přehledu výše postaveným manažerům, kteří nepotřebují vědět technické detaily. Jako příklad, v případě koncepčního návrhu vRA bývají obvykle uvedené obchodní požadavky takto:

- Poskytování autentizace pomocí jednotného přihlášení (Single sign-on).
- Nastavení dedikované sítě pro každou business group (termín ve vRA, bude vysvětleno později, ve zkratce – logické oddělení uživatelů, jejich přiřazených zdrojů, sítí, úložišť atd.).
- Potřeba nasazení služby load balanceru.
- Nastavení interního systému správy IP adres (IPAM – IP Address Management).

Logický návrh obsahuje více podrobností a definuje high-level pokyny pro implementaci. Obsahuje sice high-level pokyny pro implementaci, ale stále neobsahuje žádné detaily. Kupříkladu, logický návrh prostředí vRA může například charakterizovat následující obchodní požadavky:

- Každé oddělení bude mapováno do business group, tak aby v dané business group bylo možné nasadit maximálně 50 virtuálních strojů. Všechny business group jsou od sebe izolované.
- Sítě mohou být interní nebo externí. Stroje nasazené do interních sítí budou mít privátní IP adresy, zatímco stroje v externí síti získají veřejnou IP adresu.
- Správci business group mají povoleno nasadit do sítě load balancer a nastavit pravidla firewall-u.

Fyzický návrh zahrnuje podrobnosti implementace a je založen na logickém návrhu. Například to může být:

- Počet a velikost (CPU a paměť) každé komponenty vRealize Automation.
- Obecný návrh sítě včetně informací o konfiguraci VLAN⁷, routing-u, gateway (směrovačů), serverů DNS a DHCP.

⁷ VLAN – Virtual Local Area Network (Virtual LAN)

3.8.2 Části návrhu

3.8.2.1 Návrh cloudové infrastruktury

Na rozdíl od mnoha jiných řešení VMware je fáze návrhu řešení platformy pro správu cloudu složitější než ostatní, protože vRealize Automation nedokáže fungovat samostatně, ale musí být integrován do existujícího virtuálního prostředí. Na základě zadání z fáze plánování musíme určit, zda se plánuje nasazení pro malé, střední nebo velké prostředí. Obvykle by měly být vytvořeny následující dokumenty architektonického návrhu:

- Návrh vrstvy správy VMware vSphere (VMware vSphere management layer design)
- Návrh síťové vrstvy (Network layer design)
- Návrh vrstvy úložišť (Storage layer design)
- Návrh kapacit a škálování (Sizing and scaling capability design)

Návrhový dokument by měl zahrnovat také následující dílčí komponenty:

- Přípravenost
- Výkon
- Spravovatelnost
- Předpoklady
- Omezení
- Rizika a důsledky
- Bezpečnost
- Zotavení (Recovery)

Protože obvykle bude vRA nasazeno na existující virtuální infrastrukturu, je zřejmé, že při vytváření návrhu a implementaci na tuto infrastrukturu, bude potřeba vykonat spoustu práce. Co se týče praktické části práce, základní infrastruktura, na kterou probíhá implementace vRA už byla navržena, používá se několik let a nebude měněna. Toto téma je již hojně zpracováno v literatuře a tzv. white papers, takže se tomu nebudeme dále věnovat. (28) (29) (30)

3.8.2.2 Návrh vRealize Automation

Mezi nejdůležitější dokumenty popisující návrh vRA patří:

- Katalog služeb
- Možnosti nabízené IaaS (Infrastructure-as-a-Service offerings)
- Pokročilý návrhář služeb (Advanced Service Designer)
- Integrace vRealize Business a náklady na služby
- Monitorování kapacity a výkonu
- Workflow v nástroji Orchestrator a možné integrace třetích stran
- Integrace NSX

Níže jsou charakterizované zmíněné body.

Prostřednictvím **katalogu služeb** mohou organizace vytvářet a spravovat katalogy IT služeb, které jsou schváleny pro použití ve vRealize Automation. Tyto IT služby mohou zahrnovat vše od virtuálních strojů, serverů, softwaru a databází, až po kompletní vícevrstvé aplikační architektury. Katalog služeb pomůže centrálně spravovat běžně nasazené IT služby a dosáhnout konzistentního řízení a splnění požadavků na dodržování předpisů (compliance). A co je nejdůležitější, uživatelé budou moci rychle nasadit pouze schválené IT služby, které potřebují. Při sestavování projektového dokumentu pro katalog služeb musíme zvážit, jak roztrdit publikované položky katalogu do služeb, rozhodnout, kdo je oprávněn provádět akce na položkách katalogu, a zajistit, aby položky katalogu byly řádně publikovány.

Modul **IaaS** nabízí možnost vytvořit a nasadit servery nebo desktopy ve fyzických, virtuálních a cloudových prostředích. vRealize Automation pracuje se zmíněnými platformami a snaží se návrháři služeb poskytnout abstraktní a konzistentní model tak, aby bylo nasazení IaaS zdrojů co nejuniverzálnější. Kromě toho, že je možné nasadit služby do výše uvedených prostředí, vRealize Automation centrálně řídí životní cyklus všech nasazených prostředků. Při návrhu části IaaS je opět zapotřebí hodně práce. Model IaaS v rámci vRealize Automation musí být sladěn s použitou infrastrukturou, která zahrnuje například definování výpočetních zdrojů, fabric groups (český překlad neexistuje), rezervací a blueprintů (návrhových plánů služeb – dále v práci bude používáno slovo blueprint pro jednoznačné označení).

Nástroj **Advanced Service Designer** pomáhá implementovat a publikovat operace typu „day-2“ (dle terminologie vRA se tím myslí všechny operace, které následují po prvních 24 hodinách, v průběhu, kterých má nastat samotné nasazení vRA bez katalogu služeb) na nasazených zdrojích a integrovat workflow XaaS (Everything-as-a-Service) založené na vRealize Orchestrator. Podobně jako u služeb IaaS musí být i v tomto případě navrženo mnoho komponent, včetně plánů služeb, mapování zdrojů a akcí.

Náklady poskytnutých zdrojů vyžadují také důkladný návrh. **vRealize Business** v podstatě pomáhá identifikovat náklady na cloudovou infrastrukturu a mapovat tyto náklady na jednotlivé nabídky služeb. V mnoha organizacích je klíčovým požadavkem důsledný mechanismus stanovení cen a zpětného zúčtování zdrojů jednotlivým oddělením. To zahrnuje nastavení výkazů a zohlednění variant nabídky služeb. Jako příklad lze uvést možnost nechat koncového uživatele vybrat si z různých možností, jako je zálohování, úroveň SLA, nebo monitorování integrace. Všechny tyto možnosti mohou vést k rozdílným cenám, i když je nasazen stejný obraz virtuálního stroje. vRealize Business není součástí práce, protože je licencován samostatně a nebyl pro projekt zakoupen, zmíněn je jen jako součást rodiny vRealize.

Monitorování a správa kapacit musí být také řádně navrženy. Z operačního hlediska virtualizace mění tým IT infrastruktury z tvůrce systému na poskytovatele služeb. Aplikační tým již nevlastní fyzickou infrastrukturu, ta je místo toho nyní sdílenou infrastrukturou. Tím se vytvoří dvouvrstvé řízení kapacit. Na úrovni virtuálních strojů provádí správu kapacity aplikační tým a na úrovni fyzické infrastruktury se o to musí postarat infrastrukturní tým.

Virtuální stroje nasazené pomocí vRealize Automation se obvykle nevytvoří ve vakuu, ale musí být nějak integrovány do existujícího prostředí. Typická integrace zahrnuje instalaci antivirového softwaru nebo monitorovacího agenta, integraci konfigurace stroje do infrastruktury serveru Puppet nebo připojení k Active Directory. Některé z těchto kroků lze nakonfigurovat tzv. out-of-the-box, zatímco jiné lze nainstalovat pomocí **vRealize Orchestrator** a proto je nutné plánování i v této oblasti.

Virtuální stroje nasazené pomocí vRealize Automation vyžadují síťové připojení. Pro mnoho společností stačí rozmístit stroje do stávajících sítí (nebo jen vytvořit síť určené předem pro vRealize Automation). Někdy však existuje potřeba sofistikovaného návrhu sítě založeného na principech mikrosegmentace. V takovém případě může pomoci **VMware**

NSX. Je to produkt, který je určen pro dynamické vytváření softwarově definovaných sítí, firewallů nebo load-balancerů. VMware NSX není součástí práce, protože je licencován samostatně a nebyl pro projekt zakoupen, zmíněn je jen jako součást rodiny vRealize. (30)

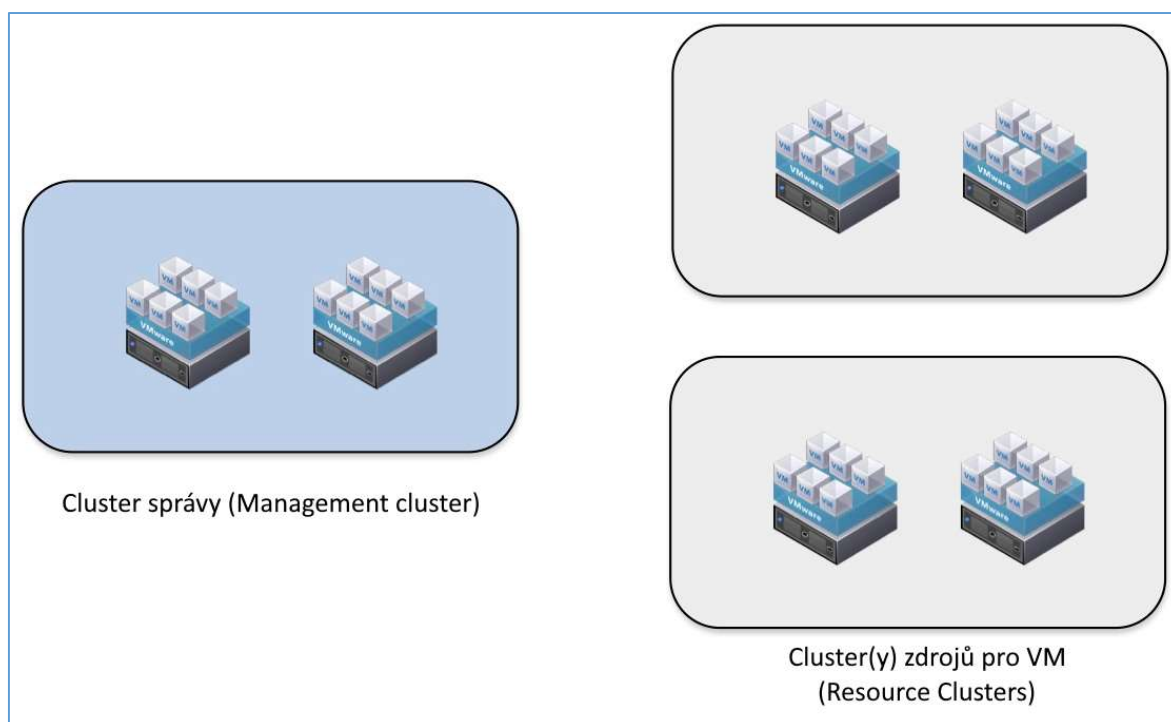
3.8.3 Návrh prostředí vRealize Automation

Jak již bylo zmíněno, vytvoření solidního řešení vyžaduje důkladné plánování. Na rozdíl od jiných produktů VMware, je nasazení vRealize Automation pořád poměrně těžkopádné (i když nový průvodce instalací to ve velké míře usnadňuje).

3.8.3.1 Konfigurace clusteru

Než začneme vytvářet návrh prostředí, musíme zvážit tři druhy virtuálních strojů v prostředí automatizace vRealize:

- Virtuální stroje, které byly nasazeny pro koncové uživatele skrze katalog služeb.
- Virtuální stroje, které vytváří infrastrukturu, jako je Active Directory, vSphere, monitoring, SQL Server atd.
- Virtuální stroje, které jsou produkční, nesouvisí s vRealize Automation, ale nachází se na ESXi serverech, které budou hostovat i VM vytvořené skrze vRA.



Obrázek 17 - Typu clusterů (zdroj: <https://cloudadvisors.net/vra-buch/kap-3/>)

Obrázek 13 znázorňuje doporučení, jak by měla vypadat logická high-level architektura řešení.

Zdroje pro cluster správy, tedy ty, kde bude nainstalované vRA, by měly být odděleny od těch, které se budou využívat pro samotné virtuální stroje koncových uživatelů. Odvíjí se to samozřejmě od toho, jestli má daná společnost dostatek finančních prostředků a jestli to její infrastruktura dovoluje. Proto je to pouze doporučení, ne nutná podmínka pro chod samotného vRA.

Existuje několik důvodů pro použití vyhrazených clusterů pro správu a pro uživatelské VM:

- Díky oddělení je zvýšená bezpečnost a vyšší izolace pracovních zátěží (workload) správy.
- Pracovní zátěž (workload) zdrojů a správy se navzájem neovlivňuje a nevyčerpají si výpočetní kapacitu.
- Zlepší se zotavení po katastrofě (disaster recovery) a kontinuita podnikání (business continuity).
- Cloudové prostředky mohou být konzistentně a transparentně spravovány, rozčleněny a škálovány horizontálně.
- Protože jsou všechny komponenty správy nasazeny v relativně malém a spravovatelném clusteru správy, umožňuje to rychlejší hledání a řešení problémů.

Množství clusterů závisí na různých kritériích. Za prvé, záleží na druhu virtuálních strojů, a tedy na pracovní zátěži (workload), která má být zajištěna. Pokud existuje potřeba nastavení různé úrovně služeb (Service Level Agreement – SLA) (například úroveň 1 pro kritické aplikace, úroveň 2 pro standardní produkční systémy a úroveň 3 pro testování), doporučuje se pro každou z úrovní služeb vytvořit dedikovaný cluster. Dalším případem užití pro více než jeden cluster může být požadavek na poskytování služeb v jiných lokacích (geolocation). (28) (30)

3.8.3.2 Hardwarové požadavky

Instalace vRealize Automation se skládá z různých virtuálních strojů a komponent. Zatímco v malých prostředích mohou být některé z těchto komponent nainstalovány na jednom serveru, ve větším prostředí (z důvodů vysoké dostupnosti a škálování), je více než žadáné distribuovat tyto komponenty na různé servery. Tabulka níže zobrazuje role a jejich požadavky na hardware.

Role serveru	Komponenta	Požadované HW specifikace	Doporučené HW specifikace
vRealize Automation Appliance	<ul style="list-style-type: none"> vRealize Automation Services vRealize Orchestrator vRealize Automation Appliance Database 	<ul style="list-style-type: none"> CPU: 4 vCPU⁸ RAM: 18 GB Disk: 140 GB Síť: 1 GB/s 	Stejně jako požadované
Infrastructure IaaS Core Server	<ul style="list-style-type: none"> Web site Manager Service DEM Orchestrator DEM Worker Proxy Agent 	<ul style="list-style-type: none"> CPU: 4 vCPU RAM: 8 GB Disk: 40 GB Network: 1 GB/s 	Stejně jako požadované
Infrastructure IaaS Web Server	<ul style="list-style-type: none"> Web site 	<ul style="list-style-type: none"> CPU: 2 vCPU RAM: 8 GB Disk: 40 GB Network: 1 GB/s 	<ul style="list-style-type: none"> CPU: 2 vCPU RAM: 8 GB Disk: 40 GB Network: 1 GB/s
Infrastructure IaaS Manager Server	<ul style="list-style-type: none"> Manager Service DEM Orchestrator 	<ul style="list-style-type: none"> CPU: 2 vCPU RAM: 8 GB Disk: 40 GB Network: 1 GB/s 	<ul style="list-style-type: none"> CPU: 2 vCPU* RAM: 8 GB Disk: 40 GB Network: 1 GB/s <p>*4 vCPU, pokud probíhá více než 100 současných nasazení.</p>

⁸ vCPU – Virtual CPU

Infrastructure IaaS Web/Manager Server	<ul style="list-style-type: none"> • Infrastructure Web/ManagerServer 	<ul style="list-style-type: none"> • CPU: 2 vCPU • RAM: 8 GB • Disk: 40 GB • Network: 1 GB/s 	<ul style="list-style-type: none"> • CPU: 2 vCPU • RAM: 8 GB • Disk: 40 GB • Network: 1 GB/s
Infrastructure IaaS DEM Server	<ul style="list-style-type: none"> • (jeden nebo více) DEM Workers 	<ul style="list-style-type: none"> • CPU: 2 vCPU • RAM: 8 GB • Disk: 40 GB • Network: 1 GB/s Per DEMWorker 	<ul style="list-style-type: none"> • CPU: 2 vCPU* • RAM: 8 GB • Disk: 40 GB • Network: 1 GB/s Per DEMWorker <p>*4 vCPU, pokud probíhá více než 100 současných nasazení.</p>
Infrastructure IaaS Agent Server	<ul style="list-style-type: none"> • (jeden nebo více) Proxy Agent 	<ul style="list-style-type: none"> • CPU: 2 vCPU • RAM: 8 GB • Disk: 40 GB • Network: 1 GB/s 	<ul style="list-style-type: none"> • CPU: 2 vCPU • RAM: 8 GB • Disk: 40 GB • Network: 1 GB/s
Microsoft SQL Database Server	<ul style="list-style-type: none"> • Infrastructure Database 	<ul style="list-style-type: none"> • CPU: 2 vCPU • RAM: 8 GB • Disk: 40 GB • Network: 1 GB/s 	<ul style="list-style-type: none"> • CPU: 8 vCPU • RAM: 16 GB • Disk: 80 GB • Network: 1 GB/s

Tabulka 3 - HW požadavky pro jednotlivé role (zdroj: <https://docs.vmware.com/en/vRealize-Automation/7.6/vrealize-automation-76-reference-architecture.pdf>)

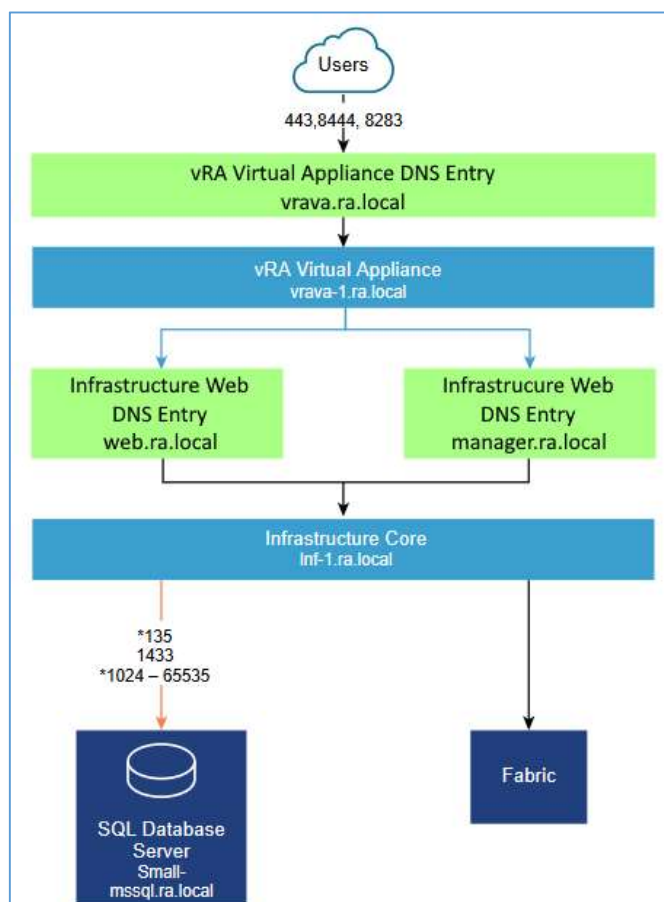
3.8.3.3 Architektura nasazení (Deployment architecture)

Všechny předešlé kapitoly charakterizovaly vRealize Automation, jeho části a směřovaly k závěrečné části, tedy k samotnému návrhu jak a kde instalovat komponenty vRA. VMware jednoduše rozděluje nasazení na dvě možnosti – minimální a enterprise (to je ve skutečnosti rozděleno na dva podtypy)

Minimální nasazení (Minimal deployment)

Minimální typ nasazení je vhodný pro vývojové účely, trénink nebo proof of concept. Je velice jednoduchý na instalaci, protože obsahuje pouze dva servery. První je samotná vRealize Automation Appliance a druhý server obsahuje všechny IaaS komponenty. Tento typ má samozřejmě omezení: (30)

- 10000 spravovaných serverů
- 500 položek katalogu
- 10 současných nasazení (ve stejnou chvíli)



Obrázek 18 - Minimální typ nasazení (zdroj: <https://docs.vmware.com/en/vRealize-Automation/7.6/vrealize-automation-76-reference-architecture.pdf>)

Enterprise nasazení (Enterprise deployment)

Tento typ VMware lze rozdělit na dva podtypy – Střední a Velký (Medium and Large Deployments). Rozdíl v nich je především ve větší robustnosti a vysoké dostupnosti ale také v počtu podporovaných položek v katalogu nebo nasazení. Zároveň je tento způsob doporučován pro produkční použití. (30)

Následující tabulka zobrazuje rozdíl mezi středním a velkým nasazením a umístěním jednotlivých komponent.

Komponenta	Počet instancí ve středním nasazení	Počet instancí ve velkém nasazení
vRealize Automation Appliance	3	3
Infrastructure Web	2	2
Manager Server	instaluje se společně s Infrastructure Web	2
Infrastructure DEM Server	2	2
Infrastructure Agent Server	2	2
MSSQL Failover Cluster Instanc	1	1
vRealize Automation appliance load balancer	1	1
Infrastructure Web load balancer	1	1
Infrastructure Manager Service load balancer	1	1

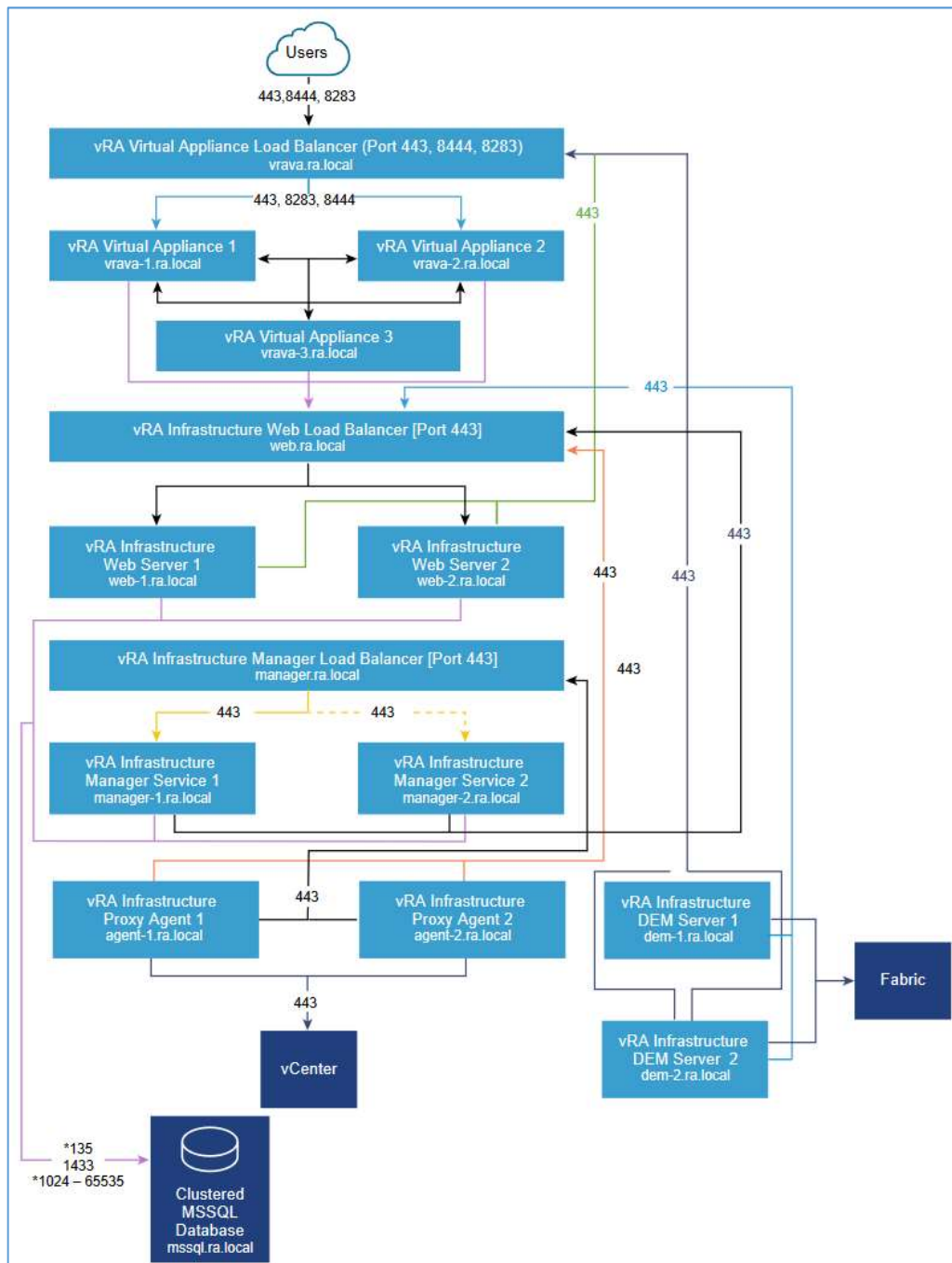
Tabulka 4 - Rozdíl v počtu instancí jednotlivých nasazení (zdroj: <https://docs.vmware.com/en/vRealize-Automation/7.6/vrealize-automation-76-reference-architecture.pdf>)

Maximální množství položek, které jsou podporovány:

Položka	Střední nasazení	Velké nasazení
Spravované stroje	30000	75000
Položky v katalogu	1000	2500
Počet současných nasazení	50	100

Tabulka 5 - Maximální počet položek (zdroj: <https://docs.vmware.com/en/vRealize-Automation/7.6/vrealize-automation-76-reference-architecture.pdf>)

Jak je vidět z požadavků výše, v obou případech už je vyžadován load-balancer. Ten ale není součástí vRealize Automation, VMware proto doporučuje použít buď jejich vlastní řešení NSX, nebo load-balancery třetích stran, jako např. F5 Big IP nebo Citrix Netscaler. Jelikož se jedná o standardní HTTP load-balancer, lze kromě výše uvedených použít i Apache HTTPD, nebo Nginx. Následující schéma zobrazuje největší a nejrobustnější typ nasazení vRealize Automation. (30)



Obrázek 19 - Enterprise deployment (zdroj: <https://docs.vmware.com/en/vRealize-Automation/7.6/vrealize-automation-76-reference-architecture.pdf>)

3.8.3.4 Posouzení vysoké dostupnosti a škálovatelnosti

Rozhodnutí, zda využít vysoké dostupnosti a pro kterou komponentu vždy závisí na posouzení konkrétního případu užití a zátěže, kterou bude prostředí vRealize Automation vytěžováno. Na základě těchto informací se rozhodne, kolik serverů bude finálně instalováno a nasazeno. Zatímco pro proof-of-concept nebo testovací prostředí bude postačovat minimální instalace, pro produkční prostředí bude potřeba instalovat řešení s vysokou dostupností.

3.8.3.5 vRealize Automation appliance

vRealize Automation Appliance podporuje instalaci v módu vysoké dostupnosti typu active-active. Pro zajištění vysoké dostupnosti musí být servery umístěny za load-balancerem.

Od verze vRA 7 a výše je interní databáze vPostgres automaticky v módu cluster. Zároveň je doporučeno ji používat přednostně než dedikovanou instanci, která byla preferována v předešlých verzích. Bohužel, prozatím tato interní databáze nemá automatický failover. V případě pádu je nutné manuálně povýšit jinou instanci na hlavní.

Stejný princip platí pro interní instanci Orchestratoru. Nicméně z důvodu vyššího výkonu je možné nainstalovat dedikované servery Orchestratoru a umístit je za load-balancer.

V případě clusteru active-active je žádoucí, aby byli uživatelé pro každý svůj navazující požadavek přesměrováni na stejný server. Proto musí load-balancer podporovat funkci „sticky-session“ nebo „session affinity“. (30)

3.8.3.6 Služby ověření

Podobně jako v předešlých případech, služby ověření jsou součástí vRealize Automation Appliance a není nutné konfigurovat žádné jiné instance.

3.8.3.7 Komponenty IaaS

IaaS web server

Tak jako vRealize Automation Appliance, i webový server IaaS může být umístěn za load-balancer a pracovat v módu active-active. Nicméně, uživatelé nepřístupují přímo na webový server, pouze výše zmíněnou vRA Appliance a proto není potřeba, aby load-

balancer podporoval „sticky-session“ nebo „session affinity“. Namísto toho je možné použít algoritmy „Least Response Time“ (Nejkratší doba odezvy) nebo „Round-Robin“.

I když je možné instalovat několik instancí webového servere IaaS, pouze jedna obsahuje Model Manager (nejčastěji první server). Ostatní servery obsahují pouze web.

Výkon webového serveru je velice jednoduché navyšovat. V případě nárůstu latencí stačí monitorovat využití CPU a RAM a eventuálně je navýšit, což je jednodušší, než instalovat a přidat další server. (30)

Manager Service

Na rozdíl od výše uvedených součástí, Manager Service nepodporuje mód active-active. Místo toho funguje druhý server pouze v pohotovostním režimu pro případ pádu aktivního serveru. Z hlediska výkonu by to neměl být problém, protože jedna instance může snadno obsluhovat desítky tisíc virtuálních strojů. Samotný failover při selhání lze provést ručně, nebo prostřednictvím load-balanceru. V tomto případě však nelze použít žádný load-balance algoritmus, ale pouze detekci, že primární server neběží. (30)

DEM Orchestrator

Podobně jako Manager Service může být aktivní pouze jedna instance DEM Orchestratoru. Proto zde existuje konfigurace v módu active-passive. Pro ni však není vyžadován load-balancer, neboť DEM Orchestrator se umí automaticky monitorovat navzájem. Ve chvíli, kdy DEM Orchestrator startuje, hledá ostatní běžící instance. Když není žádná nalezena, stane se primární instancí. Pokud je nalezena alespoň jedna instance, spustí se jako sekundární a monitoruje primární. Ve chvíli, kdy primární vypadne, sekundární automaticky přebírá její místo. V případě, že se původní instance opět nainstaluje, po detekci nové primární instance se tato přepne do sekundárního módu.

Již dříve bylo zmíněno, že DEM Orchestrator a Model Manager velice úzce spolupracují, tudíž je dobré je mít nainstalované navzájem co nejbližší a současně by měli mít k dispozici vysokou šířku pásma pro komunikaci. (30)

DEM Worker

Komponenta, která skutečně spouští a vykonává workflow, by měla být umístěna co nejbližší externím zdrojům, se kterými pracuje a komunikuje. Například, v případě vícero lokací data center, by mělo být zajištěno, že DEM Worker bude běžet v každé lokaci.

DEM Worker využívá hlavně CPU a může vykonávat až patnáct workflow paralelně. Pokud je fronta workflow konstantně vysoká, je vhodné zvážit nejdříve vertikální škálování (navýšit CPU a RAM).

Ve chvíli výpadku konkrétního DEM Workeru, stornuje DEM Orchestrator všechny workflow tímto Workerem vykonávané a přiřadí je jinému. DEM Worker rovněž nevyžaduje žádný load-balancer, protože je řízen DEM Orchestrátorem. (30)

Microsoft SQL Server

Poslední komponenta, u které je vhodné zvážit možnost vysoké dostupnosti, je Microsoft SQL Server. SQL Server podporuje několik metod pro vytvoření vysoké dostupnosti jako „cold standby server“, „mirrored standby server“ (zrcadlení) nebo cluster (doporučeno společností VMware). Způsob konfigurace ale není součástí této práce. (30)

4 Vlastní práce

Praktická část se zabývá aplikací teoretických znalostí na reálnou implementaci ve společnosti ZOOM International a.s. Při realizaci praktické části bylo vycházeno z existujícího technického řešení virtualizační farmy a souvisejících firemních procesů. Práce vychází z požadavků, které vzešly od zaměstnanců společnosti, primárně z oddělení vývoje a testování, ale také zákaznické podpory, tedy všech oddělení, která používají virtuální prostředí pro vývoj nebo testování produktu vyvíjeného společností. Na základě analýzy těchto požadavků byla navržena architektura a provedena implementace řešení. Poslední část práce je zaměřena na výpočet TCO souvisejícího s implementací.

Mezi činnosti autora v rámci společnosti spadají, mimo jiné, také řízení IT oddělení a úkony spojené se správnou uvedenou infrastrukturou. IT oddělení je součástí týmu Operations, které má na starosti všechny provozní záležitosti společnosti.

Z důvodu bezpečnosti a zachování firemního tajemství budou veškeré reálné údaje (např. IP adresy, doménová jména, uživatelská jména) maskovány nebo pozměněny tak, aby nebylo možné zjistit žádné informace o skutečné infrastruktuře. Architektura, rozložení komponent, nebo síťových prvků je znázorněná ve skutečné podobě a dostatečná reprezentativnost pro účely práce zůstala zachována za podmínky uvedené výše.

4.1 Charakteristika společnosti

Vybraná společnost s názvem ZOOM International a.s. působí na trhu od roku 1999, byla založena v Praze a její hlavní zaměření je vývoj software a řešení, která jsou určena pro call centra po celém světě. Mezi hlavní produkty patří nahrávání telefonních a video hovorů, hodnocení agentů v call centrech, hlasová analýza, nebo plánování směn agentů. V současné době má společnost více než 150 zaměstnanců a působí ve více než 90 zemích. Její centrála sídlí v Praze, další pobočky jsou umístěné v Americe, na Slovensku, v Rusku nebo Velké Británii. Práce se zaměřuje primárně na potřeby pražské centrály a pobočky na Slovensku, kde jsou v obou případech umístěná vývojová centra.

4.2 Analýza stavu před implementací (Současný stav)

Pro udržení konkurenceschopnosti začala společnost s vývojem software, který bude možné nasadit jak on-premise tak v cloud-u. IT oddělení, jež odpovídá za kompletní

infrastrukturu, architekturu a rozvoj sítě musí rozvoj infrastruktury pečlivě plánovat, protože je úzce svázán se strategií firmy a tím pádem veškerá rozhodnutí a změny musí tuto strategii podporovat. Tím, že její strategie směřuje ke cloudovým řešením začali se množit požadavky z různých oddělení, které by se dali charakterizovat jako poptávka pro dynamičtějších procesech a větší volnosti v rámci vývoje a testování.

Z důvodu potřeby cloudového řešení musí společnost využívat dvou typů infrastruktury. V první řadě dlouhodobě buduje on-premise infrastrukturu, která neslouží pouze pro provozní účely (doménové řadiče, emailové servery apod.), ale je primárním místem pro vývoj všech produktů a běh všech podpůrných systémů pro vývoj a testování. Z důvodu unifikace, jednoduchosti a faktu, že pro vývoj produktu nemusí být infrastruktura umístěná lokálně v jednotlivých vývojových centrech, je tato infrastruktura pouze v jednom datovém centru v Praze.

Druhým zdrojem výpočetních zdrojů, který je společností využíván, je komerční cloudová služba AWS od společnosti Amazon. Společností bylo rozhodnuto, že tento prostor a výpočetní výkon je určen pouze pro projekty a produkty zákazníků a zdroje nebudou využívány pro vývoj a testování z důvodu vysokých provozních nákladů, které odpovídají skoro třinásovkou provozních nákladů on-premise infrastruktury. Zároveň tam není možné migrovat většinu podpůrných systémů, protože takové nasazení není technicky možné a podporované výrobci těchto systémů. AWS se tedy využívá pouze pro preprodukční a produkční prostředí produktů určených pro cloud a přímo využívané zákazníky.

Následující část práce se zabývá pouze analýzou on-premise částí, která je primárně využívána interními týmy R&D⁹.

4.2.1 Charakteristika prostředí virtualizace

Prostředí a platforma virtualizace je tvořena technologiemi společnosti VMware, konkrétně produkty VMware vSphere ESXi a VMware vSphere vCenter Server. Prostředí hostuje přes 850 virtuálních strojů, z toho přes 550 je spojených s vývojem a testováním, ostatní jsou určeny pro jiná oddělení nebo jako produkční virtuální servery pro provozní systémy. Těchto 550 virtuálních strojů má svůj dedikovaný výpočetní výkon v podobě čtrnácti serverů (více informací v další části).

⁹ Research and Development – v prostředí vybrané společnosti se jedná o týmy vývojářů a testerů

Mandatorní část virtuálních strojů určených pro R&D, přes 90 %, používá operační systém CentOS Linux 7, který je zároveň použit jako operační systém pro vyvíjené produkty, nebo jeho odnože založené na edici Red Hat Enterprise Linux, např. Fedora. Zbýlých 10 % virtuálních strojů představují jiné verze systému Linux nebo Windows. S vydáním systému CentOS Linux 8 se plánuje na tento systém přejít ve velmi blízké budoucnosti.

Využití zdrojů, které jsou dostupné, lze vidět na následujících grafech.

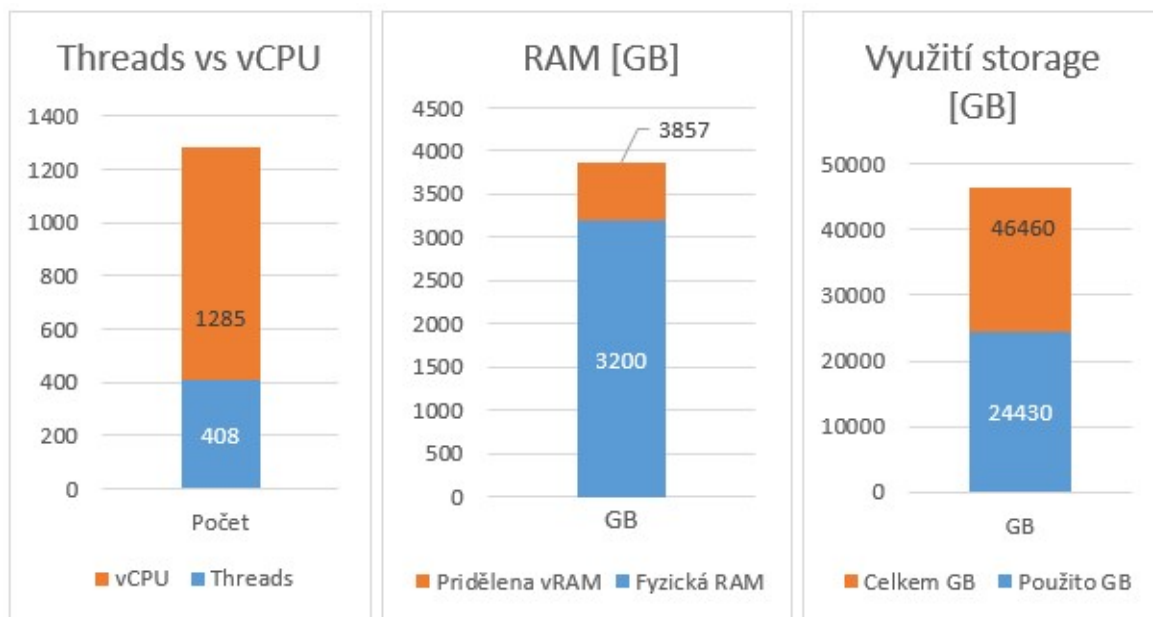


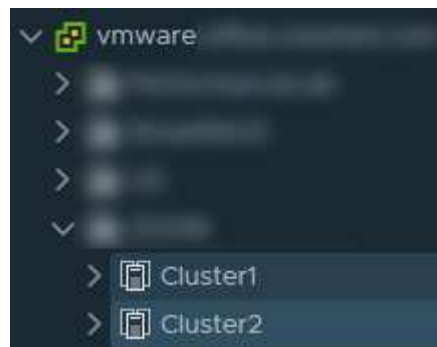
Diagram 1 - Využití zdrojů (zdroj: vlastní)

Z grafů na diagramu 1 je vidět, že růst na stávající infrastruktuře už není možný, primárně z důvodu, že RAM přidělená jednotlivým virtuálním strojům přesáhla fyzickou kapacitu. Pro chod to prozatím nemá vliv, protože se jedná pouze o přidělenou RAM a reálná obsazenost fyzické RAM se pohybuje kolem 75 %. Při dalším růstu počtu virtuálních strojů však začnou jednotlivé systémy RAM swapovat a vytěžovat tím disková pole. Dále je vidět, že koeficient pro CPU overcommitment je 3,14 (poměr 1285:408). Dle best practices definované společností VMware, koeficient vyšší, než tři může představovat potenciální problém a degradaci výkonu, ovšem v tomto případě se vytížení průměrně pohybuje na úrovni 65 %. Vzniká potřeba monitorovat zdroje v průběhu špiček, ale výkon to aktuálně v typickém provozu neomezuje. VMware dále definoval obecné doporučení udržet metriku CPU Ready na 5 % nebo méně. (31) (32)

Typické dopady na výkon při různém koeficientu CPU overcommitment (vCPU : CPU):

- Od 1:1 do 3:1 – typicky nepředstavuje problém a nemá vliv na výkon
- Od 3:1 do 5:1 – občasně viditelná degradace výkonu
- Od 6:1 a více – velmi častá degradace výkonu

Jak již bylo zmíněno, pro management všech virtualizačních serverů (hostů) se využívá vCenter Server 6.7. Z pohledu jednoduchosti správy a nákladů, je při tak velkém počtu VM nepraktické používat administrační rozhraní na jednotlivých serverech. Prostředí vCenter serveru je logicky rozděleno do tzv. clusterů podle typu CPU, tak aby bylo dosaženo co



Obrázek 20 - Clustery (zdroj: vlastní)

největší kompatibility a zároveň nebyl výkon těchto serverů limitován nastavením Enhanced vMotion (EVC) a maskováním instrukcí novějších CPU. Oba clustery používají technologii vSphere HA a vSphere DRS. vSphere HA zabezpečuje vysokou dostupnost v případě pádu jednoho nebo více serverů a zajistí, že se všechny VM ovlivněné pádem serveru spustí na jiném. vSphere DRS zajišťuje správu zdrojů a prioritizuje využití těchto zdrojů.

Ověření (autentizaci) zajišťuje propojení s podnikovou doménou Active Directory. Z důvodu, že se nevyužívá Integrated Windows Authentication, SAML, OAuth a podobně, není řešení plnohodnotné SSO, uživatelé se musí přihlašovat pokaždé znova.

Uživatelé mají v prostředí virtualizace přesně definovaná práva toho, co mohou vykonávat. Společnost preferuje model „implicitně zakázáno“ a proto jsou možnosti uživatelů omezena pouze na nezbytně nutné činnosti související s jejich denními úkoly (například Zapnutí VM, Vypnutí VM, vytvoření snapshotů, atd.). Více o právech v části charakteristiky procesů.

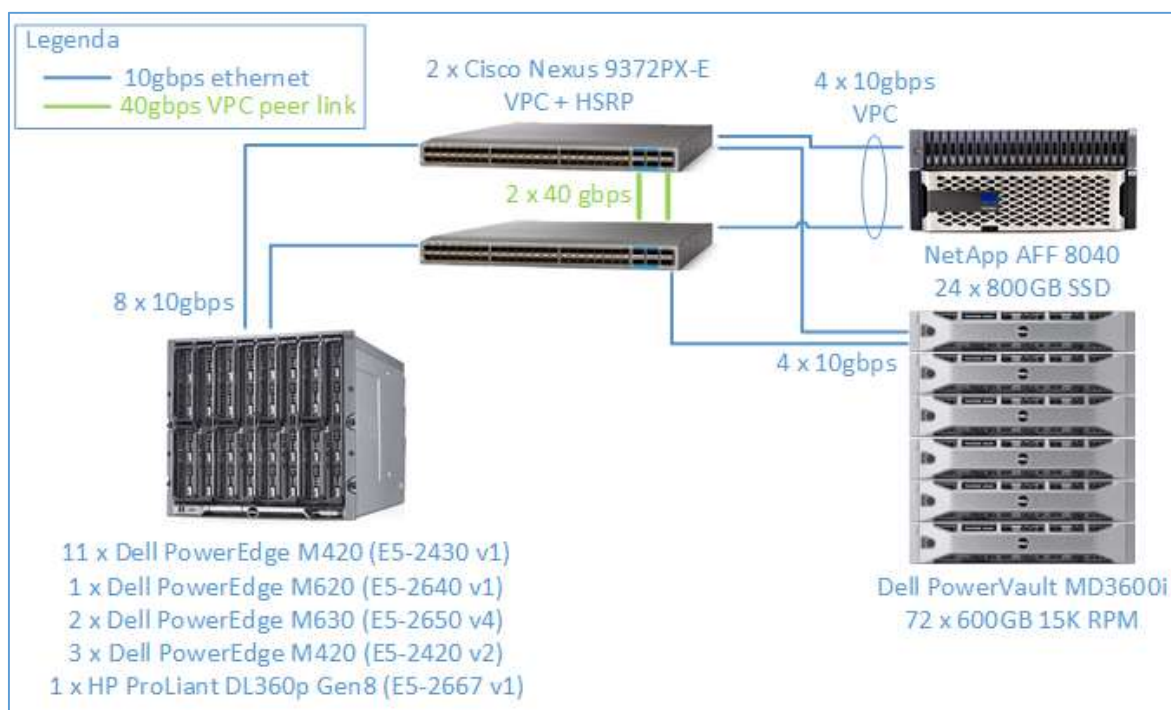
4.2.2 Charakteristika hardware a software

Infrastrukturu společnost buduje již několik let. Její upgrade probíhá v pravidelných intervalech dvou až tří let dle potřeb oddělení R&D a požadavků na zdroje vyvíjeného software. Následující high-level schéma infrastruktury zachycuje její část relevantní pro výše zmíněných 550 VM a část produkční infrastruktury, která se v rámci implementace stane vývojovou z důvodu výměny produkčních serverů za nové.

Serverový hardware

Infrastruktura před implementací automatizace a vRA obsahovala:

- 11 x server Dell PowerEdge M420 (CPU Intel Xeon E5-2430 v1, 96 GB RAM) (vývoj)
- 1 x server Dell PowerEdge M620 (CPU Intel Xeon E5-2620 v1, 384 GB RAM) (vývoj)
- 2 x server Dell PowerEdge M630 (CPU Intel Xeon E5-22650 v6, 768 GB RAM) (vývoj)
- 3 x server Dell PowerEdge M420 (CPU Intel Xeon E5-2420 v2, 96 GB RAM) (produkce)
- 1 x server HP ProLiant DL360p Gen8 (CPU Intel Xeon E5-2667 v1, 224 GB RAM) (vývoj)
- Storage NetApp AFF8040 (24 x 800 GB SSD)
- Storage Dell PowerVault 3600i (72 x 600 GB 15k RPM)
- 2 x core switche Cisco Nexus 9372PX-E



Obrázek 21 - High-level infrastruktura před implementací (zdroj: vlastní)

Servery označené jako „produkce“ tedy byly použité primárně pro služby podporující běh společnosti jako je Active Directory, VMware vCenter atd. Switche a storage jsou zde uvedené pouze pro doplnění povědomí o infrastruktuře, budou krátce popsány, ale nejsou součástí praktické části, protože se jejich konfigurace nemění.

Všechny servery, kromě serveru HP, jsou umístěné v rámci Blade šasi Dell PowerEdge M1000e. To poskytuje unifikované řešení, které šetří místo, spotřebu a síťovou konektivitu. Na všech výše zmíněných serverech je povolený hyperthreading a poskytují 408

threadů (logické CPU) (204 fyzických jader). Objem paměti RAM je 3200 GB, což je bohužel vzhledem k nárokům vyvíjeného software pro další růst nedostačující.

Síťovou konektivitu zabezpečují karty QLogic 57810, které mají 2 porty s 10gbps propustností pro zajištění vysoké dostupnosti a přenosové kapacity. V serverech M420 se nachází vždy pouze jedna tato karta, protože servery nemají fyzické místo pro další. Představuje to potenciální problém „single point of failure“ a vysokou dostupnost zde představují pouze dvě nezávislé cesty ke switchům. Ostatní servery mají tyto karty dvě, proto servery neztratí konektivitu ani v případě poruchy na jedné z nich. Karty jsou v případě blade serverů připojené na interní backplane šasi. Konektivitu blade šasi s externím světem zajišťují interní switche, které jsou umístěné na zadní straně tohoto šasi.

Jak je vidět ze schématu, všechny servery využívají externího úložiště. Na tyto disková pole se ukládají veškeré virtuální stroje. Proto zmíněné servery obsahují pouze dva malé SSD disky v konfiguraci RAID 1, která zajišťuje základní úroveň ochrany dat pomocí zrcadlení obsahu. Na tyto disky se instaluje hypervizor.

Hypervizor zajišťující běh celé platformy je VMware vSphere ESXi 6.7 a spravovaný výše zmíněným vSphere vCenter Server.

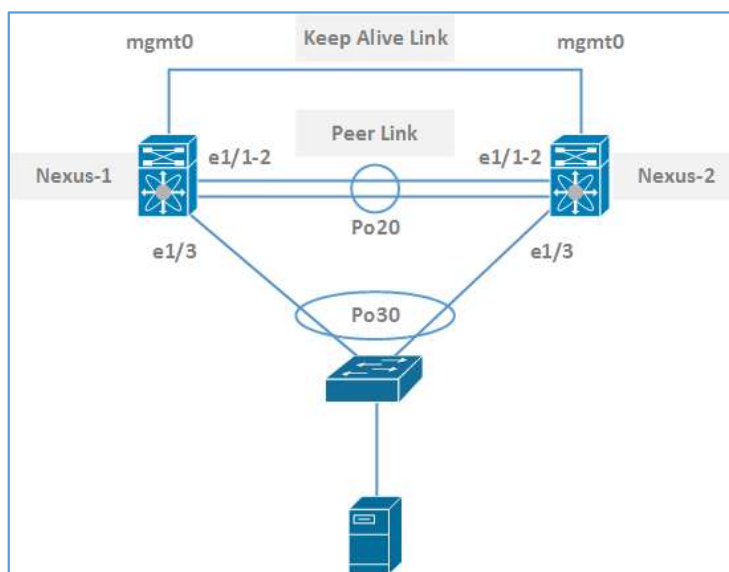
Switche a LAN¹⁰ síť

Páteřní LAN síť tvoří pár L3¹¹ switchů typu Cisco Nexu 9732PX-E, které disponují 48 porty o rychlosti 10gbps a šesti porty o rychlosti 40gbps a s velmi nízkou latencí méně než dvě mikrosekundy. Tento pár vytváří redundanci na páteřní síti, protože veškeré další switche, nebo disková pole, jsou připojena do obou switchů zároveň. Je to možné díky technologii Virtual Port Channel (vPC) a Hot Standby Router Protocol (HSRP). Switche jsou navzájem propojeny skrze dva 40gbps porty a tento propoj se v terminologii VPC nazývá peer link. První zmíněná technologie vPC umožňuje vytvořit Port Channel (Ether Channel) propoj z jednoho switchu, nebo jiného zařízení, do dvou core switchů a vytvořit tak redundantní spojení. Druhá technologie HSRP je proprietární protokol firmy Cisco. Zjednodušeně se dá charakterizovat jako sdílení L3 gateway adresy mezi dvěma routry

¹⁰ Local Area Network

¹¹ Layer 3 dle OSI modelu

v módu Active/Standby. V případě pádu aktivního routeru, převezme L3 adresu záložní a zamezí nedostupnosti služeb, z důvodu nefungujícího routování.



Obrázek 22 - Příklad vPC (zdroj: <https://www.letsconfig.com/how-to-configure-cisco-nexus-vpc/>)

Samozřejmostí je využívání VLAN z důvodu bezpečnosti a logického rozdělení sítě. VLAN určené pro konektivitu diskových polí nejsou routované, tudíž se není možné na toto diskové pole dostat jinak než z konkrétní VLAN. Do těchto VLAN jsou striktně připojeny pouze VMkernel adaptéry jednotlivých hypervizorů, uživatelé k nim nemají žádný přístup.

Disková pole

Infrastruktura disponuje dvěma diskovými poli o celkové použitelné kapacitě necelých 46 TB. Tato disková pole jsou určena pouze pro VM R&D oddělení. S obsazeností kolem 55 % zde lze najít i místo pro další růst. Obě disková pole využívají standardní Ethernet připojení, odpadá nutnost pořizovat SAN¹² infrastrukturu. První diskové pole Dell PowerVault 3600i využívá protokolu iSCSI a druhé pole NetApp AFF 8040 protokolu NFS¹³. Diskové pole Dell je starší, tudíž nedisponuje žádnými moderními technologiemi, kromě konektivity 4x10gbps. Představuje tradiční formu diskového pole SAN v porovnání s moderními diskovými poli označovanými jako software-defined storage, která se dostávají do popředí s masivní produkcí SSD disků a nabízí funkce jako například datovou

¹² Storage Area Network

¹³ Network File System

deduplikaci. Pole využívá technologii RAID¹⁴ 5 pro nižší míru redundance dat a vyšší rychlost čtení a jeho použitelná kapacita je 33TB.

Druhé zmiňované pole NetApp patří do skupiny software-defined storage. I přes jeho menší kapacitu, pouze bezmála 13 TB, je zde využito výhod deduplikace a komprese. Při stávající struktuře a typu dat na něm uložených disponuje koeficientem efektivity (poměr potenciálně uložených dat a fyzické kapacity) na hodnotě 2,5, tedy umožní uložit až 35 TB dat. Tento koeficient se mění s typem dat a není možné ho brát jako pevně daný. Nelze tedy s touto kapacitou počítat na 100 %. NetApp využívá své vlastní technologie RAID-DP pro redundanci dat. Toto diskové pole je připojeno protokolem NFS v3 a v4.1. Hlavním důvodem pro využití tohoto protokolu je podpora reportingu reálně využitého místa po deduplikaci zpátky do prostředí VMware. Připojení je opět 4x10gbps. (33)

4.2.3 Charakteristika procesů

Správu celé infrastruktury má na starosti IT oddělení, které je plně odpovědné za stav hardwaru a softwaru, jeho upgrady, konfiguraci, administraci, kontrolu dostupných zdrojů, plánování a návrh architektury. Jde o celou řadu činností, které jsou díky správně nastaveným procesům řádně evidované v systému správy požadavků (issue tracking). Na druhou stranu, všechny tyto procesy představují byrokratickou zátěž, která je bohužel častým jevem ve větších společnostech a korporacích. Dalším negativním jevem je prodlužování doby odezvy a vyřešení požadavků. I přes správně nastavené a vyhodnocené priority jednotlivých úkolů se může stát, že jednoduchý úkol bude ve frontě déle, než je standard. Na druhou stranu tato byrokracie přispívá k větší kontrole a bezpečnosti dané infrastruktury, primárně z důvodu, že bez řádně zadaného a evidovaného požadavku nesmí IT oddělení provést změnu.

V rámci řešené virtualizační platformy se dělí odpovědnost IT na dvě větve. V případě osobních virtuálních strojů jednotlivých zaměstnanců končí odpovědnost IT vytvořením VM a přidělením IP adres. Vše od instalace operačního systému je v odpovědnosti daného vývojáře, nebo testera. Zároveň ale existují případy, kdy IT oddělení provádí zásahy i do operačního systému na těchto VM a pomáhají zaměstnancům R&D vyřešit problémy týkající se sítě, nebo jiných aspektů nastavení operačního systému.

¹⁴ Redundant Array of Independent Disks

Naopak v případě podpůrných systémů odpovědnost IT není limitována a sahá až po správu konkrétních aplikací instalovaných v dané VM.

Odpovědnost je daná zavedenými procesy a nastavením oprávnění uživatelů. Společnost preferuje politiku „default deny“, tedy implicitně žádné oprávnění, pokud není potřeba. To pomáhá zajistit určitý stupeň bezpečnosti. Společně s omezeností výpočetních zdrojů jejich kontrolou jsou běžní uživatelé omezeni pouze na činnosti související s jejich denními úkoly. Uživatelé nemají oprávnění vytvářet virtuální stroje, jakkoli manipulovat s výpočetními zdroji, nebo je mazat, přidělovat IP adresy, nebo je měnit. Oprávnění se uživatelům přidělují dle oddělení na konkrétní složky ve vCenter Server prostředí. Zde znova platí, že práva jsou přidělena pouze pro složky, které opravdu potřebují.

Seznam standardních oprávnění:

- Zapnout / Vypnout / Resetovat / Restartovat VM
- Vytvořit / Smazat / Revert snapshotů
- Připojit CD/DVD nebo ISO soubor z lokálního disku nebo síťového úložiště
- Spustit VM konzoli
- Instalace VMware Tools

Z nastavení procesů, odpovědností a oprávnění pro IT oddělení a uživatele se odvíjí i struktura a složitost uživatelských požadavků směřujících k IT oddělení. Složitost požadavků začíná od nejjednoduššího vytvoření VM, změny již existující, až po komplexní konfigurace, které napodobují prostředí zákazníka, nebo nasazení nových podpůrných systémů, jako například Cisco Unified Communications a systému provozovaných v call centrech, s kterými se software společnosti integruje. Nejjednodušší požadavky tvoří až 90 % všech typů požadavků, a to zejména proto, že atypických scénářů u zákazníku je minimum a z větší části se zákazníci řídí best practice. Nasazování nových verzí podpůrných systémů tudíž není tak časté a většinou probíhá jednou za půl roku, když vychází nové minoritní verze. Zároveň platí, že tyto podpůrné systémy stačí jednou nainstalovat a pokud nenastane žádný pád infrastruktury, tak do nich není potřeba zasahovat.

Níže jsou uvedeny nejtypičtější žádosti, které musí IT oddělení řešit (z kategorie jednodušších):

- Vytvoření osobní/statické VM
- Navýšení HW zdrojů pro stávající VM
- Vytvoření nestandardní nebo větší HW konfigurace
- Vytvoření prostředí podobné konfigurace jak u zákazníka
- Potřeba testovat různé verze a konfigurace produktu jak z důvodu stability, nebo výkonnosti tak z důvodu ověření, že je možné provést upgrade produktu

Všechny tyto scénáře se dají generalizovat dvěma základními úkony – vytvoření a změna VM. Další část práce se tedy bude zabývat automatizací těchto dvou generalizovaných jednoduchých úkolů, vytvoření VM nebo její změna, a tím zjednodušení procesů a byrokracie pro koncové uživatele z oddělení R&D. Dá se říct, že nezáleží na tom, kolik VM je potřeba vytvořit nebo změnit, workflow bude stejný, pouze čas na provedení delší. Složitější úkoly nebo instalaci podpůrných systému nelze automatizovat, protože ve většině případů to není podporované a není součástí analýzy, ani rozsahu této práce.

Proces vytvoření nové VM

Proces lze rozdělit na dvě části, a to vytvoření VM bez OS a včetně instalace OS. Obě části zahrnují přidělení IP adresy. Workflow níže charakterizuje jednotlivé úkony dvou aktů, uživatele a IT oddělení, před implementací automatizace.

Kroky:

- Uživatelé vytvoří požadavek v systému pro evidenci požadavků
- Systém evidence na základě předdefinovaných pravidel automaticky přidělí požadavek členovi IT oddělení (dále jen IT) a notifikuje zadavatele o úspěšném vytvoření a přiřazení
- IT provede validaci požadavku na jeho formální úplnost
- V případě neúplné žádosti, nebo chybějících detailů, tento požadavek vrátí zadavateli k opravě a doplnění
- Uživatel doplní chybějící nebo nekompletní údaje a vrátí IT k řešení
- Po úspěšné validaci požadavku IT provede kontrolu dostupnosti HW zdrojů (CPU, RAM, HDD)
- V případě nedostatku HW zdrojů je požadavek zamítnut a uzavřen

- Společně s uzavřením se spouští vedlejší proces možnosti reklamace požadavku uživatelem
- Pokud je HW zdrojů dostatek, IT vyhledá dle předem definovaného schématu hostname a IP adresu (nic se nevytváří, vše je předem definované architektem dle případů užití)
- Pokud požadavek nepožaduje instalaci OS, IT pouze vytvoří VM
- Pokud je OS vyžadován, IT naklonuje předem připravenou šablonu s už nainstalovaným OS, a upraví v ní IP adresu
- Provedené změny se zdokumentují v dokumentačním systému
- Požadavek je v systému uzavřen. Systém automaticky notifikuje uživatele do emailu včetně komentáře přidaného IT
 - Společně s uzavřením se spouští vedlejší proces možnosti reklamace požadavku uživatelem

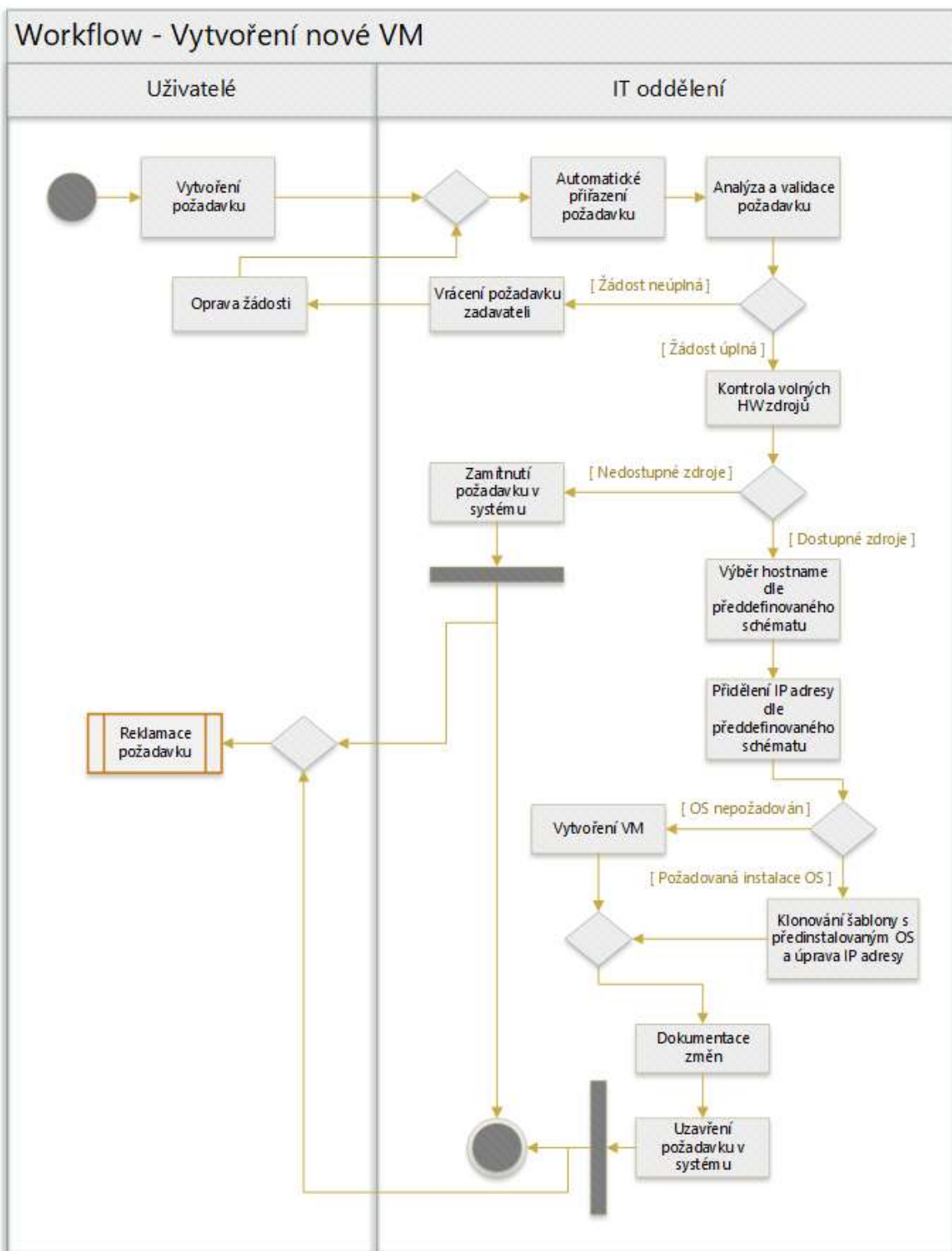


Diagram 2 - Proces vytvoření nové VM (zdroj: vlastní)

Proces změny stávající VM

Proces je téměř podobný vytvoření VM, odpadají však části související s výběrem hostname a IP adresy a klonování. Workflow níže charakterizuje jednotlivé úkony dvou aktorů, uživatele a IT oddělení, před implementací automatizace.

Kroky:

- Uživatelé vytvoří požadavek v systému pro evidenci požadavků
- Systém evidence na základě předdefinovaných pravidel automaticky přidělí požadavek členovi IT oddělení (dále jen IT) a notifikuje zadavatele o úspěšném vytvoření a přiřazení
- IT provede validaci požadavku na jeho formální úplnost
- V případě neúplné žádosti nebo chybějících detailů tento požadavek vrátí zadavateli k opravě a doplnění
- Uživatel doplní chybějící nebo nekompletní údaje a vrátí IT k řešení
- Po úspěšné validaci požadavku IT provede kontrolu dostupnosti HW zdrojů (CPU, RAM, HDD)
- V případě nedostatku HW zdrojů je požadavek zamítnut a uzavřen
 - o Společně s uzavřením se spouští vedlejší proces možnosti reklamace požadavku uživatelem
- Pokud je HW zdrojů dostatek, IT provede požadovanou změnu a navýší dané VM zdroje (vCPU, vRAM, HDD)
- Provedené změny se zdokumentují v dokumentačním systému
- Požadavek je v systému uzavřen. Systém automaticky notifikuje uživatele do emailu včetně komentáře přidaného IT
 - o Společně s uzavřením se spouští vedlejší proces možnosti reklamace požadavku uživatelem

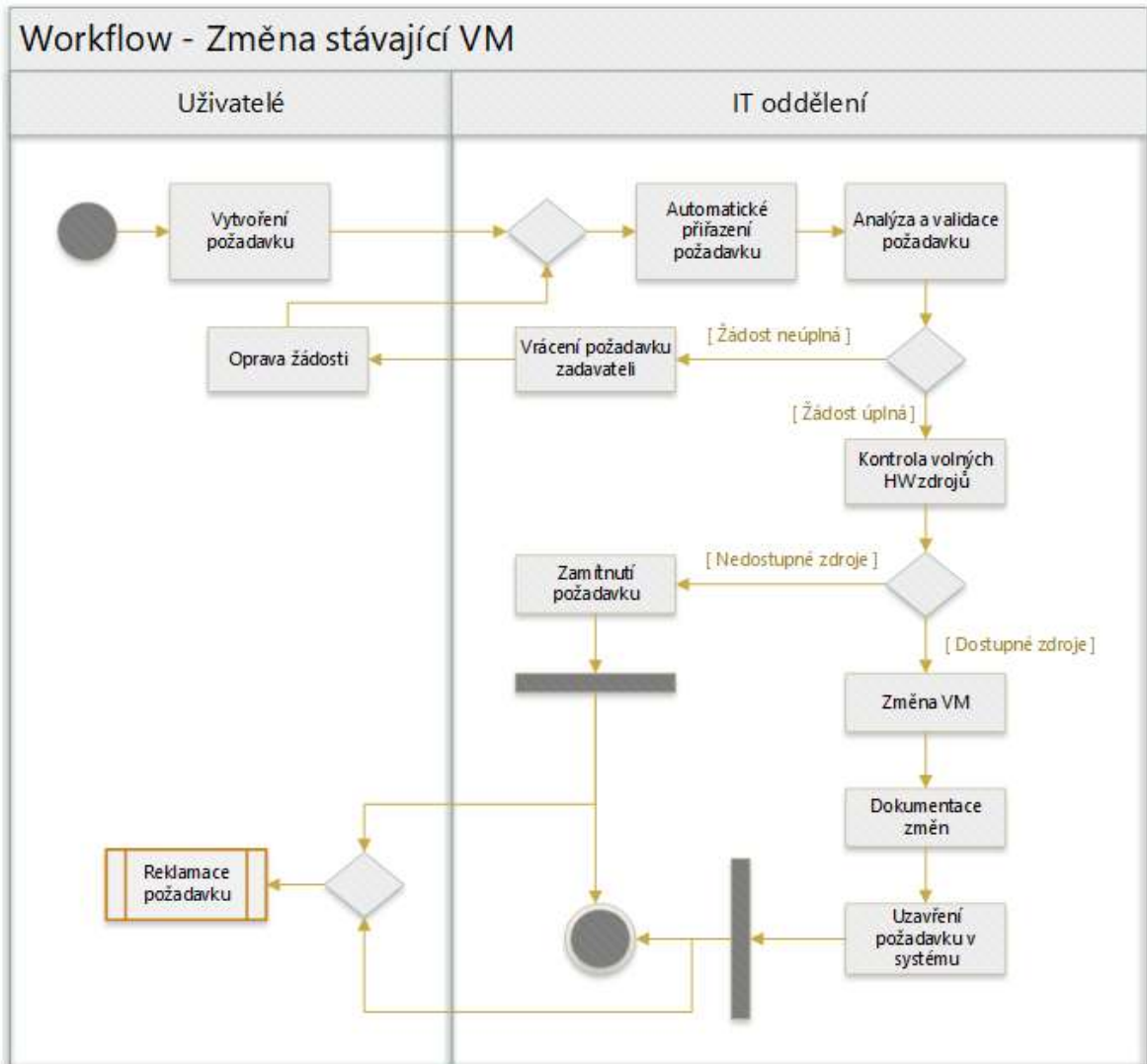


Diagram 3 - Proces změny stávající VM (zdroj: vlastní)

4.2.4 Analýza a kalkulace provozních nákladů před automatizací

Než bude přistoupeno k návrhu nové infrastruktury a optimalizaci procesů pomocí automatizace, je nutné vyčíslit provozní náklady na čas zaměstnanců spojen s úkoly a prostoji spojenými s virtualizací.

Pro výpočet jsou použita data ze systému pro evidenci požadavků, konkrétně suma množství času od založení požadavku po jeho vyřešení za měsíc. Ze systému bylo zjištěno, že IT oddělení věnuje 10 % pracovních hodin dvou lidí měsíčně do správy prostředí VMware (včetně tzv. skrytého času, který je spojen s jinými typy úkolů, které vzejdou z původních požadavků), tedy $16 \text{ hodin/člověk/měsíc} * 2 = 32 \text{ hodin/měsíc}$ (uvažujeme, že typický měsíc = 20 pracovních dní = 160 pracovních hodin).

Provozní náklady jsou ale tvořeny i náklady na zaměstnance z oddělení v případě jejich potenciálních prostojů v době, kdy IT oddělení řeší jejich požadavek. Relevantní údaj je velmi těžké zjistit, proto je výpočet proveden na dvou případech, tzv. „best case“ (nejlepší scénář) a „worst case“ (nejhorší scénář). Jako „best case“ uvažujeme situaci, kdy členové R&D týmů nemají žádné prostoje, protože v době řešení požadavků IT oddělení mohou řešit jiné úkoly a nezáleží na době zpracování požadavku. Naopak „worst case“ uvažujeme situaci, kdy členové týmů čekají na IT oddělení a jejich prostoje tvoří 80 % z času stráveného IT oddělením na vyřešení požadavků, tedy $25,6 \text{ hodiny/měsíc}$. Ze systému správy požadavků bylo zjištěno, že na IT oddělení nečeká více než dva týmy současně a z těch týmu pouze 50 % členů. Společnost definuje tým jako čtyřčlenný, z toho jsou tři členové vývojáři a jeden tester.

Bohužel společnost neumožnila zveřejnit náklady na mzdy zaměstnanců, proto byl pro výpočet použit materiál s názvem „Mzdový průzkum 2019 – Trendy na pracovním trhu v České republice“ od společnosti Hays Czech Republic s.r.o. (34), v němž jsou uvedeny typické mzdy v oboru IT. Tento průzkum se provádí každý rok. Z uvedeného průzkumu je využit údaj označen jako „Typical“. Následující tabulka je výtahem údajů použitých pro výpočet (údaje v CZK/měsíc).

Role		Min	Max	Typical
IT	Administrator / Support Engineer	40 000	65 000	60 000
IT	IT Manager	50 000	110 000	80 000
R&D	Developer / Programmer – Java	60 000	130 000	120 000
R&D	Test Analyst	35 000	80 000	65 000

Tabulka 6 - Mzdový přehled - IT a Telekomunikace (zdroj: https://cloud.email.hays.com/cz_salary_guide)

Pro výpočet mzdových provozních nákladů je použit následující vzorec:

$$\frac{\sum \text{mzda členů IT}}{\text{počet členů} * 160} * p_{hod} + 2 * \frac{\sum \text{mzda členů jednoho týmu R\&D}}{0,5 * \text{počet členů} * 160} * (0,8 * p_{hod})$$

kde:

- p_{hod} – počet hodin strávených prací na požadavcích

Následující tabulka uvádí výsledky mzdových provozních nákladů za měsíc a tři roky (společnost počítá náklady na tři roky).

Typ	Náklad IT (CZK / měsíc)	Náklad R&D (CZK / měsíc)	Náklad za 3 roky (CZK)
Best case (nejlepší scénář) (bez prostojů)	14 000	0	504 000
Průměrný scénář	14 000	34 000	1 728 000
Worst case (nejhorší scénář) (prostoje dvou R&D týmů v délce 80 % času potřebného pro IT oddělení)	14 000	68 000	2 952 000

Tabulka 7 - Přehled nákladů měsíčně a sumárně za 3 roky (zdroj: vlastní)

Výpočet nákladů uvedený výše zcela jistě nepokrývá všechny relevantní náklady, ale slouží spíše jako záchytný bod uvádějící minimální náklady spojené s provozem. Pro výpočet veškerých souvisejících nákladů by bylo potřeba provést hlubší analýzu všech činností prováděných oběma odděleními, včetně výpočtu veškerých skrytých nákladů, které ještě nebyly zahrnuty. To ale daleko překračuje rozsah práce.

4.3 Návrh řešení automatizace

4.3.1 Požadavky společnosti

Požadavky na zefektivnění každodenních úkolů vývojářů a testerů pomocí automatizace, byly předem dané a vznikly na základě zpětné vazby, osobních konzultací s jednotlivými vedoucími týmů, nebo týmy samotnými v rámci řešení každodenních úkolů, ještě před vznikem této práce. V práci tedy nebylo provedeno žádné dotazníkové, nebo panelové šetření s konkrétním dotazníkem, ani konkrétní skupinou lidí.

Požadavky:

- Nativní podpora on-premise VMware vSphere infrastruktury
 - o Volitelně podpora AWS, Azure a jiných providerů, kde by bylo možno v budoucnu automatizovat procesy a využívat jejich výpočetní kapacity
- Musí obsahovat možnost omezit maximální počet zdrojů, které lze použít koncovými uživateli (CPU, RAM, HDD)
- Integrace s Windows Server DHCP (vytváření a aktualizace DHCP rezervací)
- Integrace s Windows Server DNS (vytváření a aktualizace DNS záznamů)
- Podpora Workflow pro nasazování včetně možnosti skriptování
- Podpora schvalování žádostí
- Podpora šablon pro různé OS a HW konfigurace (obdobu veřejného cloudu, například Linux – Malá VM – 1 vCPU a 2 GB RAM, nebo Linux – Velká VM – 4 vCPU a 16 GB RAM)
- Definování životnosti VM ve dnech
 - o Po expiraci se VM musí sama vypnout a smazat, nebo zůstat po nějakou dobu v archivu a pak smazat
 - o Musí být možnost prodloužit životnost automaticky i na základě schválení
- Nasazování VM nesmí použít žádné jiné zdroje než definované administrátorem, v kontextu sítě a úložiště
- Autentizace a autorizace integrované s Active Directory
- Samoobslužný portál pro koncové uživatele
 - o Možnost nasazení šablon nebo celých prostředí
 - o Přehled o všech nasazených prostředcích
- REST API pro externí integrace

- Podpora linkovaných klonů ve VMware
- Integrace s Microsoft SQL Server databází, která může být použita při workflow skriptování
- Cena řešení není důležitá, primárním kritériem je funkčnost

4.3.2 Hodnotící kritéria při výběru řešení

Pro hodnocení variant jsou jako hodnotící kritéria použity požadavky uvedené výše a z nich je sestavena hodnotící tabulka. Pro hodnocení bude použita metoda bodování jednotlivých kritérií. Bodové hodnocení bude nabývat hodnot jedna až pět následovně:

- 1 – zcela nepodporuje požadavek / zcela nevyhovující funkcionality / dosažení cíle není možné
- 2 – jen velmi omezená podpora požadavku a funkcionality / není nativní podpora, ale dosažení cíle je obtížné
- 3 – podporuje požadavek ale pouze 50 % funkcionality / není nativní podpora, ale je možné dosáhnout cíle
- 4 – podporuje požadavek z větší části / vyhovující funkcionality z větší části / není nativní podpora, ale je jednoduché dosáhnout cíle
- 5 – zcela podporuje požadavek / zcela vyhovující funkcionality / nativní podpora

Varianta s nejvyšším počtem dosažených bodů bude zvolena jako vítězná a bude použita pro implementaci.

4.3.3 Vícekriteriální hodnocení variant

Požadavek	VMware vRealize Automation	Terraform	OpenStack
Nativní podpora on-premise VMware vSphere infrastruktury	5	5	5
Podpora AWS, Azure	5	5	5
Omezení maximálních zdrojů pro uživatele	5	3	3
Integrace s Windows Server DHCP	4	1	2
Integrace s Windows Server DNS	4	3	3

Podpora Workflow pro nasazování včetně možnosti skriptování	5	1	3
Podpora schvalování žádostí	5	2	2
Podpora šablon pro různé OS a HW konfigurace	5	2	3
Definování životnosti VM ve dnech	5	1	4
Nasazování musí používat pouze administrátorem definované zdroje	5	2	2
Autentizace a autorizace přes Active Directory	5	1	5
Samoobslužný portál	5	1	5
Rest API	5	1	5
Podpora linkovaný klonu ve VMware	5	5	5
Podpora SQL Serveru při workflow skriptování	5	1	2
Průměr	4,87	2,27	3,6

Tabulka 8 - Vícekriteriální hodnocení variant (zdroj: vlastní)

Na základě provedeného hodnocení bylo rozhodnuto vybrat jako nejvhodnější řešení VMware vRealize Automation.

4.3.4 Návrh nového prostředí a architektury

Návrh nové architektury vychází ze situace charakterizované v části 4.2.2 a zároveň byl připravován tak, aby vyžadoval co nejméně změn, ale s přínosem co nejvíce funkcí pro koncové uživatele.

4.3.4.1 Změny původní infrastruktury

Serverový hardware

Pro automatizaci budou nově použity servery v původním stavu označené jako „produkční“ (tedy 3 x Dell PowerEdge M40 s CPU Intel Xeon E5-2420 v2), protože jejich obsah byl migrován na nový hardware, jenž byl dodán v rámci jiného projektu. Tyto servery poskytnou dostatečný prostor a zdroje pro počáteční fázi implementace. V budoucnu se dle využitých zdrojů a potřeb bude serverový hardware rozšiřovat. Snahou automatizace je eliminovat ve větší míře co nejvíce virtuálních strojů, které jsou statické, a využívat virtuální stroje generované skrze VMware vRealize Automation. Schéma architektury se nemění a schéma na obrázku č. 21 je nadále platná.

Switche a LAN síť

Všechny nově vytvořené virtuální stroje pomocí automatizace vRA, budou využívat nově vytvořené VLAN a s ní související rozsah IP adres a zónu DNS. Vše z důvodu bezpečnosti, přehlednosti a jednoduchosti pro koncové uživatele. Virtuální stroje vytvořené v nové VLAN budou využívat DHCP a DNS serverů na systému Microsoft Windows Server 2019. DHCP server bude obsahovat rezervace pro každou použitelnou IP adresu z daného rozsahu pro předem definované MAC adresy (nebo DUID a IAID pro IPv6). Virtuální stroje budou moci získat IP adresu automaticky nebo staticky, na základě předem definovaného schématu. Více o přiřazování skrze DHCP nebo statické, bude uvedeno v další části této práce. Jak již bylo zmíněno, pár switchů Nexus využívá HSRP, které je definované i pro tuto VLAN a ukázka konfigurace je níže.

Nová VLAN:

- Vlan ID: 564
- IPv4 Network: 10.71.0.0
- IPv4 Subnet: /23
- IPv4 Gateway: 10.71.0.1
- DNS zóna. *.tst.domain.com

Ukázka konfigurace:

Core Switch 1
<pre>interface Vlan564 no shutdown no ip redirects ip address 10.71.0.2/23 ipv6 address 2001:4cc8:fa30:71::2/64 ipv6 nd managed-config-flag ipv6 nd prefix default no-advertise no ipv6 redirects ip access-group acl-vlan-564-in in hsrp version 2 hsrp 564 preempt priority 90 ip 10.71.0.1 hsrp 564 ipv6</pre>

```
preempt
priority 90
ip 2001:4cc8:fa30:71::1
ip dhcp relay address 10.70.10.10
ipv6 dhcp relay address 2001:4cc8:fa30:10::a
```

Core Switch 2

```
interface Vlan564
no shutdown
no ip redirects
ip address 10.71.0.3/23
ipv6 address 2001:4cc8:fa30:71::3/64
ipv6 nd managed-config-flag
ipv6 nd prefix default no-advertise
no ipv6 redirects
ip access-group acl-vlan-564-in in
hsrp version 2
hsrp 564
preempt
priority 80
ip 10.71.0.1
hsrp 564 ipv6
preempt
priority 80
ip 2001:4cc8:fa30:71::1
ip dhcp relay address 10.70.10.10
ipv6 dhcp relay address 2001:4cc8:fa30:10::a
```

Disková pole

Na diskovém poli NetApp AFF 8040 byl vytvořen svazek (volume) o velikosti 1TB a se zapnutou deduplikací a kompresí. Tento svazek bude připojen pouze ke třem výše zmiňovaným serverům pomocí NFS v4.1 protokolu. Kapacita je dostačující, protože typické scénáře nasazení personálních virtuálních strojů nespotřebovaly více než 50 GB na jednu VM. Zároveň je počítáno s deduplikačním poměrem 1:2 (který může být i větší) a tím pádem by se na tento svazek mělo vejít min. 2TB dat, což by mělo zaručit místo pro min. cca 40 virtuálních strojů ve stejný čas.

VMware infrastruktura

VMware infrastruktura musela projít přípravou pro nasazení automatizace vRA. Z výše zmíněných tří nových serverů byl vytvořen cluster tak, aby byly výpočetní zdroje odděleny od stávající infrastruktury. Zároveň to jasně vymezuje výpočetní zdroje, které budou moci být využity automatizací. Pro automatizaci byl v rámci Active Directory vytvořen nový aplikační uživatel, kterému byla dle instalačního manuálu vRA přidělena práva na tento nový cluster tak, aby vRA mohlo provádět veškeré potřebné operace (například vytvářet/měnit/mazat VM). Pro logické uspořádání byla vytvořená složka pro samotné VM vytvořené automatizací a složka pro šablony VM, které budou využity v rámci vRA.



Obrázek 23 - VMware cluster (zdroj: vlastní)

4.3.4.2 Návrh architektury vRealize Automation

Dle teoretické části se nasazení vRA dělí na dva typy – minimální a enterprise. Minimální nasazení nelze využít, protože jej nelze dále rozšiřovat a byla by nutná jeho reinstalace. Enterprise na druhou stranu vyžaduje daleko více serverů a komponent, které se musí instalovat, ale je rozšiřitelný, a hlavně poskytuje vysokou dostupnost. Pro počáteční fázi a plánovanou zátěž implementace bylo ale nutné zvolit kompromis mezi těmito dvěma možnostmi. Architektura bude založena na modelu Enterprise, tedy bude ji možné dále rozšiřovat, ale počet serverů zůstane na minimálním počtu, tedy dvou. Oficiální dokumentace tento typ instalace nezmiňuje, ovšem je podporován a běžně nasazován. Tento typ nasazení byl doporučen dodavatelem software, který jej nasazuje u jiných zákazníků. Nepřímo je zmíněn v rámci konfiguračního průvodce (Obrázek 24 níže), který se spustí po instalaci vRA. Ten po volbě typu Enterprise uvádí, že minimální počet serverů je jeden pro každou roli, ale role IAAS infrastruktury a DEM lze instalovat v rámci jednoho serveru společně. Zároveň zpočátku není vyžadován load balancer a pro oba servery je dostačující pouze doménový záznam, který na ně bude směřovat. Protože zátěž tohoto řešení nebude velká a jedná se o několik jednotek kusů nasazení VM denně, nemá HA řešení opodstatnění. V budoucnu se s ním ale počítá, protože se počítá s až 200 nasazeními denně.

Deployment Type

Choose a deployment type for your installation and whether to install the vRealize Automation infrastructure component, IaaS. Each deployment type has different hardware and configuration requirements.

Minimal deployment
 Enterprise deployment

Enterprise deployments are typically used for production, and allows you to design the topology best suited to your organization's needs.

In the table below are listed the typical machine configurations for a medium-sized high-availability deployment and for a typical distributed deployment. For more information about planning your deployment architecture, see the vRealize Automation Reference Architecture.

Components	High-Availability	Distributed
Deployed vRealize Appliance	2+	1
IaaS Web Server	2+	1
IaaS Manager Service Server	2+	1
IaaS DEM Server	1+	1
Agent Server	1+	1
Microsoft SQL Server (clustered)	1	1

Install Infrastructure as a Service

Infrastructure as a Service (IaaS) allows you to rapidly model and provision servers and desktops across private or public, virtual, or hybrid cloud infrastructure.

[More information about IaaS Installation.](#)

Obrázek 24 - Konfigurační průvodce - Volba typu nasazení vRA (zdroj: vlastní)

Obrázek níže zobrazuje možnost umístit všechny serverové role na jeden server.

Server Roles

Select one or more vRealize Automation components to install on a deployed Windows host.

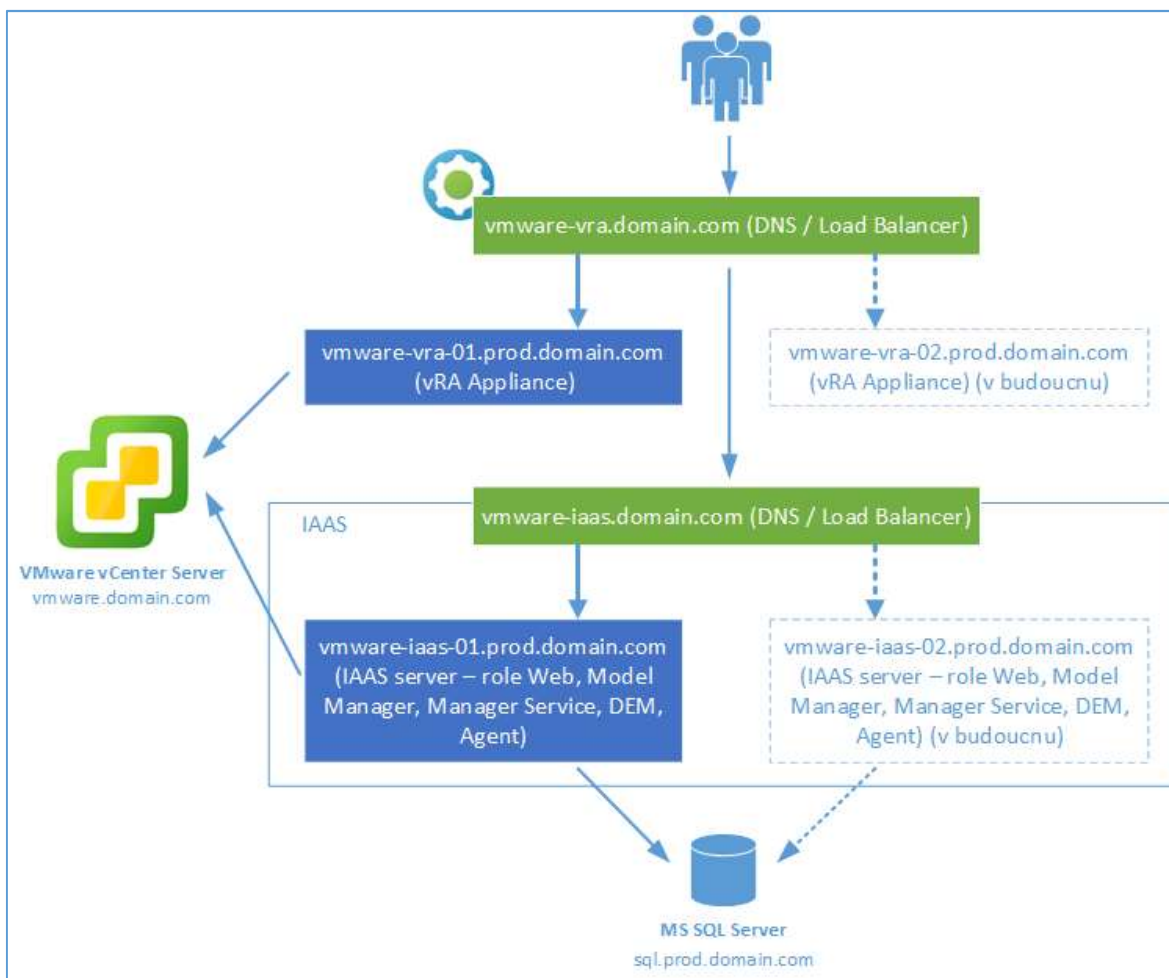
Component	Description
Website	Provides the infrastructure administration and service authoring capabilities to the vRealize Automation console.
Manager Service	The Manager Service coordinates communication between agents, the database, Active Directory and SMTP.
Distributed Execution Manager (DEM)	A Distributed Execution Manager (DEM) executes the business logic of custom models. DEMs also manage cloud machines.
Agent	vRealize Automation uses agents to integrate with external systems. A system administrator can select agents to install to communicate with other virtualization platforms.

Hosts	Initial Web Server and Model Manager	Other Webs	Manager Service	DEM	Agent
VMWARE-IAAS-01	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Obrázek 25 - Konfigurační průvodce - Volba umístění rolí (zdroj: vlastní)

Na základě všech dostupných informací byla navržena doménová jména a IP adresy pro vRA. Níže je uvedeno finální high-level schéma nasazení řešení. vRA je instalováno do stávající sítě obsahující produkční servery a samotný VMware vCenter Server. Nová VLAN definovaná výše, bude použita pouze pro nově vytvořené VM pomocí vRA.

- vRA load-balancer hostname: vmware-vra.domain.com (v tomto případě pouze DNS CNAME směřující na vRA Server 1)
- vRA Server 1 hostname: vmware-vra-01.prod.domain.com
- vRA Server 1 IP adresa: 10.50.50.21
- vRA IAAS load-balancer hostname: vmware-iaas.domain.com (v tomto případě pouze DNS CNAME směřující na vRA Server 1)
- vRA IAAS Server 1 hostname: vmware-iaas-01.prod.domain.com
- vRA IAAS Server 1 IP adresa: 10.50.50.22



Obrázek 26 - High-level architektura vRA (zdroj: vlastní)

4.3.5 Instalace a konfigurace VMware vRealize Automation

Proces instalace je rozdělen na několik částí, které by se daly charakterizovat jako import OVA, příprava IAAS Windows Serveru a konfigurace pomocí průvodce samotného vRA. Samotný proces instalace vychází z oficiální dokumentace, je poměrně zdlouhavý a délkou přesahuje rozsah práce, avšak jsou zde popsány alespoň záchytné body. (41)

Import OVA

Import OVA souboru probíhá rychle a jednoduše. Pomocí průvodce, který poskytuje vCenter, stačí vybrat ova soubor na disku a průvodce se postará o zbylé záležitosti. Hned v prvním kroku je nutné zadat jméno virtuálního stroje a vybrat adresář, ve kterém se bude VM nacházet. V dalším kroku se zvolí umístění v rámci resource pool-u, potvrdí se licenční ujednání, vybere diskové úložiště a síť / portgroup, která se bude používat. Posledním krokem je dle průvodce přizpůsobení konfigurace operačního systému, kde je nutné

definovat heslo pro uživatele root, jestli má, nebo nemá být povolen přístup skrze SSH, hostname, ip adresa a další síťová nastavení.

Deploy OVF Template

1 Select an OVF template

Select an OVF template

Select an OVF template from remote URL or local file system

Enter a URL to download and install the OVF package from the Internet, or browse to a location accessible from your computer, such as a local hard drive, a network share, or a CD/DVD drive.

URL

Local file

Choose Files VMware-vR-App..80_OVF10.ova

VMware-vR-Appliance-7.6.0.317-13027280_OVF10.ova

Obrázek 27 - Výběr souboru OVA (zdroj: vlastní)

Initial root password

This will be used as an initial password for the root user account. You can change the password later (by using the passwd command or from the appliance Web console).

Password

Confirm Password

Enable SSH service in the appliance

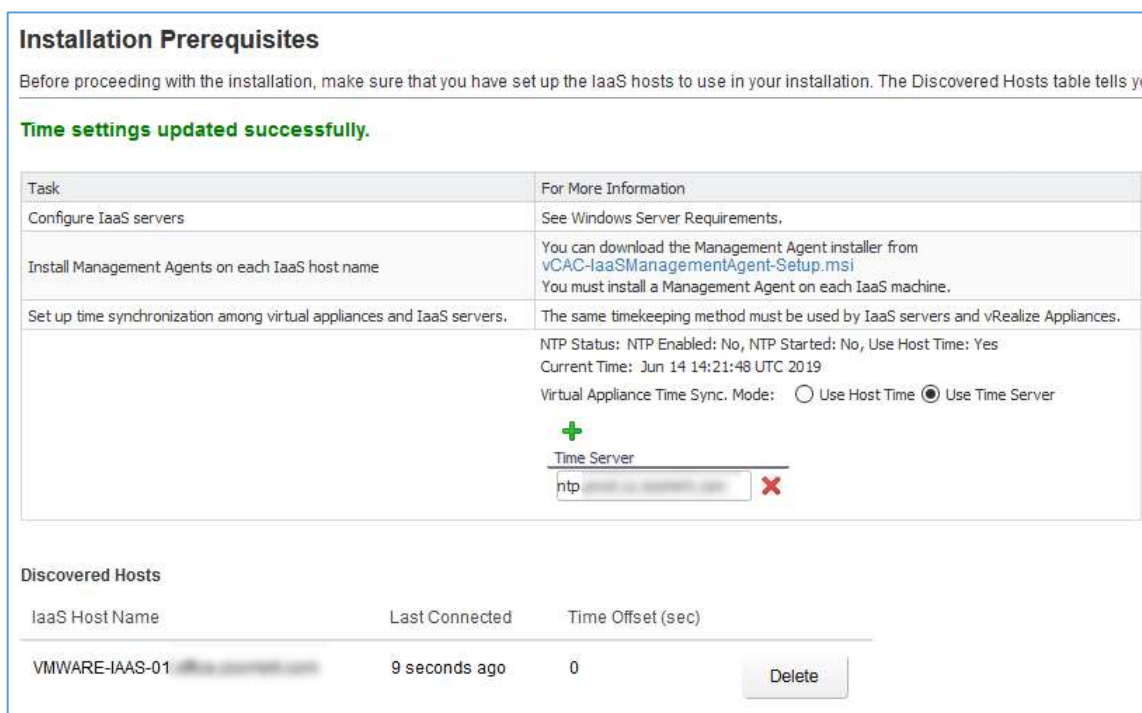
This will be used as an initial status of the SSH service in the appliance. You can change it later from the appliance Web console.

Obrázek 28 - přizpůsobení operačního systému (zdroj: vlastní)

Po úspěšném importu je možné virtuální stroj zapnout a nechat provést úvodní inicializaci a konfiguraci na základě poskytnutých údajů.

Příprava IAAS serveru

Příprava IAAS serveru je vcelku jednoduchá, protože z větší části se o ní postará samotný konfigurační průvodce, kterého je nutné spustit, i když ještě nekonfigurujeme samotné vRA. VMware doporučuje pro IAAS server použít čistý operační systém bez předešlé konfigurace, protože změny prováděné při instalaci rolí, by mohly mít vliv na jiné běžící aplikace. Po nalogování do vRA rozhraní se tedy automaticky spustil konfigurační průvodce, v němž je nutné zvolit typ nasazení – minimální nebo enterprise (viz Obrázek 24) a až poté nabídne ke stažení instalační soubor s komponentou nazvanou *Management Agent*. V rámci stejného kroku je zároveň nastaven server NTP¹⁵ pro synchronizaci času. IAAS Servery se zde objeví okamžitě po instalaci agenta. Zároveň je možné agenty odstranit, pokud by nastal nějaký problém.



Installation Prerequisites

Before proceeding with the installation, make sure that you have set up the IaaS hosts to use in your installation. The Discovered Hosts table tells you

Time settings updated successfully.

Task	For More Information
Configure IaaS servers	See Windows Server Requirements.
Install Management Agents on each IaaS host name	You can download the Management Agent installer from vCAC-iaasManagementAgent-Setup.msi You must install a Management Agent on each IaaS machine.
Set up time synchronization among virtual appliances and IaaS servers.	The same timekeeping method must be used by IaaS servers and vRealize Appliances. NTP Status: NTP Enabled: No, NTP Started: No, Use Host Time: Yes Current Time: Jun 14 14:21:48 UTC 2019 Virtual Appliance Time Sync. Mode: <input type="radio"/> Use Host Time <input checked="" type="radio"/> Use Time Server + Time Server ntp [redacted] X

Discovered Hosts

IaaS Host Name	Last Connected	Time Offset (sec)	
VMWARE-IAAS-01 [redacted]	9 seconds ago	0	Delete

Obrázek 29 - Stažení Management Agent-a a nastavení NTP serveru (zdroj: vlastní)

Průběh instalátoru je standardní, pouze vyžaduje zadat údaje, kam (ke které vRA) a s jakým uživatelem se připojit a pod jakým uživatelem má běžet agent samotný.

¹⁵ NTP – Network Time Protocol

Obrázek 30 - poskytnutí detailů připojení (zdroj: vlastní)

Obrázek 31 – Uživatel, pod kterým bude běžet Management Agent (zdroj: vlastní)

Všechny ostatní závislosti, nebo role operačního systému, doinstaluje konfigurační průvodce skrze tohoto agenta.

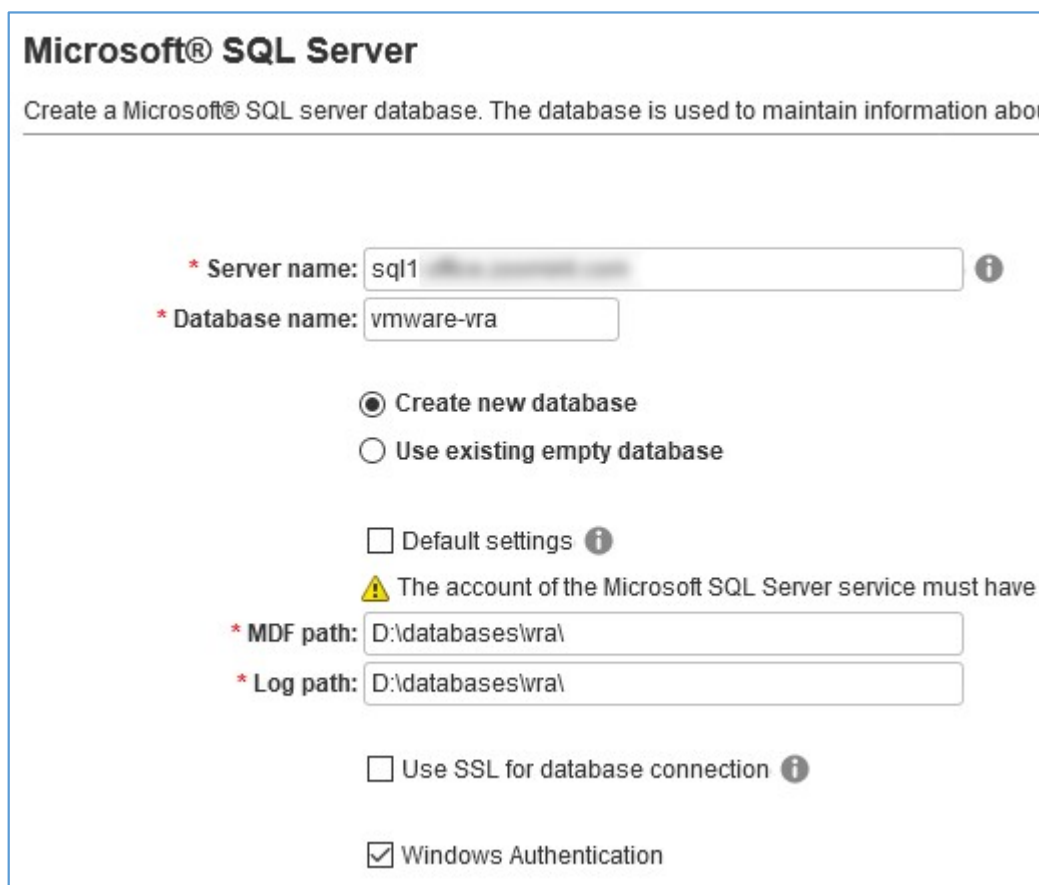
Konfigurace vRA

Konfigurace samotné vRA probíhá ve dvou úrovních. První je základní konfigurace pomocí už zmíněného konfiguračního průvodce, který se postará o správnou konfiguraci všech komponent a rolí tak, aby všechno spolu komunikovalo. Druhá úroveň konfigurace je v rámci webového portálu vRA, a věnuje se konfiguraci tenantu a nastavení ověřování uživatelů, šablon, katalogu služeb, workflow a tzv. „day 2 operations“ atd.

Konfigurace vRA – průvodce, instalace rolí

Průvodce je rozsáhlý a jeho správné nastavení vyžaduje dostatek času, existuje však několik postupů a návodů, které jsou veřejně dostupné na internetu a pomohou s prvotní orientací v produktu. Poté, co je připraven agent na IAAS serveru z předešlého kroku, je nutné pro tento server zvolit všechny požadované role (viz Obrázek 25). Průvodce zkontroluje všechny závislosti, požadavky a doinstaluje je. Následně je požadováno zadat FQDN jméno, které bude použito jako adresa pro load balancer, jehož úkolem je přerozdělovat zátěž mezi jednotlivými vRA. V našem případě se jedná pouze o DNS záznam směřující na jeden server. Volíme heslo pro první administrátorský účet hlavního tenantu, heslo pro šifrování databáze a adresu load balanceru pro IAAS servery. Po těchto základních

nastaveních je nutné nakonfigurovat a vytvořit SQL databázi, zvolit její jméno a umístění na disku SQL serveru.



Microsoft® SQL Server

Create a Microsoft® SQL server database. The database is used to maintain information about...

* Server name: sql1 i

* Database name: vmware-vra

Create new database

Use existing empty database

Default settings i

⚠ The account of the Microsoft SQL Server service must have...

* MDF path: D:\databases\wra\

* Log path: D:\databases\wra\

Use SSL for database connection i

Windows Authentication

Obrázek 32 - Konfigurace SQL Server databáze (zdroj: vlastní)

Další kroky zahrnují konfiguraci všech rolí IAAS serveru (DEM, Manager Service, agenti pro jednotlivé platformy jako vSphere atd.), uživatele, pod kterým budou běžet služby a eventuálně cesta pro instalaci.

Agents

You can optionally install agents for your deployment. Select a machine to host the agent and the agent type from the drop-down menus a

✓ All parameters are valid.

For high-availability mode, you must install two or more instances of the same agent on different servers. Give each instance of the agent

+

iaaS Host Name	Agent Name	Endpoint	Installation Path (Optional)	✘
VMWARE-IAAS-01	iaas-agent	vmware	(Optional)	
Agent Type	Username			
vSphere				
	Password			
			

Obrázek 33 - Konfigurace agenta pro vSphere (zdroj: vlastní)

Poslední částí před spuštěním procesu instalace na základě dat získaných z konfiguračního průvodce, je vytvoření TLS/SSL certifikátu pro webové portály v rámci vRA. vRA doporučuje po této části vytvořit snapshot na vRA i IAAS serverech. Jde o velmi užitečný krok, neboť se při jakékoli chybě dá vrátit a opravit ji. Občas se stane, že se vyskytne chyba i v samotné vRA, která vyžaduje workaround. Možnost vrátit se byla v rámci implementace využita několikrát, protože implementovaná verze vRA 7.6, obsahovala chyby spojené s Windows Server 2019, které vyžadovaly nestandardní kroky.

Installation Details

10% Installation is in progress.

Retry Failed button allows retrying of single or multiple failed commands. Depending on the reason for failure some commands cannot be retried. If Retry Failed is enabled yo

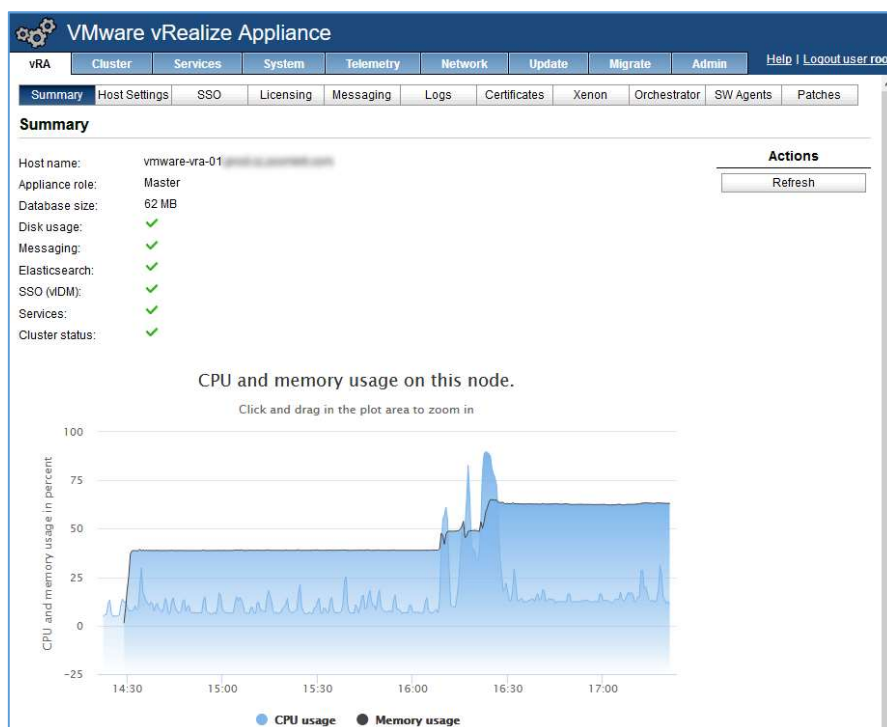
Retry All IaaS button retries all installation commands for IaaS components only. All IaaS servers must be reverted to the snapshots created on the snapshot page before the i

To change input or restart the whole installation after the installation has started revert all Virtual Appliances and IaaS machines and start installation again.

Host Name \ Instance	Role	Command Status
▶ Configure Single Sign-On	VA	✓ Succeeded
▶ Configure Postgres	VA	⚙ In progress ...
▶ Verify that all services are started	VA	⌚ Pending
▶ VMWARE-IAAS-01 .Web Certificate		⌚ Pending
▶ VMWARE-IAAS-01 (SQL Server:)	Database	⌚ Pending
▶ VMWARE-IAAS-01 .WEB	Web	⌚ Pending
▶ VMWARE-IAAS-01 Manager Service	Manager Service	⌚ Pending
▶ VMWARE-IAAS-01 .DEO	DEO/DEM	⌚ Pending
▶ VMWARE-IAAS-01 .dem-01	DEO/DEM	⌚ Pending
▶ VMWARE-IAAS-01 iaas-agent	Agent	⌚ Pending

Obrázek 34 - Instalace komponent a rolí (zdroj: vlastní)

Po úspěšné instalaci je správce přeměřován na hlavní stránku správy operačního systému a služeb vRA. Táto sekce, resp. část vRA je věnovaná primárně operacím souvisejícím se správou samotného operačního systému, monitoringu služeb a vytížení, správou clusteru a aktualizacím. Výhodou tohoto rozhraní je, že je téměř okamžitě vidět, jestli se ve vRA vyskytuje nějaký problém, ať už na nějaké službě, nebo clusteru, a proto je vhodné jej kontrolovat.



Obrázek 35 - Dashboard vRA pro akce související s platformou a operačním systémem (zdroj: vlastní)

Konfigurace vRA – Konfigurace tenantu, ověřování a zdrojů

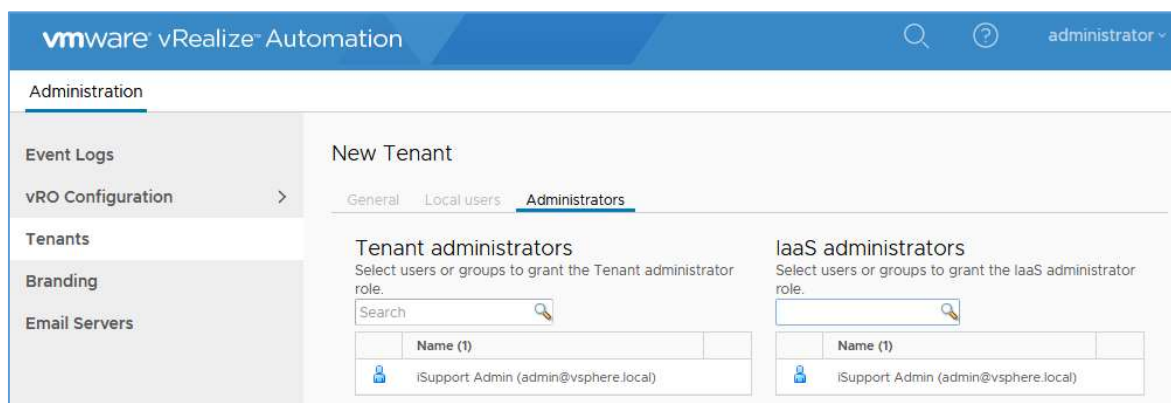
Jako základ celého vRA slouží výchozí tenant, pojmenován jako „vsphere.local“. V menších společnostech tento tenant bohatě postačuje a není nutné vytvářet další. Protože je vRA navrženo jako multi-tenantní prostředí, hodí se to v situacích a nasazeních u poskytovatelů služeb, kde jeden tenant může představovat jednoho zákazníka.

Než se správce dostane k nastavení zdrojů nebo šablon, je nutné tento výchozí tenant nakonfigurovat. Po přihlášení výchozím uživatelem „administrator@vsphere.local“, je možné pouze vytvořit prvního administrátora pro výchozí tenant, nastavit

The screenshot shows the login page for the 'Workspace ONE' tenant. It features a 'Username' field with the value 'vsphere.local' and a 'Password' field. Below the fields is a blue 'Sign in' button. There are also links for 'Forgot Password?' and 'Change to a different domain'.

Obrázek 36 - Přihlášení k výchozímu tenantu (zdroj: vlastní)

emailový server, branding a vytvářet další tenanty. Tento výchozí účet není možné použít na nic jiného, protože mu nelze přidělit práva v rámci tenantu.



Obrázek 37 - Nastavení práv pro první administrátorský účet (zdroj: vlastní)

Ve chvíli, kdy je vše připraveno, a prvnímu uživateli byla přidělena práva, je možné se znova přihlásit a začít konfigurovat ověřování pro koncové uživatele a vytvářet infrastrukturu pro běh virtuálních strojů.

Pro konfiguraci ověření je zde připravena možnost přidat více adresářových služeb pomocí Active Directory, nebo standardního protokolu LDAP. Výhodou vRA v tomto ohledu je možnost úzce specifikovat na základě skupin, kteří uživatele se budou synchronizovat. To zajišťuje určitou míru bezpečnosti tak, aby se neoprávnění uživatelé nepřihlásili. Je běžné vidět, že synchronizace v jiných typech software vždy synchronizuje všechny účty a není tam možnost omezení. V tomto má vRA, co se týče bezpečnosti, navrch.

Další částí je vytvoření tzv. endpointů, které slouží pro komunikaci s virtualizačními platformami jako např. vSphere, AWS, Hyper-V, atd. Jde o definici adresy a účtu, který bude s danou platformou komunikovat a provádět úkoly. Tento účet byl v tomto případě vytvořen při přípravě IAAS serveru, kde se zároveň definovala potřebná oprávnění v prostředí vCenter Serveru. V tuhle chvíli je vše připraveno, aby bylo možné začít s konfigurací infrastruktury, rezervací a zdrojů.

Prvním krokem je definice tzv. „Fabric Groups“, které reprezentují prostředí virtualizace a jeho zdroje. V Prostředí VMware se jedná o selekci clusteru v rámci vCenter Serveru. Z daného clusteru se pak použijí veškeré servery. Pro tuto skupinu je nutné definovat administrátory, kteří ji budou moci použít dále v konfiguraci.

New Fabric Group
Create a fabric group by assigning fabric administrators and select the compute resources that they can manage.

* Name: Cluster3

Description:

Fabric administrators:

VMware vRA Admins

Compute resources:

	Name	Endpoint	Platform Type
<input checked="" type="checkbox"/>	Cluster3	vmware	vSphere (vCenter)

Obrázek 38 - Definice Fabric Group (zdroj: vlastní)

Než je ale možné tyto zdroje použít, je nutné provést několik dílčích nastavení. Prvním z nich jsou prefixy pro jednotlivé virtuální stroje. Většina administrátorů použije svoje vlastní schéma, bohužel musí být vytvořen minimálně jeden prefix, většinou označovaný jako „fallback“. Dále je nutné vytvořit „Business Group“ skupinu, která z dostupných adresářů ověření specifikuje role pro její členy, např. role uživatele, nebo manažera. S rolí jsou spojená i některá oprávnění. Posledním dílčím nastavením je vytvoření síťového profilu pro virtuální stroje, který definuje základní síťová nastavení, jako IP adresu, masku, gateway, dns, atd., tedy reprezentaci interního IPAM serveru. Implementace v této práci ale tento profil nepoužívá, protože interní IPAM ve vRA nemá návaznost IP adresa – Hostname. Proto byl vytvořeno vlastní IPAM řešení, které bude popsáno v další části této práce.

Všechna předešlá dílčí nastavení jsou použita v rámci tzv. „rezervací“. Ty reprezentují vztah, kdo může použít definované zdroje (CPU, RAM, diskový prostor, síť) do jaké výše a kolik virtuálních strojů může být v této rezervaci vytvořeno početně (z důvodu licencování, ne zdrojů).

Edit Reservation - vSphere (vCenter)
Change settings in a listed reservation.

General Resources Network Properties Alerts

* Name: res_cluster3

Tenant: vsphere.local

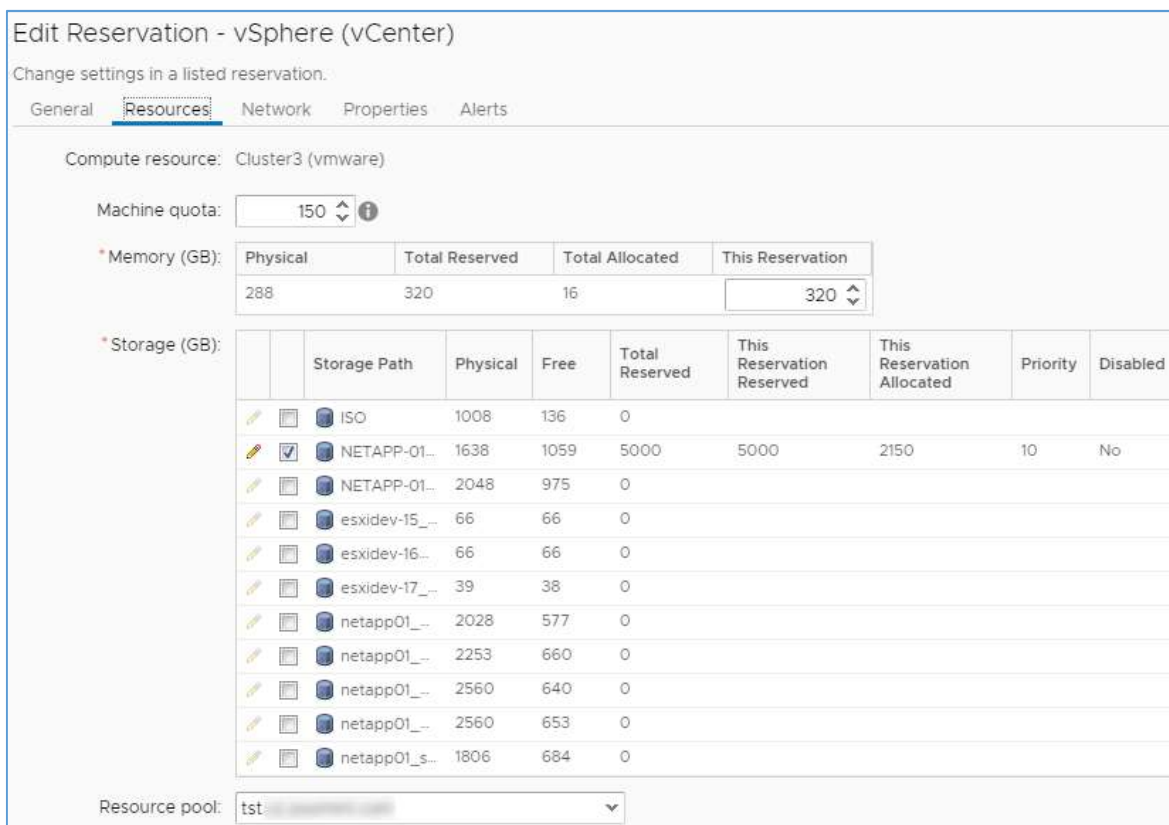
Business group: engineering

Reservation policy: respolicy_cluster3

* Priority: 10

Enable this reservation

Obrázek 39 - Vytvoření rezervace (zdroj: vlastní)



Obrázek 40 - Vytvoření rezervace (zdroj: vlastní)

vRA obsahuje velké množství dalších možností a způsobů nastavení pro různé platformy. Nastavení popsané výše je pro účely této práce více než dostačující. Zároveň však není možné popsat veškeré možnosti, protože to daleko přesahuje rozsah práce.

4.3.6 Nasazení automatizace

Při stavbě samotné automatizace, workflow a možnosti koncovým uživatelům vyžádat nové virtuální stroje, bylo potřeba začít přípravou šablon VM, které mohou být použity v rámci navrhovacího rozhraní plánů (blueprint designer). Těchto šablon bylo několik, ale pro účely práce bude popsán pouze jeden typ šablony, plán a workflow včetně skriptů, které se spouští při nasazení. V předešlé části bylo zmíněno, že vestavěný IPAM ve vRA není dokonalý, a protože společnost aktuálně žádný nepoužívá a ani neplánuje (pro velikost sítě to není potřeba), bylo potřeba situaci vyřešit jiným způsobem. Zároveň existoval požadavek, že nasazené virtuální stroje musí IP adresou odpovídat konkrétnímu hostname bez ohledu na to, jestli je použito DHCP, nebo statická adresa, a nesmí nastat konflikt.

Správa IP adres

Jako řešení bylo zvolená obyčejná tabulka v SQL databázi, která bude obsahovat všechny potřebné informace, které se využijí v průběhu nasazování VM. Struktura je jednoduchá a přehledná:

- [id] [uniqueidentifier] – ID řádku pro identifikaci
- [hostname] [nvarchar](30) – hostname VM v krátkém formátu
- [domainname] [nvarchar](100) – doménové jméno
- [ip] [nvarchar](15) – IP adresa
- [subnet] [nvarchar](15) - Maska sítě
- [gw] [nvarchar](15) - Gateway
- [dns1] [nvarchar](15) – DNS Server 1
- [dns2] [nvarchar](15) – DNS Server 2
- [mac] [nvarchar](17) - MAC adresa
- [lock] [bit] – Bit pro uzamykání záznamu ve chvíli použití
- [purpose] [nvarchar](50) – účel záznamu, možné rozdělovat záznamy podle použití v případě, že by integrace, pro které se VM nasazuje, musely mít konkrétní rozsah IP adres

	id	hostname	domainname	ip	subnet	gw	dns1	dns2	mac	lock	purpose
1	016EFF15-D7B9-E911-80EF-005056876651	auto011	tst.domain.com	10.71.0.11	255.255.254.0	10.71.0.1	10.70.10.10	10.50.10.10	00:50:56:AF:00:11	0	ANY
2	026EFF15-D7B9-E911-80EF-005056876651	auto012	tst.domain.com	10.71.0.12	255.255.254.0	10.71.0.1	10.70.10.10	10.50.10.10	00:50:56:AF:00:12	0	ANY
3	036EFF15-D7B9-E911-80EF-005056876651	auto013	tst.domain.com	10.71.0.13	255.255.254.0	10.71.0.1	10.70.10.10	10.50.10.10	00:50:56:AF:00:13	0	ANY
4	046EFF15-D7B9-E911-80EF-005056876651	auto014	tst.domain.com	10.71.0.14	255.255.254.0	10.71.0.1	10.70.10.10	10.50.10.10	00:50:56:AF:00:14	0	ANY
5	056EFF15-D7B9-E911-80EF-005056876651	auto015	tst.domain.com	10.71.0.15	255.255.254.0	10.71.0.1	10.70.10.10	10.50.10.10	00:50:56:AF:00:15	0	ANY
6	066EFF15-D7B9-E911-80EF-005056876651	auto016	tst.domain.com	10.71.0.16	255.255.254.0	10.71.0.1	10.70.10.10	10.50.10.10	00:50:56:AF:00:16	0	ANY
7	076EFF15-D7B9-E911-80EF-005056876651	auto017	tst.domain.com	10.71.0.17	255.255.254.0	10.71.0.1	10.70.10.10	10.50.10.10	00:50:56:AF:00:17	0	ANY
8	086EFF15-D7B9-E911-80EF-005056876651	auto018	tst.domain.com	10.71.0.18	255.255.254.0	10.71.0.1	10.70.10.10	10.50.10.10	00:50:56:AF:00:18	0	ANY
9	096EFF15-D7B9-E911-80EF-005056876651	auto019	tst.domain.com	10.71.0.19	255.255.254.0	10.71.0.1	10.70.10.10	10.50.10.10	00:50:56:AF:00:19	0	ANY
10	0A6EFF15-D7B9-E911-80EF-005056876651	auto020	tst.domain.com	10.71.0.20	255.255.254.0	10.71.0.1	10.70.10.10	10.50.10.10	00:50:56:AF:00:20	0	ANY

Obrázek 41 - Ukázka dat z tabulky správy IP adres (zdroj: vlastní)

Pro to, aby s co největší mírou byla v rámci procesu nasazení vybraná pro každou VM pouze jedna unikátní nepoužitá adresa, a aby se zabránilo (aspoň ve větší míře) náhodnému výběru stejné adresy pro dva nezávislé požadavky, byla napsána následující procedura. Procedura vybere náhodný nezamčený záznam, a hned jej uzamkne a vrátí jeho ID volajícím. Náhodnost je docílena částí „ORDER BY NEWID()“.

```

CREATE PROCEDURE [dbo].[getFreeMachine]
    @purpose nvarchar(50) = 'ANY'
AS
BEGIN
    SET NOCOUNT ON;

    DECLARE @MyTableVar table( id [uniqueidentifier] );

    UPDATE [machines]
        set lock = 1
        output inserted.id INTO @MyTableVar
        where id in
            ( SELECT TOP 1 id FROM [machines] WHERE lock = 0 AND PURPOSE = @purpose
            ORDER BY NEWID() )

    Select * from [machines] where id = (select id from @MyTableVar)
END

```

Pro uvolnění VM po její expiraci a vymazání se používá pouze update:

```
UPDATE [machines] SET lock = 0 WHERE id = @machineid
```

Tyto skripty jsou sice jednoduché, ale za to efektivní a rychlé. Pro dané vývojové prostředí není nutné vyvíjet žádné složité konstrukce, které by zpomalovaly celý proces nasazení VM.

Šablona virtuálního stroje

Dříve, než je navržen plán (blueprint), je nutné vytvořit šablonu VM s nainstalovaným operačním systémem. Nasazení operačního systému je možné dělat i jinými způsoby, např. přes PXE, ale v tomto případě byla potřeba mít operační systém dostupný co nejrychleji po odeslání požadavku. Šablona je standardním virtuálním strojem s nainstalovaným CentOS 7.7 a balíčky „*open-vm-tools*“ a „*perl*“, aby bylo možné při nasazení využít VMware Customization Specifications (možnost modifikace nastavení OS). Ve chvíli, kdy je šablona připravena k použití, bylo nutné vytvořit snapshot, v opačném případě by ji nebylo možné použít ve vRA.

Příprava plánu (blueprint)

Poté co vRA synchronizuje inventář vCenter Server (automaticky, nebo manuálně), je možné začít s přípravou plánu. Skrze menu „Design -> Blueprints -> New“ byl vytvořen nový blueprint, jemuž byly definovány základní parametry, jako jméno, limit nasazení v jednom požadavku, délka zápůjčky (lease) ve dnech (minimum a maximum), délka archivace po expiraci zápůjčky a set vlastností, které definují operační systém VM a její umístění ve struktuře.

Následující obrázky zobrazují úvodní nastavení.

Blueprint Properties

General NSX Settings Properties

* Name: CentOS 7 (SSD)

ID: CentOS7SmallVM

Description:

Deployment limit: 1

Lease (days): Minimum 1 Maximum 10

* Archive (days): 1

Propagate updates to existing deployments

Obrázek 42 - Základní parametry blueprintu (zdroj: vlastní)

Blueprint Properties

General NSX Settings Properties

Property Groups Custom Properties

+ Add Move Up Move Down View Properties Remove

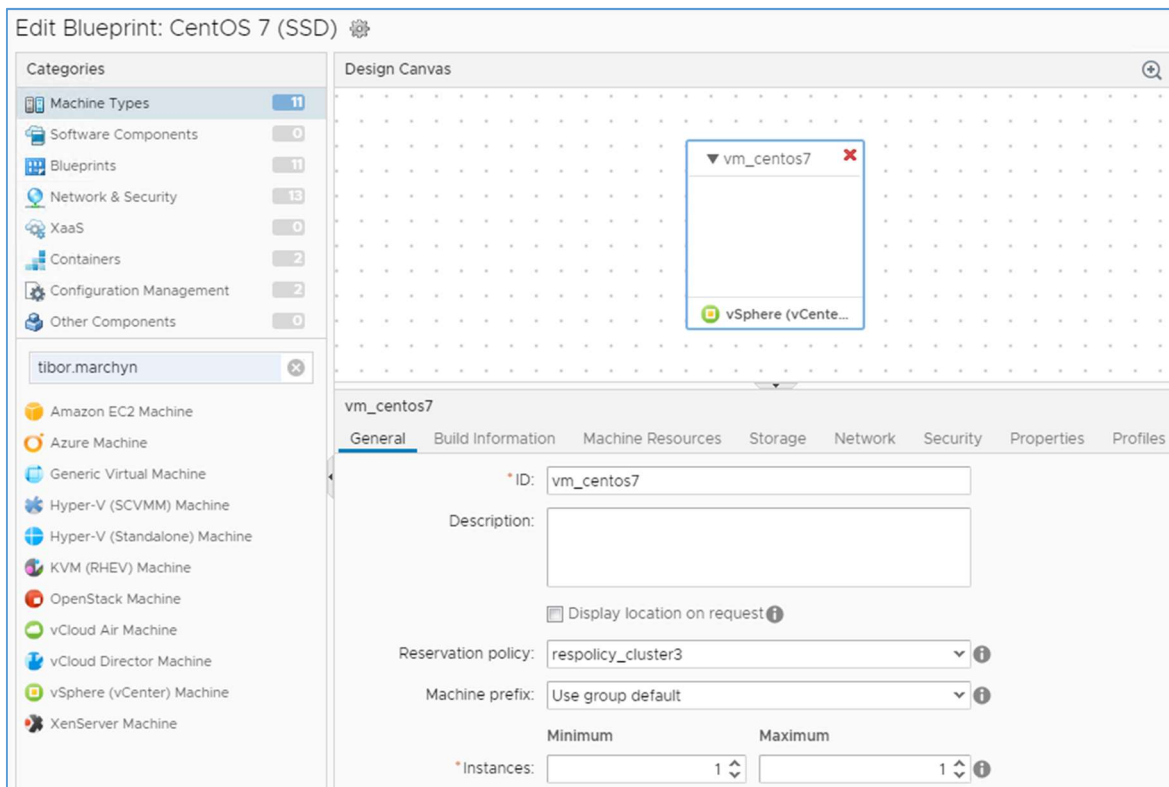
Name Description

Blueprint Required Properties Centos 7

Name	Value	Encrypted	Overrida...	Show in Request
VMware.VirtualCenter.OperatingSystem	centos7-64	No	Yes	No
VMware.VirtualCenter.Folder	Automation	No	Yes	No

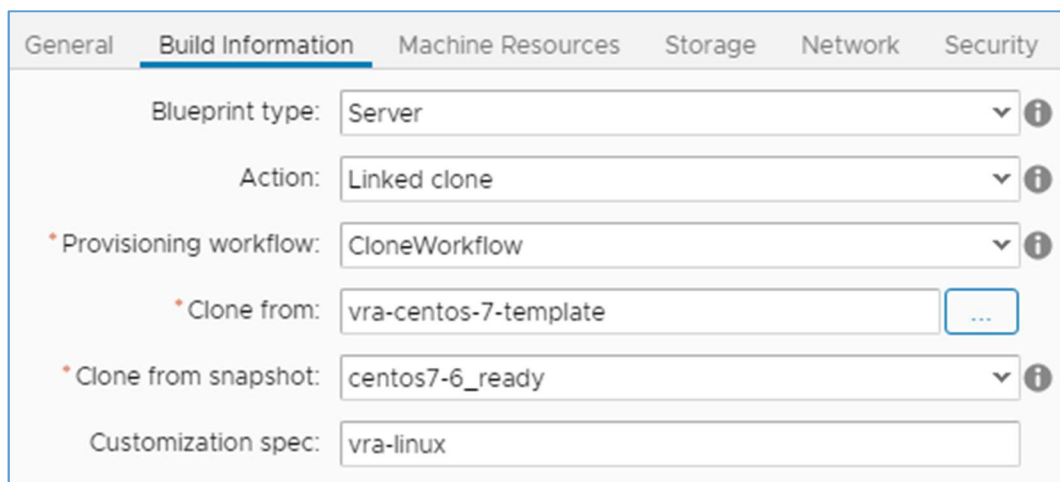
Obrázek 43 - Rozšiřující parametry blueprintu (zdroj: vlastní)

Po definici základních parametrů byl proveden návrh samotného plánu. Protože se jedná pouze o jeden virtuální stroj, návrh byl vcelku jednoduchý. Nejdůležitější částí byla definice parametrů reprezentace virtuálního stroje v návrháři tak, aby odpovídal finálnímu stavu. V následujících obrázcích jsou charakterizované jednotlivé části s definicí, proč jsou použité.



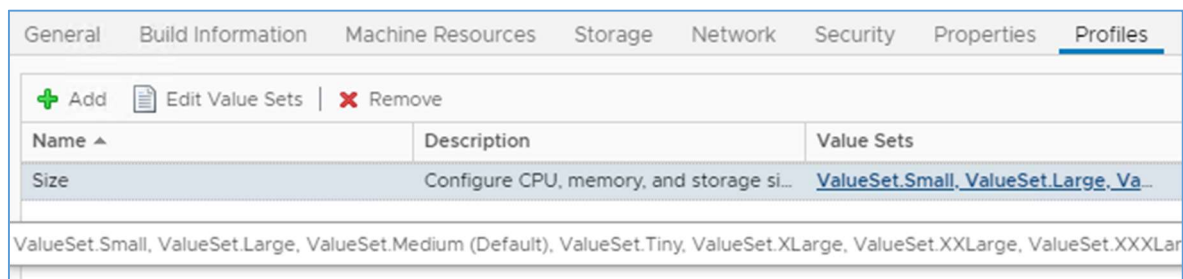
Obrázek 44 - Návrhář blueprintu (zdroj: vlastní)

Na záložce „Build Information“ bylo potřeba definovat zdrojovou šablonu VM a její snapshot. Informace o snapshotu je potřebná proto, že všechny klonované instance vytvořené tímto blueprintem jej budou využívat jako svůj zdrojový disk, a zápis se bude provádět pouze do rozdílových souborů. Tento způsob se nazývá „linked clone“. Výhodou tohoto typu je extrémně rychlé nasazení, kdy uživatelé nemusí čekat na zdlouhavé kopírování dat, bez ohledu na to, jestli jsou použity klasické nebo SSD disky.



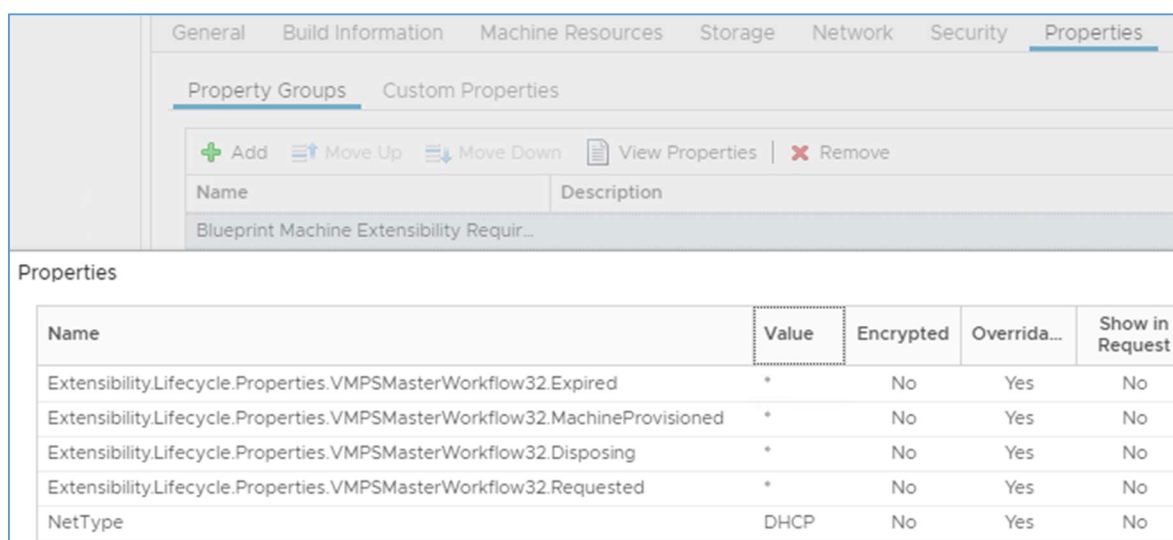
Obrázek 45 - Definice zdrojové šablony a způsob nasazení (zdroj: vlastní)

Dále jsou definované profily pro hardware zdroje. Profily umožňují uživateli vybrat si z předem nadefinovaných možností konfigurace zdrojů, obdobně jako ve veřejném cloudu, tedy „malá VM“, „střední VM“, „velká VM“ atd.



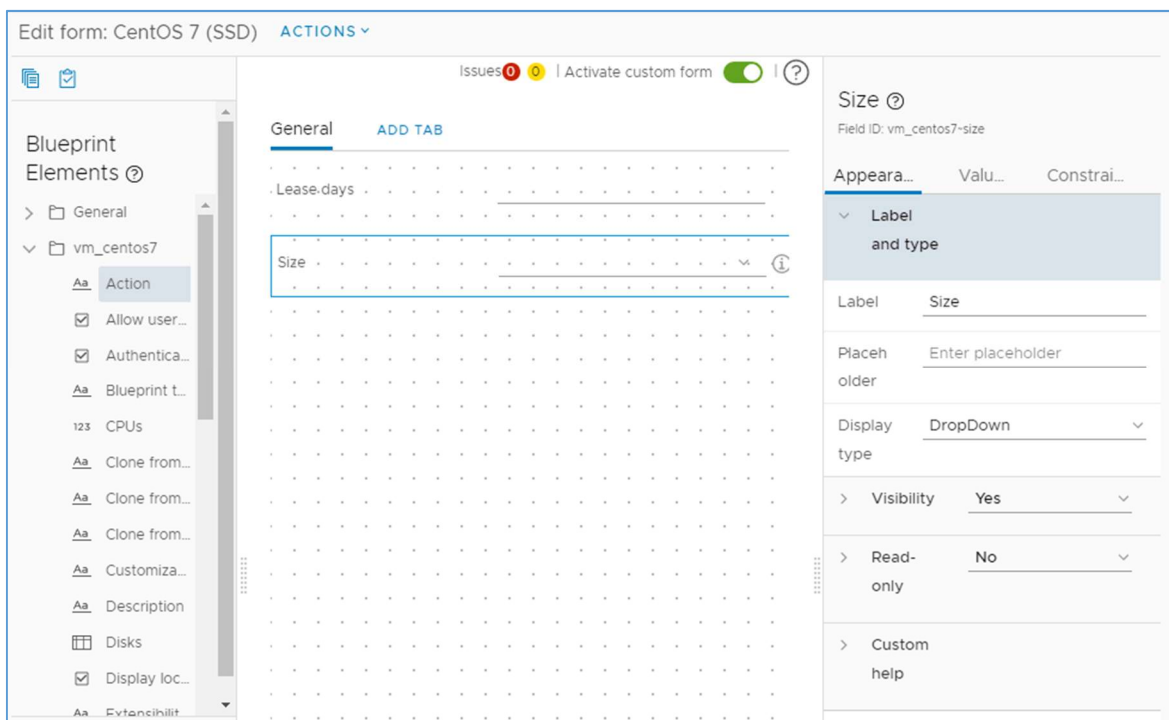
Obrázek 46 - Definice profilů v blueprintu (zdroj: vlastní)

Poslední věcí, která musí být definovaná, jsou vlastnosti instance. Bez vlastností, které jsou z třídy VMPSMasterWorkflow32, by se ve workflow nepředávaly informace o právě vytvářeném virtuálním stroji do workflow. Vlastnost „NetType“ je vlastní parametr říkající, jak se bude nastavovat IP adresa a obdobně je použit i ve workflow.



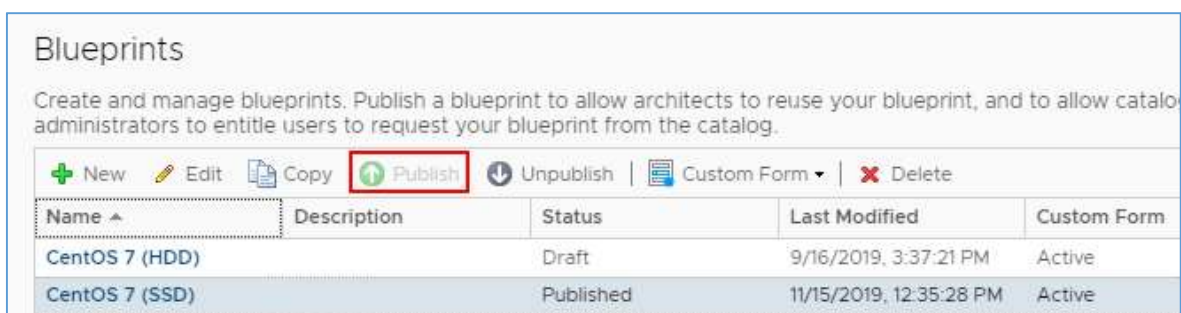
Obrázek 47 - Vlastnosti instance v návrhu (zdroj: vlastní)

Zbylé záložky nebyly v tomto návrhu použity, protože nejsou potřebné, nebo jsou modifikovány jiným způsobem. Po uložení návrhu se ještě editoval formulář žádosti, v němž se smazala veškerá pole a ponechána byla pouze volba profilu a délka zápůjčky (lease).



Obrázek 48 - Editor formuláře žádosti (zdroj: vlastní)

Po uložení formuláře je nutné daný blueprint publikovat, aby jej mohli uživatelé použít.



Obrázek 49 - Publikování blueprintu (zdroj: vlastní)

Pro to, aby jej uživatelé viděli, bylo ještě potřeba vytvořit tzv. katalog, který definoval, jací uživatelé mají přístup, ke kterým položkám a jakým akcím (myšleno vytvořit VM, exspirovat VM, prodloužit zápůjčku atd.). Tento katalog je definován v menu „Entitlements -> Catalog Management -> Entitlements“. Pro „Business Group“ a vybrané uživatele, se vyjmenují šablony nebo služby a příslušné akce, které lze používat. V katalogu lze modifikovat i kosmetické parametry nebo maximální počet nasazení konkrétní šablony pro „entitlement“.

Následující obrázky zobrazují nastavení Entitlementu pro katalog.

Edit Entitlement

General | Items & Approvals

* Name:

Description:

Expiration Date:

* Status: ⓘ

Last Updated By:

Last Updated On:

* Business Group:

* Users & Groups:

All Users and Groups

Search

Name
local admin (admin@vsphere.local)
VMware vRA Use (VMware vRA Use...)

Obrázek 50 - Definice Entitlement katalogu (zdroj: vlastní)

Entitled Items +

Search

Name	Approval Policy
CentOS 7 (SSD)	Large VMs Admin Approval [...]

Entitled Actions +

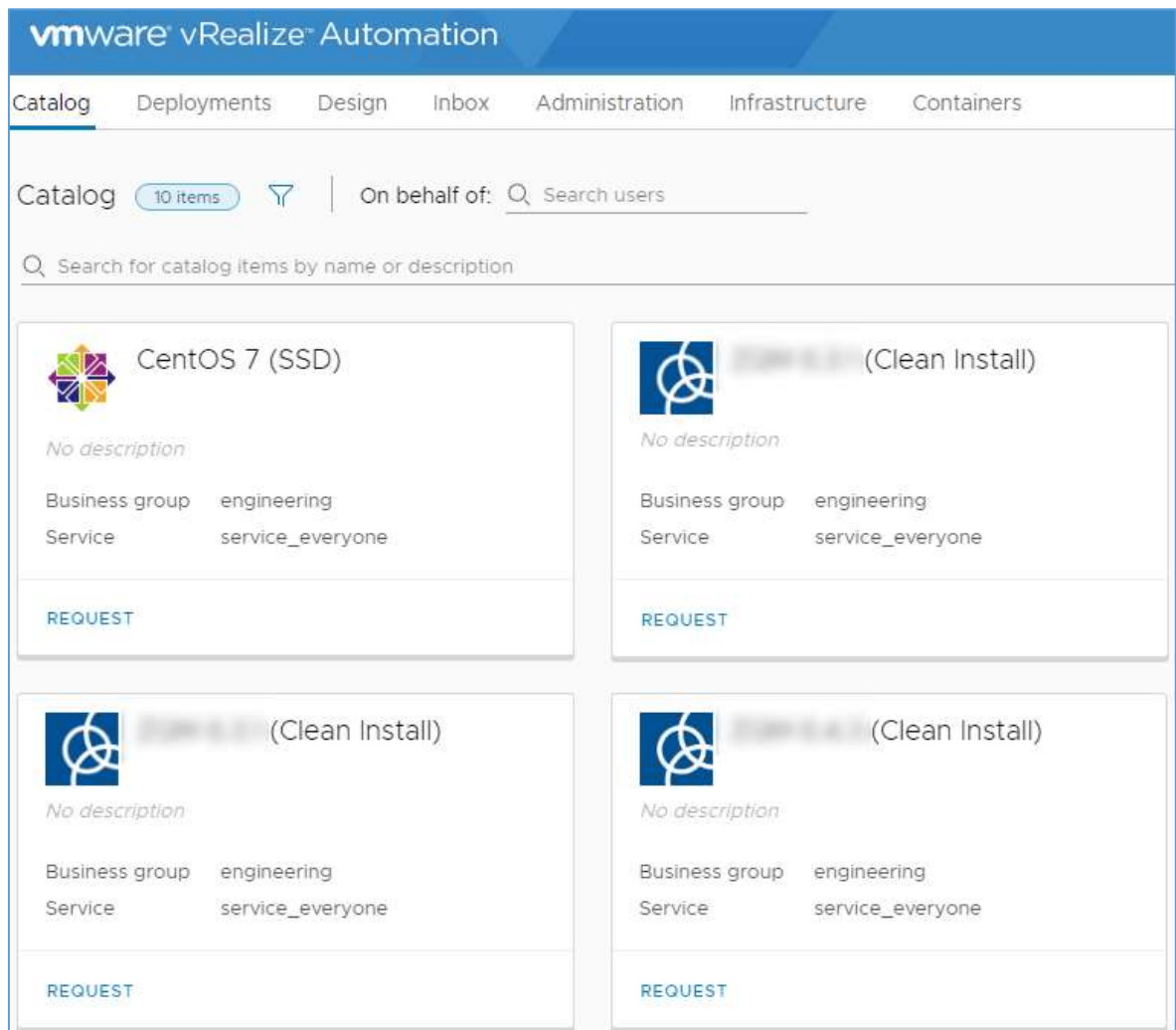
Actions only apply to items defined in this entitlement

Search

Name	Approval Policy
Change Lease (Deployment)	Change lease approval [Servi...]
Change Owner (Deployment)	(none)
Connect to Remote Console (...)	(none)
Connect using Console Ticke...	(none)
Connect using ICA (Machine)	(none)
Connect using RDP (Machine)	(none)
Connect using SSH (Machine)	(none)

Obrázek 51 - Definice Entitlement katalogu (zdroj: vlastní)

Výsledek nastavení je viditelný po navigaci do katalogu, podobně jako na následujícím obrázku.



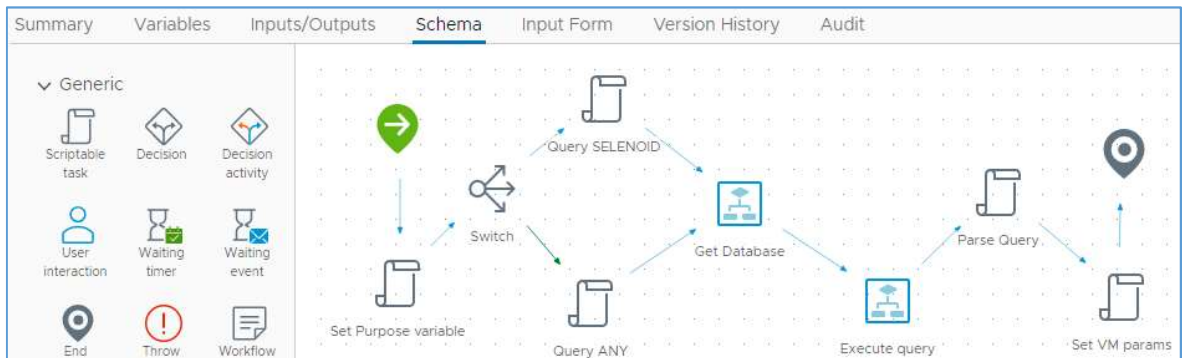
Obrázek 52 - Katalog pro uživatele (zdroj: vlastní)

Pokud by ale uživatelé začali vytvářet požadavky na jednotlivé virtuální stroje, nebyli by schopní na ně přistoupit, protože ještě nebylo nastaveno workflow. To je finálním krokem pro dokončení celého procesu.

Konfigurační workflow VM

Workflow, které bylo definováno, bylo vytvořeno pro, pokud možno, nejuniverzálnější přístup. Jeho účelem je na základě požadavku získat z databáze údaje o volné IP adrese a hostname jí náležící. Tyto údaje vrátí zpátky do procesu nasazení v konkrétně definovaném formátu dle dokumentace a proces zodpovědný za nasazení

modifikuje „customization specification“ tak, aby nastavila tyto údaje v operačním systému (viz část příprava šablony VM).



Obrázek 53 - Editor workflow (zdroj: vlastní)

Z obrázku je vidět rozdělení workflow podmínkou, která přizpůsobuje workflow pro výše zmíněné určení typu virtuálního stroje pro zařazení do konkrétního rozsahu IP adresy. Toto rozdělení pouze modifikuje SQL dotaz. Způsob získání záznamu byl definován výše v části „Správa IP adres“. V posledním kroku workflow se tyto parametry vrací zpátky, k čemuž je použit následující JavaScript kód.

```
var props = new Properties();

props.put("VirtualMachine.Admin.Name", vm_hostname);
props.put("VirtualMachine.Admin.Description", vm_hostname+"."+vm_domainname);
props.put("Hostname", vm_hostname);
props.put("DomainName", vm_domainname);
props.put("FQDN",vm_hostname+"."+vm_domainname);
props.put("ip_address",vm_ip);
props.put("vm_ip",vm_ip);
props.put("VirtualMachine.Network0.AddressType", "DHCP");
props.put("VirtualMachine.Network0.Name", "Vlan564_10-71-0-0__24");
props.put("VirtualMachine.Network0.MacAddressType", "static");
props.put("VirtualMachine.Network0.MacAddress", vm_mac);
props.put("machine_id", vm_id);
props.put("_deploymentName", vm_hostname);

virtualMachineAddOrUpdateProperties = props;
```

Pro uvolnění IP adresu bylo vytvořeno druhé workflow, které pouze aktualizuje databázi na základě ID, které se zapíše k nasazené VM jako „machine_id“

```
var machine_id = payload.get("machine").get("properties").get("machine_id");
query = "setMachineAsFree '" + machine_id + "'"
```

Obě workflow je nutné přiřadit ke konkrétní události, která vzniká v průběhu zpracování požadavku, a upřesnit podmínkami, které definují konkrétní stav této události. V opačném případě by se workflow nespustilo, protože samo o sobě neví kdy. V případě workflow pro přidělení IP adresy to je událost VMPSMasterWorkflow32.Requested, pro uvolnění IP adresy je to událost VMPSMasterWorkflow32.Disposing. Toto přiřazení se provádí v menu „Administration -> Events -> Subscriptions“.

The screenshot shows the 'Edit Workflow Subscription' window with the 'Conditions' tab selected. It includes a section for specifying conditions, a 'Selected workflow' section with input/output parameters, and a general configuration section.

Conditions:

- Run based on conditions (selected)
- All of the following
- Data > Lifecycle state > Lifecycle state name Equals VMPSMasterWorkflow32.Requested
- Data > Lifecycle state > State phase Equals PRE
- Data > Machine > Properties > NetType Equals DHCP

Selected workflow:

Name: DHCP IP Deployment

Description:

Input parameters:

Name	Type
payload	Properties

Output parameters:

Name	Type
virtualMachineAddOrUpdateProperties	Properties
_deploymentName	string

Configuration:

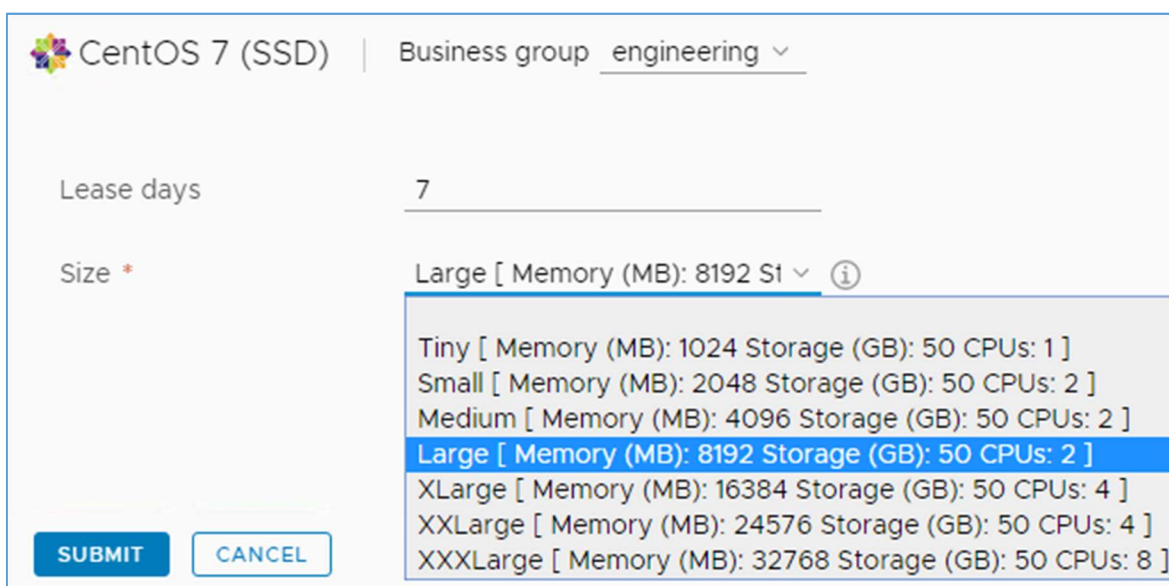
- Name: DHCP Deployment
- Priority: 10
- Timeout (min):
- Description:
- Blocking
- Stop processing if the workflow fails.

Obrázek 54 - Přiřazení workflow k události (zdroj: vlastní)

Na obrázku jsou vidět podmínky a události, workflow k nim přiřazené a obecné nastavení, jak se má workflow chovat. Blocking znamená, že celý proces nasazení VM čeká, než dané workflow proběhne. Zároveň je nastaveno, aby se zastavilo zpracování požadavku v případě pádu workflow.

Vytvoření požadavku na virtuální stroj

Všechny předešlé části této práce měly za cíl zpřístupnit automatizaci uživatelům. Po provedení veškerých nastavení se mohou uživatelé přihlásit do katalogu a sami vytvořit požadavek na předem definované šablony. Zároveň mohou využívat REST API, kterým své vlastní procesy ještě více automatizují. Dokumentace a charakteristika REST API (35) je rozsáhlá, daleko převyšuje rozsah této práce, proto je zde zobrazen pouze postup vytvoření požadavku skrze webové rozhraní. Následující obrázek zobrazuje možnost volby velikosti virtuálního stroje na základe požadovaných zdrojů a délku zápůjčky.



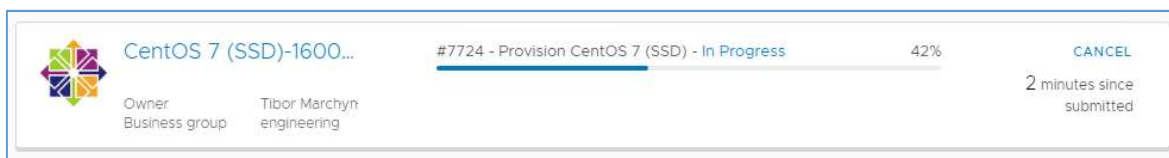
The screenshot shows a web interface for creating a VM. At the top, it says 'CentOS 7 (SSD)' and 'Business group engineering'. Below that, 'Lease days' is set to '7'. The 'Size' dropdown menu is open, showing several options with their respective specifications:

- Tiny [Memory (MB): 1024 Storage (GB): 50 CPUs: 1]
- Small [Memory (MB): 2048 Storage (GB): 50 CPUs: 2]
- Medium [Memory (MB): 4096 Storage (GB): 50 CPUs: 2]
- Large [Memory (MB): 8192 Storage (GB): 50 CPUs: 2]** (highlighted)
- XLarge [Memory (MB): 16384 Storage (GB): 50 CPUs: 4]
- XXLarge [Memory (MB): 24576 Storage (GB): 50 CPUs: 4]
- XXXLarge [Memory (MB): 32768 Storage (GB): 50 CPUs: 8]

At the bottom left, there are 'SUBMIT' and 'CANCEL' buttons.

Obrázek 55 - Vytvoření požadavku na VM (zdroj: vlastní)

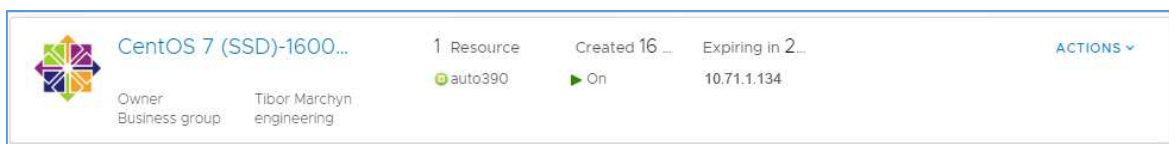
Po vytvoření požadavku uživatel vidí průběh zpracování požadavku a může jej sledovat.



The screenshot shows a progress bar for a VM request. The title is 'CentOS 7 (SSD)-1600...'. The progress is at 42%. The status is '#7724 - Provision CentOS 7 (SSD) - In Progress'. The owner is 'Tibor Marchyn' from the 'engineering' business group. It says '2 minutes since submitted' and has a 'CANCEL' button.

Obrázek 56 - Průběh zpracování požadavku na VM (zdroj: vlastní)

Po úspěšném dokončení požadavku může uživatel vidět název virtuálního stroje, jeho IP adresu a čas, za jaký expiruje.



The screenshot shows the details of a VM resource. The title is 'CentOS 7 (SSD)-1600...'. It shows '1 Resource' with a status of 'On'. The IP address is '10.71.1.134'. It says 'Created 16 ...' and 'Expiring in 2...'. The owner is 'Tibor Marchyn' from the 'engineering' business group. There is an 'ACTIONS' dropdown menu.

5 Výsledky a diskuse

5.1 Úprava procesů

Implementace přinesla nové možnosti v technologickém, ale i procesním rozvoji společnosti. Snížení požadavků na IT oddělení bylo možné pozorovat se stupňující se tendencí téměř okamžitě po předání dokumentace, jak tuto automatizaci používat, uživatelům. Zpočátku byla část původních úkolů požadujících vytvoření nového virtuálního stroje nahrazena požadavky na vytvoření nových šablon a plánů. Na základě toho byly upraveny původní procesy charakterizované v kapitole 4.2.3 tak, aby odpovídaly novým možnostem s přidáním nového aktora s názvem vRA.

Proces vytvoření nové VM

Proces byl rozšířen o možnost automatizace a vyřešení nového virtuálního stroje skrz samoobslužný portál. Hned na úvod bylo přidáno rozhodnutí, jestli uživatel může požadavek vyřešit sám, nebo potřebuje asistenci IT oddělení. V případě, že vyžaduje asistenci, je proces shodný s původní verzí definovanou v kapitole 4.2.3. V případě, že je uživatel schopný vyřešit daný požadavek sám, tak je proces následující.

Kroky části s automatizací:

- Uživatel se přihlásí do samoobslužného portálu, nebo se přihlásí pomocí API
- Žádá o vytvoření nového VM / vytváří JSON definující požadavek o novou VM
- Portál zobrazí formulář uživateli / vRA vrátí šablonu požadavku v JSON formátu
- Uživatel vyplní formulář a potvrdí jej / Uživatel upraví JSON a odešle jej zpět na server
- vRA validuje požadavek, jestli požadované zdroje vyžadují schválení IT oddělení
- V případě, že je vyžadováno potvrzení, vRA odešle notifikaci IT oddělení
- Člen IT oddělení se přihlásí do samoobslužného portálu a schválí, nebo zamítne požadavek
- V Případě zamítnutí, vRA odešle zamítavou odpověď uživateli.
- V případě že zdroje nepotřebují schválení, nebo IT oddělení schválilo požadavek, vRA daný požadavek zpracuje a vytvoří VM
- Po vytvoření VM je uživateli odeslána notifikace o zpracování a informacích o vygenerovaném virtuálním stroji

- vRA spouští podproces pro Správu životního cyklu VM, v němž se monitoruje její délka životnosti

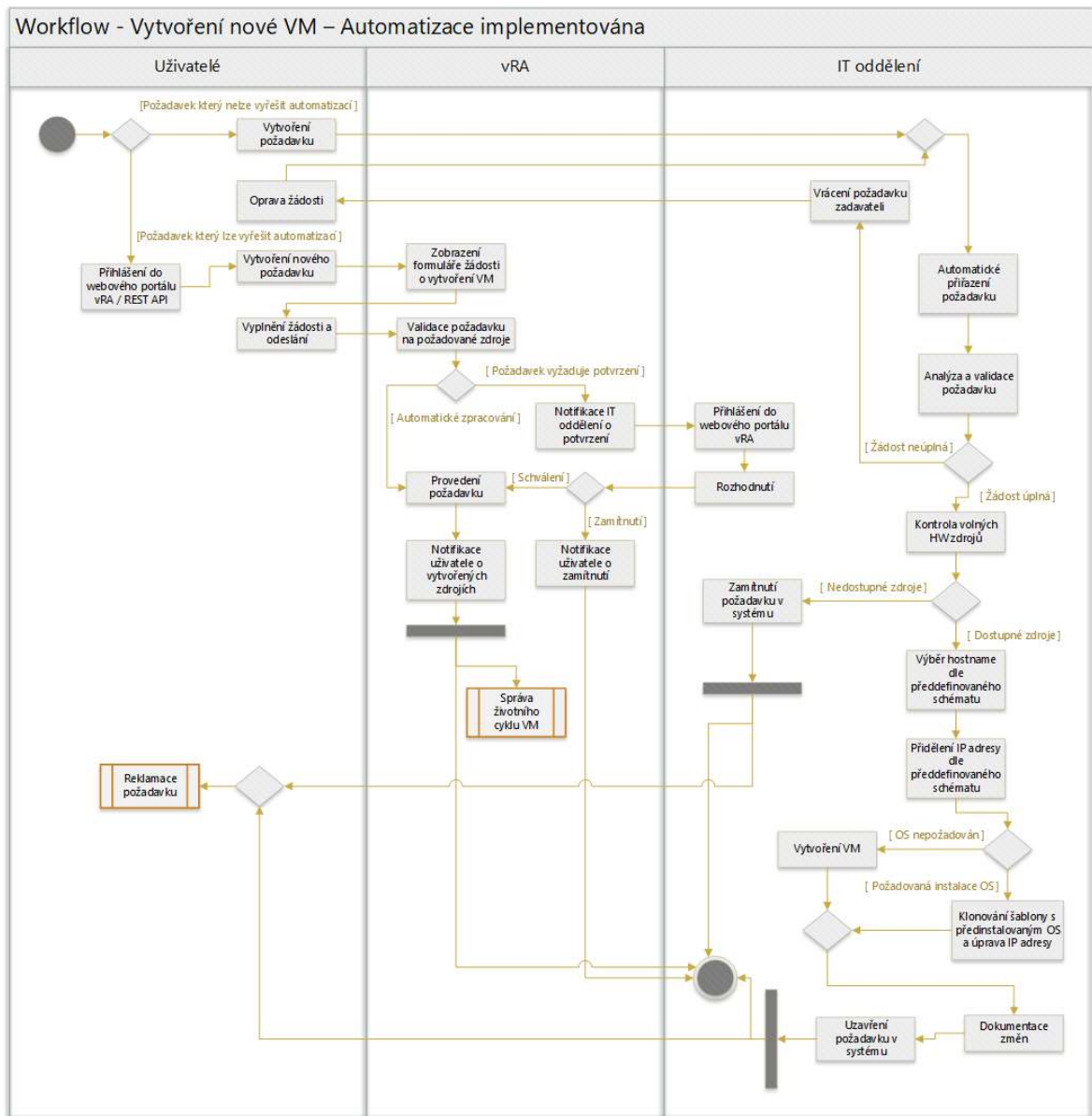


Diagram 4 - Proces vytvoření nové VM s implementovanou automatizací (zdroj: vlastní)

Proces změny VM

Tento proces byl obohacen o shodnou část kroků, jako vytvoření nové VM. Taktéž platí, že zbylá část procesu vyžadující asistenci IT, se shoduje s verzí definovanou v kapitole 4.2.3. Proces tedy už není potřeba znova definovat, protože obsahuje shodné části s už definovanými procesy. Diagram níže zachycuje proces změny VM.

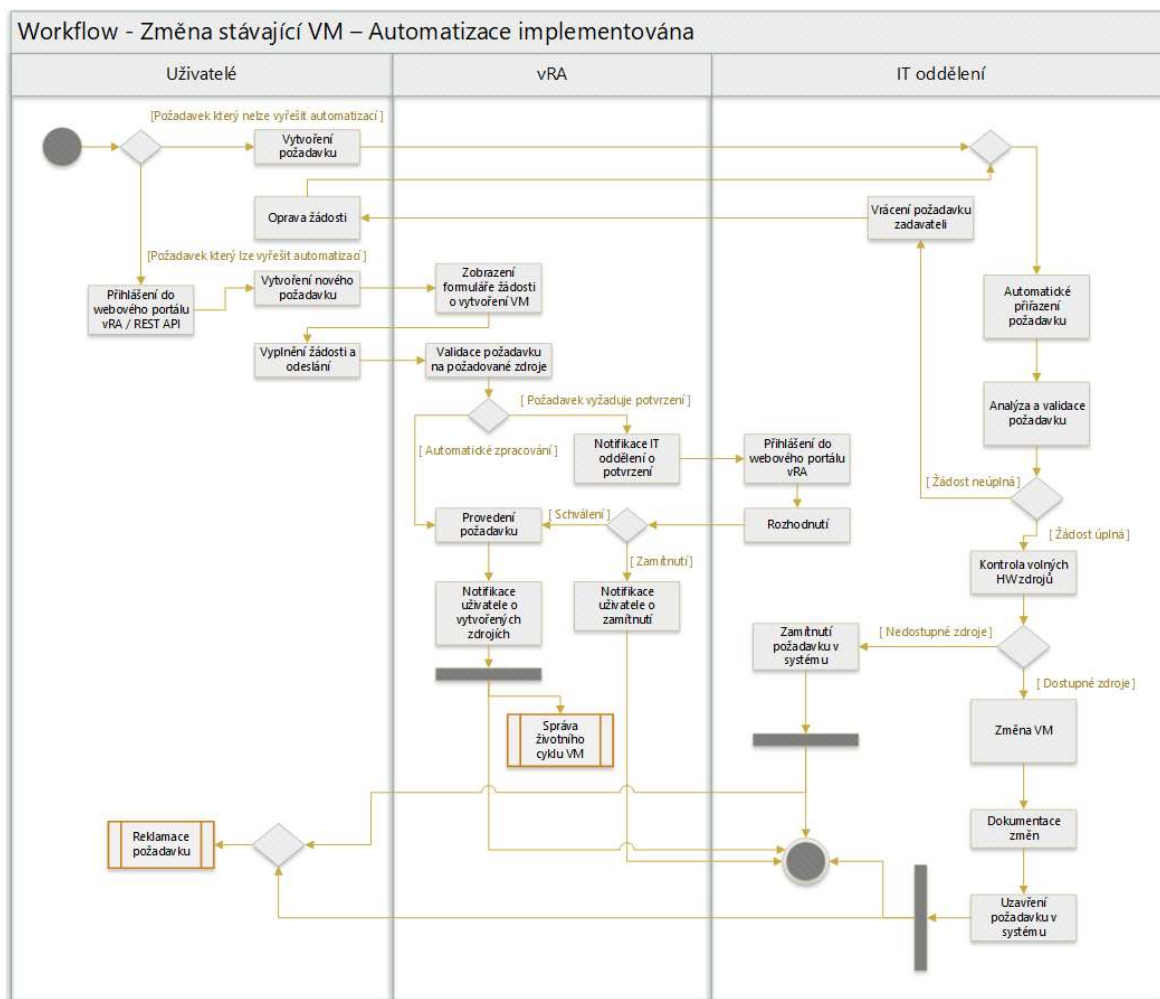


Diagram 5 - Proces změny VM s implementovanou automatizací (zdroj: vlastní)

5.2 Komparace původního a nového stavu

Všechna optimalizační opatření, která byla uvedena v předešlých kapitolách, byla provedena a produkt VMware vRealize Automation byl pořízen a implementován tak, jak bylo zachyceno v praktické části. Tento je využíván nejenom pro manuální požadavky, ale také pro automatické integrace s různými systémy. Tato implementace se odzrcadlila ve snížení zátěže IT oddělení, které místo práce na požadavcích týkajících se VMware infrastruktury, může tento čas věnovat dalšímu rozvoji infrastruktury a zvyšování efektivity procesů. Pro ostatní zaměstnance, kteří jsou uživateli této automatizační platformy, to přináší úsporu v čase stráveném čekáním na požadavky, které byly závislé na IT oddělení, a tento čas je přenesen do vývoje produktu, z kterého vznikají společnosti příjmy.

Pro komparaci stavu před implementací a po implementaci automatizace byl využit podobný princip, jako v kapitole 4.2.4, kde byly upraveny vstupní data o vytížení jednotlivých týmů.

Ze systému správy požadavků bylo zjištěno, že požadavky na VMware infrastrukturu klesly a IT oddělení věnuje práci na této platformě pouze 5 % pracovních hodin dvou lidí měsíčně (včetně tzv. skrytého času, který je spojen s jinými typy úkolů, které vzejdou z původních požadavků), tedy 8 hodin/člověk/měsíc * 2 = 16 hodin/měsíc (uvažujeme typický měsíc = 20 pracovních dní = 160 pracovních hodin).

Co se týče R&D týmů, tak se změnila délka prostojů, které z 80 % času klesly na pouhých 10 % času (čas, který IT oddělení potřebuje na zpracování požadavku), tedy na 1,6 hodiny. Tento výrazný pokles je způsoben tím, že pro většinu činností R&D týmům postačuje pouze dočasná VM, na které otestují, nebo vyvinou požadovanou funkcionalitu, a nechají ji automaticky smazat samotným vRA po skončení jejího životního cyklu, ve většině případu po sedmi dnech.

Na základě těchto informací můžeme upravit vzorec výpočtu provozních nákladů následovně:

$$\frac{\sum \text{mzda členů IT}}{\text{počet členů} * 160} * p_{hod} + 2 * \frac{\sum \text{mzda členů jednoho týmu R\&D}}{0,5 * \text{počet členů} * 160} * (0,1 * p_{hod})$$

kde:

- p_{hod} – počet hodin strávených prací na požadavcích

Následující tabulka uvádí výsledky mzdových provozních nákladů za měsíc a tři roky.

Typ	Náklad IT (CZK / měsíc)	Náklad R&D (CZK / měsíc)	Náklad za 3 roky (CZK)
Best case (nejlepší scénář)	7 000	0	252 000
Průměrný scénář	7 000	2 125	328 500
Worst case (nejhorší scénář)	7 000	4 250	405 000

Tabulka 9 - Přehled nákladu měsíčně a sumárně za 3 roky po implementaci automatizace (zdroj: vlastní)

Dle nové kalkulace je vidět značný rozdíl pro nejhorší scénář, kde se náklady za tři roky pohybují na úrovni 13,7 % původních nákladů. I nadále platí, že výpočet představuje pouze minimální provozní náklady a neobsahuje skryté náklady.

Do provozních nákladů zároveň vstupuje aspekt toho, že ušetřený čas se investuje zpátky do produktu společnosti a dalších činností s ním souvisejícími, za účelem zvýšení kvality a zhodnocení produktu.

5.3 Kalkulace TCO

Výpočet celkových nákladů spojených s vlastnictvím (TCO) je důležitým pro vytvoření představy o celkových nákladech, které budou firmu zatěžovat po dobu tří let. Zároveň to tvoří základ pro výpočet návratnosti investic (ROI). Výpočet ROI ale není součástí této práce. (36) (37) (38)

Vyjádřením TCO získáme celkové náklady na investici a její provoz (služba / výrobek) po celou dobu životnosti a užívání. Abychom mohli spočítat náklady na projekt, musíme získat všechny nákladové vstupy, ne jenom okamžitě ovlivňující projekt, ale i ty, které vstupují do nákladů později. Jedná se například o:

- Přímé náklady na pořízení investice (cena)
- Poplatky spojené s provozem/údržbou
- Náklady spojené s vyškolením uživatelů investice
- Náklady spojené s údržbou a opravami
- Náklady spojené s nečinností investičního celku z příčiny opravy, odstávky v důsledku poruchy, inovace atp.

Nevýhodou metody je, že nebere v úvahu hodnotu peněz a hodnotí investici pouze z pohledu nákladů, nikoliv výnosů. (42)

Pro výpočet TCO byly použity hodnoty z nejhoršího scénáře po implementaci automatizace, navýšené o 30 % pro pokrytí části, nebo všech skrytých nákladů. Provozními náklady pro hardware jsou náklady na elektrickou energii a chlazení.

Položka	Pořízení	Údržba a Provoz za 3 roky	Školení
Dell PowerEdge M420	300 000	116 640	0
VMware vRealize Automation Enterprise	3 500 000	526 500	20 000
VMware vSphere Enterprise Plus	600 000	0	0
	SUMA	5 063 140	

5.4 Doporučení

Po implementaci je viditelný klesající trend počtu požadavků směrem k IT oddělení, zároveň také subjektivní spokojenost a tzv. „user experience“ uživatelů automatizace, protože získali možnosti, které neměli, což zrychluje jejich procesy. To však neznamená, že by se v daném řešení nenašla místa, která by šla vylepšit, nebo zjednodušit.

Prvním doporučením je nadále pokračovat v rozvíjení automatizace jako takové a začít využívat vRA pro širší spektrum činností. vRA obsahuje další funkcionalitu, která doposud nebyla implementována, jako například nativní podpora moderních kontejnerových služeb (např. Docker), nasazování operačních systémů jiným způsobem než klonováním (např. PXE včetně Kickstart pro Linux), nebo implementace správy zdrojů z veřejných cloudů, jako např. Amazon AWS nebo Microsoft Azure.

Druhým doporučením je přehodnocení delegace rolí s vyššími oprávněními na uživatele mimo IT oddělení. Tím by se dala zajistit ještě vyšší kontinuita v procesech například tím, že plány (blueprint) by mohli vytvářet členové oddělení DevOps. Současně by odlehčili IT oddělení a získali by nástroj, jak ještě více přizpůsobit prostředí automatizace svým potřebám.

Třetím doporučením by mělo být přehodnocení, jestli uživatele správně využívají tzv. osobní virtuální stroje. V IT oddělení převládá názor, že tyto virtuální stroje, jsou zbytečné a měly by být nahrazeny automatizací a pouze dočasnými VM. Zefektivnilo by se využití hardwarových zdrojů a snížila by se potřeba nakupovat nový hardware.

Výše zmíněná doporučení by mohla vést k vyšší spokojenosti, je ale nutné dávat pozor na hranici, kdy se začnou zvyšovat náklady oddělení R&D, které jsou mnohonásobně vyšší než náklady IT oddělení, protože tyto náklady by se zpátky promítly do kvality a ceny produktů společnosti.

6 Závěr

V dnešní době je virtualizace brána jako standardní součást a je tzv. základním stavebním kamenem každé IT infrastruktury. Pro podniky už virtualizace nepředstavuje žádnou výzvu. Šetří čas správců sítě, náklady a umožňuje optimalizovat využití kapacit zdrojů tak, aby byl hardware co nejlépe využit. S nástupem cloudových služeb, softwaru jako služby (SaaS), infrastruktury jako služby (IaaS) a s požadavky vytvářet tzv. on-premis privátní cloud infrastrukturu se výzvou dnešní doby stala optimalizace procesů a jejich automatizace a zlepšování user-experience uživatelů.

V úvodu teoretické části se práce zaměřuje na virtualizační technologii, její historii, současnost a základní koncept produktu VMware vRealize Automation. Jsou zde zmíněné typy virtualizace, hypervizorů, základní terminologie a shrnutí výhod, ale i nevýhod virtualizace, včetně příkladů použití. Další část práce se věnuje automatizaci ve virtualizaci a některým nejčastěji používaných produktů sloužících k dosažení automatizace, jako např. Terraform, OpenStack a VMware vRealize Automation.

Stěžejní součástí teoretické části práce je detailní přehled produktu VMware vRealize Automation a jeho architektury. vRA poskytuje mnoho nástrojů a možností integrace a automatické skrze různé platformy, ne jenom VMware. Využití vRA nalezne převážně v IT společnostech, které hledají řešení zjednodušování procesů a integrací v oblasti vývoje a testování software. Rozhraní API v tomto směru poskytuje požadovanou volnost a zároveň respektuje politiky a zásady definované IT oddělením, jenž je odpovědné za zdroje. Zároveň jsou zde charakterizované možnosti nasazení, workflow a modifikace procesu nasazení skrz něj. Finální pasáž se věnuje jednotlivým fázím při nasazení vRA od posouzení, přes analýzu, plánování, až po návrh nové architektury, která bude obsahovat prvky automatizace.

Úvodní pasáž praktické části práce se věnuje krátké charakteristice vybrané společnosti, ve které byla provedena analýza stavu procesů a stávající infrastruktury, která je výhradně ve správě IT oddělení a představuje pouze malou část celkové infrastruktury společnosti.

V analýze stávajícího řešení byl kladen důraz na využití prostředí virtualizace, její strukturu a stav hardwarových zdrojů, které jsou v dané infrastruktuře dostupné. Specifikovaná část infrastruktury byla charakterizovaná i na hardware úrovni, na použitých technologiích a protokolech.

Dříve, než bylo možné přistoupit k implementaci, bylo nutné rozebrat a definovat procesy související s interakcí mezi týmy R&D a IT oddělením. Pro tento účel bylo využito modelovacího jazyka UML a diagramu aktivit, jehož grafická reprezentace přidává větší přehlednost k textové verzi procesu. Po definici procesů bylo přistoupeno ke kalkulaci minimálních provozních nákladů spojených s provozem a správou virtualizace. Kalkulace byla provedena na třech scénářích – „best case“, „worst case“ a průměrný.

Na základě rozboru zpětné vazby uživatelů virtualizace definovala společnost požadavky na optimalizaci. Z těchto požadavků byla sestavena kritéria pro multikriteriální hodnocení variant zmíněných v teoretické části. Po zhodnocení variant se začalo s návrhem změn do původní infrastruktury jak pro hardwarové, tak pro softwarové stránce. Byla navržena architektura vRA, u které bylo bráno v úvahu další rozšíření a řešení vysoké dostupnosti. Následně byla provedena samotná instalace a implementace řešení vRA s vysvětlením řešení konkrétního případu, který řeší životní cyklus od vytvoření požadavku uživatelem, až po vytvoření virtuálního stroje, včetně charakteristiky workflow, událostí na pozadí a integrace mezi vRA, DHCP serverem a SQL serverem.

Poslední část práce je věnovaná úpravě procesu se zakomponováním automatizace, opět pomocí modelu v jazyce UML. Dle nových vstupních dat získaných ze systému správy požadavků byly provedeny nové propočty minimálních provozních nákladů spojených s časem zaměstnanců IT oddělení souvisejícím s prací na virtualizační infrastruktuře a prostojů, které tím vzniknou ostatním oddělením. Na základě těchto údajů byl proveden výpočet celkových nákladů spojených s vlastnictvím (TCO),

Práce je zakončena stanovením doporučení a kroků, které by měly pomoci k dalšímu rozvoji a zkvalitnění „user experience“ koncových uživatelů. Doporučení jsou například další rozvoj automatizace a možností nasazení operačního systému a virtuálních strojů, potenciální delegace rolí na R&D, odlehčení IT oddělení od administračních úkolů, nebo přehodnocení využití virtuálních strojů, které jsou osobní a přiřazené většině uživatelů.

Řešení automatizace, realizace analýz a optimalizace procesů a implementace je specificky zaměřená na vybranou společnost, nicméně zkušenosti s implementací, postupy a teoretické základy jsou použitelné i pro jiné společnosti, které se k podobnému kroku chystají.

7 Seznam použitých zdrojů

1. MARCHYN, Tibor. *Analýza a porovnání virtualizačních nástrojů*. Praha: Česká Zemědělská Univerzita v Praze. Technická fakulta. Bakalářská práce, 2014. Vedoucí práce Ing. Zdeněk Votruba.
2. BITNER, Bill a GREENLEE, Susan. *z/VM – A Brief Review of Its 40 Year History*. [Online] 2012. [cit. 2020-01-02]. Dostupné z: <http://www.vm.ibm.com/vm40hist.pdf>.
3. *IBM and HP virtualization*. [Online] 2009. [cit. 2020-01-02]. Dostupné z: <https://www.ibm.com/developerworks/aix/library/au-aixhpvirtualization/>.
4. *Hyperkonvergovaná infrastruktura*. [Online] GAPP System, spol. s r. o. [cit. 2020-01-02]. Dostupné z: <https://www.gapp.cz/hyperkonvergovana-infrastruktura>.
5. *VMware and DynamicOps*. [Online] 2012. [cit. 2020-01-02]. Dostupné z: <https://www.vmware.com/company/acquisitions/dynamicops.html>.
6. *Virtualization*. [Online] Poslední aktualizace 2012. [cit. 2020-01-02]. Dostupné z: <https://www.vmware.com/cz/solutions/virtualization.html#overview>.
7. HAMBURGER, Valentin. *Building VMware Software-Defined Data Centers*. Birmingham: Packt Publishing Ltd., 2016. ISBN 978-1-78646-437-8.
8. GOLDBERG, Robert P. *Architectural Principles for Virtual Computer Systems*. Massachusetts: Harvard University, 1973.
9. *Hypervisor types*. [Online] Publikováno 2012. [cit. 2020-01-05]. Dostupné z: <http://www.virtualcomputer.com/type-1-vs-type-2-hypervisor>.
10. *What is a cluster?* [Online] Princeton University. [cit. 2020-01-05]. Dostupné z: <https://researchcomputing.princeton.edu/faq/what-is-a-cluster>.
11. *How IT Process Automation is Opening Opportunities for Virtualization*. [Online] ayehu, 2017. [cit. 2020-01-05]. Dostupné z: <https://ayehu.com/process-automation-opening-opportunities-virtualization/>.
12. *The 2020 State of Virtualization Technology*. [Online] Spiceworks, 2019. [cit. 2020-01-09]. Dostupné z: <https://www.spiceworks.com/marketing/reports/state-of-virtualization/>.

13. *What is OpenStack?* [Online] Red Hat, Inc., 2020. [cit. 2020-01-09]. Dostupné z: <https://www.redhat.com/en/topics/openstack>.
14. *OpenStack Components*. [Online] OpenStack project, 2020. [cit. 2020-01-09]. Dostupné z: <https://www.openstack.org/software/project-navigator/openstack-components#openstack-services>.
15. *Introduction to Terraform*. [Online] HashiCorp, Inc., 2020. [cit. 2020-01-09]. Dostupné z: <https://www.terraform.io/intro/index.html>.
16. *What is vRealize Automation*. [Online] VMware, Inc, 2020. [cit. 2020-01-09]. Dostupné z: <https://www.vmware.com/products/vrealize-automation.html>.
17. *VMware vRealize Automation Datasheet*. [Online] VMware, Inc., 2017. [cit. 2020-01-11]. Dostupné z: <https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/products/vrealize/vmware-whats-new-vrealize-automation.pdf>.
18. *VMware vRealize Suite and vCloud Suite: Licensing, Pricing, and Packaging*. [Online] VMware, Inc., 2016. [cit. 2020-01-11]. Dostupné z: <https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/vcloud-suite-pricing-packaging-whitepaper.pdf>.
19. *Product Licensing Metric Reference Guide*. [Online] VMware, Inc., 2020. [cit. 2020-01-11]. Dostupné z: <https://www.vmware.com/cz/support/support-resources/licensing/product-licenses.html>.
20. SCHOLTEN, Erik. *vRealize Automation 7 - Improved Authentication*. [Online] VMGuru, 2018. [cit. 2020-01-15]. Dostupné z: <https://vmguru.com/2015/10/vrealize-automation-7-whats-new-with-authentication/>.
21. RAJENDRAN, Sriram. *Learning VMware vRealize Automation*. Birmingham: Packt Publishing Ltd., 2016. ISBN: 978-1-78588-583-9.
22. NUKULA, Arun. *RabbitMQ in vRealize Automation*. nukesccloud. [Online] 2019. [cit. 2020-01-15]. Dostupné z: <https://www.nukescloud.com/post/2019/01/09/rabbitmq-in-vrealize-automation>.
23. PARMAR, Mayur. *vRealize Automation Overview*. [Online], 2019. [cit. 2020-01-15]. Dostupné z: <https://masteringvmware.com/vrealize-automation-overview/>.

24. POWELL, J. *Mastering vRealize Automation 6.2*. Birmingham: Packt Publishing Ltd., 2015. ISBN 978-1-78217-377-9.
25. *Architecture of vRealize Automation*. vRA 7 Book. [Online] Söldner Consult GmbH, 2020. [cit. 2020-01-18]. Dostupné z: <https://cloudadvisors.net/vra-buch/kap-2/>.
26. KELLY, Ryan. *vRA7 to vRO Extensibility Magic Secrets Revealed*. [Online] vmtocloud blog, 2016. [cit. 2020-01-18]. Dostupné z: <http://www.vmtocloud.com/vra7-to-vro-extensibility-magic-secrets-revealed-part-1/>.
27. *Working with vRealize Orchestrator*. vRA 7 Book. [Online] Söldner Consult GmbH, 2020. [cit. 2020-01-18]. Dostupné z: <https://cloudadvisors.net/chapter-21-working-with-vrealize-orchestrator/>.
28. *VMware vRealize Automation Design*. vRA 7 Book. [Online] Söldner Consult GmbH, 2020. [cit. 2020-01-19]. Dostupné z: <https://cloudadvisors.net/vra-buch/kap-3/>.
29. *Reference Architecture Version 6.0 and Higher: Technical white paper*. [Online] VMware, Inc., 2014. [cit. 2020-01-22]. Dostupné z: <https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/whitepaper/products/vcloud/vrealize-automation-6x-reference-architecture-white-paper.pdf>.
30. *Reference Architecture: vRealize Automation 7.6*. [Online] VMware, Inc., 2019. [cit. 2020-01-22]. Dostupné z: <https://docs.vmware.com/en/vRealize-Automation/7.6/vrealize-automation-76-reference-architecture.pdf>.
31. *vCloud Architecture Toolkit for Service Providers Overview*. [Online] VMware, Inc., 2020. [cit. 2020-01-22]. Dostupné z: <https://www.vmware.com/solutions/cloud-computing/vcat-sp.html>.
32. ROBERTS, Adrian a FEROCCE, Danilo. *Determining an Appropriate vCPU-to-pCPU Ratio*. VMware vCloud®Architecture Toolkit™for Service Providers. [Online] 2015. [cit. 2020-01-22]. Dostupné z: <http://download3.vmware.com/vcat/vmw-vcloud-architecture-toolkit-spv1-webworks/index.html#page/Core%2520Platform%2FArchitecting%2520a%2520vSphere%2520Compute%2520Platform%2FArchitecting%2520a%2520vSphere%2520Compute%2520Platform.1.019.html%23>.

33. *NetApp RAID-DP*. NetApp website. [Online] NetApp, 2020. [cit. 2020-01-23]. Dostupné z: <https://www.netapp.com/us/products/platform-os/raid-dp.aspx>.
34. *Mzdový průzkum 2019: Trendy na pracovním trhu v České republice*. Hays website. [Online] Hays Czech Republic s.r.o., 2019. [cit. 2020-01-25]. Dostupné z: https://cloud.email.hays.com/cz_salary_guide.
35. *vRealize Automation API Documentation*. VMware Docs. [Online] VMware, Inc., 2019. [cit. 2020-01-26]. Dostupné z: <https://docs.vmware.com/en/vRealize-Automation/7.6/com.vmware.vra.api-stub.doc/GUID-38318A70-5FD2-4C3C-A1CC-7DE678E03A25.html>.
36. HOREJŠÍ, Bronislava, a další. *Mikroekonomie*. Praha: Management Press, s.r.o., 2012. ISBN 978-80-7261-218-5.
37. SYNEK, Miloslav a KISLINGEROVÁ, Eva. *Podniková ekonomika*. Praha: C.H. Beck, 2010. ISBN 978-80-7400-336-3.
38. *Total Cost of Ownership (TCO)*. [Online] Gartner, 2019. [cit. 2020-01-30]. Dostupné z: <https://www.gartner.com/it-glossary/total-cost-of-ownership-tco>.
39. ROBERTS, Trevor, DOORN, Yvo van, SIGLER, Egle a ATWELL, Josh. *DevOps for VMware administrators*. Upper Saddle River, NJ: VMWare Press, 2015. ISBN 978-0-13384-647-8.
40. NOVAK, Karel, VALSECCHI, Paolo, MAURO, Andrea, GAVANDA, Martin. *Mastering VMware vSphere 6.7 – Second Edition*. Birmingham: Packt Publishing, 2019. ISBN 978-1-78961-337-7.
41. EISNER, Matthias, SANDFORT, Yves. *Learning vRealize Automation 7.2* [Online]. Birmingham: Packt Publishing, 2017. [cit. 2020-01-17]. Dostupné z: https://learning.oreilly.com/videos/learning-vrealize-automation/9781787281882/9781787281882-video1_1
42. *Total Cost of Ownership (TCO) - celkové náklady spojené s vlastnictvím*. [online] MANAGEMENTMANIA.COM LLC, 2018. [cit. 2020-01-30]. Dostupné z: <https://managementmania.com/cs/total-cost-of-ownership>.