

TALLINNA TEHNIKAÜLIKOOL  
Infotehnoloogia teaduskond

Peeter Rõbakov 103712IAPB

**SÜSTEEMIADMINISTRAATORI  
IGAPÄEVATÖÖDE AUTOMATISEERIMINE  
VMWARE NÄITEL**

Bakalaureusetöö

Juhendaja: Marko Kääramees  
PhD

Tallinn 2019

## **Autorideklaratsioon**

Kinnitan, et olen koostanud antud lõputöö iseseisvalt ning seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on töös viidatud.

Autor: Peeter Rõbakov

21.05.2019

## Annotatsioon

Töö põhieesmärgiks on välja töötada vastava riigiasutuse jaoks töötav lahendus, millega saaks vähendada süsteemiadministraatorite igapäevatööd puudutavaid küsimusi.

Antud lõputöö raames lahendatakse ära kaks probleemi. Probleemiks on virtuaalmasinate käsitsi tegemine. See on aeganõudev ja veaaldis. Selle jaoks tuleb leida lahendus automatiseerimise näol, mis teeb virtuaalmasinad automaatselt õigete seadistusega.

Teise probleemina vaadeldakse virtuaalmasinate hetketõmmiseid ehk *snapshotte*, miks neid olla ei tohiks. Lahendusena arendatakse välja skript, mis eemaldab *snapshotid*.

Peale probleemide lahendamise, antakse töös ülevaade virtualiseerimist. Sellega kaasnevad plussid ja miinused, suuremad teerajajad. Lisaks räägitakse *vRealize Automation-i* arhitektuurist ja komponentidest.

Antud tarkvaraga lahendatud probleemide tagajärjel selgus, et tegu on väga võimeka töövahendiga ja sobiks autori arvates antud riigiasutusse mõningate kriteeriumite täitmisel.

Teise probleemi lahendust sobib antud keskkonda väga hästi ja täidab oma eesmärgi.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 32 leheküljel, 7 peatükki, 27 joonist, 2 tabelit.

## **Abstract**

### **Automating system administrator everyday tasks using VMWARE**

The main objective of this thesis is to automate system engineers everyday tasks. This is done by powershell script and a software named vRealize Automation Appliance.

The thesis is based on two problems in everyday administrating. First problem is that system administrators have to create virtual machines by hand. It can cause humanly errors like changing to wrong network card, or makin storage the wrong size. This thesis will try to solve that problem, by automating the creation of virtuaal machines.

Second concern is snapshot management. Users have the privilege make them, but most of them forget they exist. The solution is to create a script that will find every snapshot in the datacenter that is X days old, and then delete it.

The thesis will also cover overview of virtualization. The pros and cons. Giving insights to the the architecture and components of vRealize Automation Appliance.

After solving the problem with vRealize Automation Appliance the author concluded that the given software may be used in the company, but have to fulfill some of the criterias established inside the company.

On the ohter hand the solution with the snapshots worked very well and is easy to implement, so I would suggest using it.

The thesis is in Estonian and contains 32 pages of text, 7 chapters, 27 figures, 2 tables

## Lühendite ja mõistete sõnastik

API	Application programming interface
DEM	Distributed Execution Managers
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
FQDN	Fully qualified domain name
IaaS	Infrastructure as a Service
IIS	Internet Information Service
IPAM	IP address management
LDAP	Lightweight Directory Access Protocol
OVA	Open Virtual Appliance
OVF	Open Virtualized Format
URL	Uniform Resource Locator
VRA	Virtual Realize Automation
VRO	Virtual Realize Orchestrator
XaaS	Anything as a service
API	Application programming interface
DEM	Distributed Execution Managers

## Sisukord

1 Sissejuhatus .....	10
2 Virtualiseerimisest üldiselt .....	12
3 Ülevaade arhitektuurist.....	15
3.1 Komponentidest detailsemalt .....	16
3.1.1 vRealize Application Appliance.....	16
3.1.2 <i>IaaS</i> komponendid.....	17
3.1.3 <i>Authentication services</i> .....	19
3.1.4 <i>vRealize Orchestrator</i> .....	19
3.2 Füüsiline serveri pool .....	20
4 vRealize automation appliance tööle panemine. ....	21
4.1 Ülesseadmine.....	21
4.1.1 Esialgse info kogumine .....	21
4.1.2 Esimene sisse logimine ja <i>tenant</i> -i loomine .....	22
4.1.3 Domeeni lisamine aplikatsiooni .....	25
4.2 Ressursi ja gruppide määramine.....	25
4.2.1 Endpoint määramine.....	25
4.2.2 Fabric group loomine .....	26
4.2.3 Sünkroniseerimine vCenteriga .....	26
4.2.4 Machine prefix.....	26
4.2.5 Network profile .....	26
4.2.6 Business groups .....	28
4.2.7 Reservations .....	28
5 Virtuaalmasinate tegemise automatiseerimine. ....	30
5.1 vCenter mallide tegemine.....	30
5.1.1 Linuxi masina mall tegemine .....	30
5.1.2 Windowsi masina mall tegemine.....	31
5.2 Jooniste tegemine .....	31
5.2.1 Virtuaalmasina ehitamise valikud .....	32
5.2.2 Custom Properties .....	34

5.2.3 VM customization specification .....	35
5.3 Catalog management .....	35
5.3.1 <i>Services</i> .....	35
5.3.2 <i>Catalog Items</i> .....	36
5.3.3 <i>Actions</i> .....	36
5.3.4 Entitlements .....	36
5.4 <i>Service catalog</i> .....	37
6 Snapshottide kustutamine .....	41
6.1 Snapshottide tegemine .....	41
6.2 Snapshottide ohtlikkus.....	41
6.3 Snapshoti eemaldamine .....	42
7 Kokkuvõte .....	44
8 Kasutatud kirjandus .....	45
Lisa 1 – [Lisa pealkiri].....	46
Lisa 2 – [Lisa pealkiri].....	47

## Jooniste loetelu

Joonis 1. Tavaarhitektuur ja virtualiseeritud arhitektuur [1] .....	12
Joonis 2. Serverite peale installeeritavate firmade osakaal. [2].....	14
Joonis 3. Minimal VRA Deployment. [3] .....	15
Joonis 4. IaaS komponentide omavaheline suhtlus. [1] .....	17
Joonis 5. Model Manager ja DEM suhtlus. [1].....	19
Joonis 6. Vaade veebilehele minnes. ....	22
Joonis 7. VMware vRealize Automation Appliance management settings.....	22
Joonis 8. Tenant rollide ülevaade. [4].....	23
Joonis 9. <i>IaaS administrator</i> ja <i>Fabric administrator</i> . [5] .....	25
Joonis 10. Network range määramine. ....	27
Joonis 11. vRealize Automation IPAM.....	28
Joonis 12. Reservations. ....	29
Joonis 13. Skripti tööle panemine. ....	30
Joonis 14. vRA loodavate masinate valik.....	32
Joonis 15. Design Canvas.....	32
Joonis 16. Custom property, IP valideerimine. ....	34
Joonis 17. IP näitamine.....	34
Joonis 18. Custom specification .....	35
Joonis 19. Kataloogi haldamine.....	37
Joonis 20. Service Catalog. ....	38
Joonis 21. Masina ressursi täpsustamine. ....	38
Joonis 22. Masina provisjoneerimine. ....	39
Joonis 23. Masina täpsem provisjoneerimine.....	39
Joonis 24. Edukas deployment. ....	39
Joonis 25. <i>vCenteris</i> nähtav tellitud masin. ....	40
Joonis 26. Väljavõte skriptist.....	42
Joonis 27. Snapshotide eemaldamine .....	43



## **Tabelite loetelu**

Tabel 1. vRealize Automation Hardware Specifications [10] .....	47
Tabel 2. vRealize Automation Resource Capacity Maximums [10] .....	48

# 1 Sissejuhatus

Tänapäeva kiiretes ja muutuvates infotehnoloogia suundades on väga suur roll virtualiseerimisel. Kunagine arvamus, et füüsiline server, kuhu peale läheb üks operatsiooni süsteem on läbi. Virtualiseerimine on vallutanud maailma väga kiirelt. Vähe leiame selliseid asutusi ja firmasid mis ei kasuta seda. Valdav osa infrastruktuurist on põhinev virtualiseerimisel, kuna see võimaldab mitmekülgse ökonoomsuse ja paindlikkuse.

Siiani on olnud probleemid kui administraatori poolt on virtuaalmasin lohakalt tehtud. Mõni ketas on puudu, võrgukaart pole ühendatud. Seda kõike saab vähendada, kui võtta inim tegur ära. See töögi selle lõputöö põhiprobleemini.

Selle bakalaureusetöö põhieesmärgiks on välja töötada vastava riigiasutuse jaoks töötav lahendus, millega saaks vähendada süsteemiadministraatorite igapäevatööd puudutavaid küsimusi. Töös ülesseatav automatiseerimist edendav töövahend aitab kaasa virtualiseerimise järgmisele tasemele viimist, kus võetakse administraatori käest ära osade igapäeva murede lahendamine. Lõputöös ehitatus süsteem peaks olema hea indikaator, kas see sobib vastava riigiasutuse keskkonda, või on vaja mõelda alternatiivmooduseid kuidas neid probleeme lahendada.

Selle lõputöö käigus lahendatavad probleemid on:

- **Probleem 1:** Masinate virtualiseerimine praegusel kujul tähendab süsteemi sisse logimist, masina loomist, ISO külge panemist ja siis käsitsi manuaalse installi läbimist. See tänapäeva kiire tempo ja inim teguri tõttu enam ei kõlba. Läbi virtuaalmasinate automatiseerimise saab kasutaja logida sisse iseteenindusportaali oma domeenikontoga. Seal näeb mis virtuaalmasinaid on tal võimalik taotleda. Ilma administraatori sekkumiseta luuakse masin, mida on võimalik edasi seadistavalt teenuste nõuetele.
- **Probleem 2:** Virtuaalmasinaid on vaja hallata ja uuendada. Uuendamise protsess käib nagu kõikidel teistel masinatel, aga selleks et säilitada hetkemoment, enne

installatsiooni, on *snapshot*. Manuaalselt tehtavate *snapshottide* haldamisega peab tegelema kasutaja ise. Kustutamine on küll kasutajate enda teha, aga paratamatult ununeb see vahepeal ära. Põhjus miks *snapshottide* eemaldamine on oluline selgitan peatükis number 6.

Sissejuhatuse lõpetaks ülevaatega eesolevast tööst:

- Peatükis 2 antakse üldine ülevaade virtualiseerimist, mis see on, miks see nii laialdaselt levinud on, peamised suunavaldkonna edendajad.
- Peatükis 3 tuleb *vRealize Automation Appliance* (vRA) tarkvara arhitektuurist ja komponentidest täpsemalt juttu. Seal selgitatakse miks on mingit komponenti vaja ja mis on nende täpsemad ülesanded.
- Peatükis 4 käiakse läbi *vRealize Automation Appliance* esmased seadistused, vajalikud liidestused, et aplikasioon korralikult ja ootuspäraselt töötaks.
- Peatükis 5 võetakse käsile esimese probleemi lahendamine. Peatükis kirjeldatakse millised grupid on vaja luua, mis osad kokku panna, millised on spetsiaalsed väljad mida vaja täita, et asi töötaks.
- Peatükis 6 kirjutatakse lahti mis on *snapshot*, milleks seda kasutatakse ja miks see on halb kui see seisma jääb pikemaks ajaks. Lisaks eelnevale tullakse välja lahendusega kuidas saab *snapshottidest* lahti.

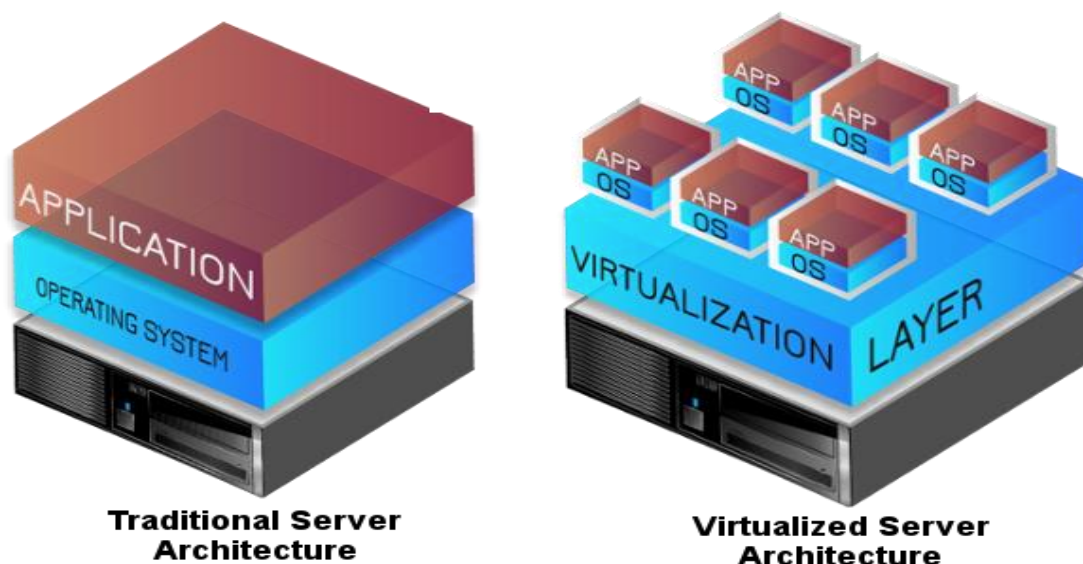
## 2 Virtualiseerimisest üldiselt

Enne kui me töö põhiosa juurde jõuame, kirjeldan ära mis on virtualiseerimine üldiselt.

Lihtsasti seletades on virtualiseerimine see suund kuhu tänapäeval liigutakse suure hooga, kuna see annab aina kiiremas tempos liikuvate infosüsteemidele võimalusele ressursse paremini majandada, liigutades neid vastavalt olukorrale. Ühe serveri peale saab panna mitu virtuaalmasinat tööle, ei ole vahet kas tegu on *Windowsi*, või *Linuxi* masinaga. Ressursside ümberjaotamist saab teha kiirelt, vastavalt olukorrale isegi masina töötades. Kõik see annab eelise ja mugavuse minna üle virtualiseerimise peale. Kui servereid on rohkem kui üks, saab need omavahel ühte loogilisse gruppi lisada, kus siis vastavalt ühe või mitmete serverite mittetöötamise puhul liigutatakse virtuaalmasinad üle teisele, kus need jälle tööle pannakse.

Virtualiseerimise eelised:

- Ühe serveri peale mahub mitu virtuaalmasinat. (Joonis 1). Sellega väheneb serverite koguarv. Serverite arvu vähenemisega pole vaja nii suurt jahutust. Lisaks kaasneb energia kokkuhoid.



Joonis 1. Tavaarhitektuur ja virtualiseeritud arhitektuur [1]

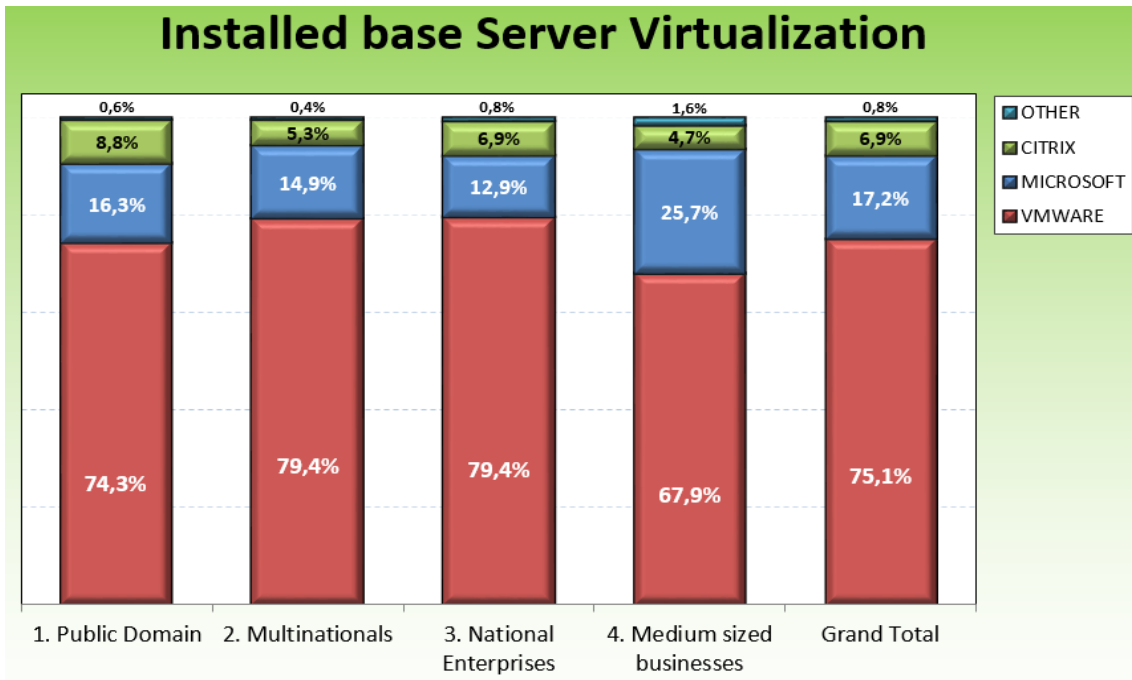
- Kiire taastamine – Virtuaalmasinad kasutavad kas erinevaid kolmanda osapoolse taasteprogramme, või snapshotte, kust on väga kiire taastamine.

- Kõrgkäideldavus- Reeglina suurtes firmades on füüsilisi servereid rohkem kui 1. Mitmete serverite saab kokku luua ühe loogilise grupi, kus ühe serveri mittetöötamine tagab virtuaalmasinate liikumise teisele serverile, kus see uuesti käima pannakse.
- Veel ühe positiivse asjana tooks välja selle, et väga kerge on luua testkeskkondi, kus saab arendada produktsiooni keskkonna jaoks välja töötavaid lahendusi.

Virtualiseerimise negatiivsed küljed:

- Litsentsi tasud – Paljud tootjad pakuvad esialgu 30 kuni 60 päevaseid proovi versioone, mille käigus lastakse kasutada kõiki võimalusi. See peaks kas huvi tekitama või mitte. Peale prooviaega peaks ostma litsentsid mis on päris kallid. Kuid ajaga lähevad litsentsid odavamaks, kuna konkurente tuleb järjest juurde.
- Uue infrastruktuuri õppimine- Iga uue asja ülesseadmine ja haldamine nõuab uusi teadmisi. Sellega kaasneb töötajate koolitamine, mis võib esialgse investeeringu suuremaks teha.

Kui praegusel ajal vaadata suuremate tarkvara firmade poole, siis virtualiseerimist kujundavad tänapäeval *VMware*, *Microsoft*, *KVM*, *Oracle*, *Citrix*. Kõik tegelevad suurelt, et arendada antud valdkonda. Kui veel 10 aastat tagasi hoidis *VMware* pea 80 % virtualiseerimise turust, siis aastal 2017 oli seis järgmine. (Joonis 2) . Tegemist on joonisega kus kujutatakse millist operatsiooni süsteemi on virtualiseerimisel kõige enim kasutatud.



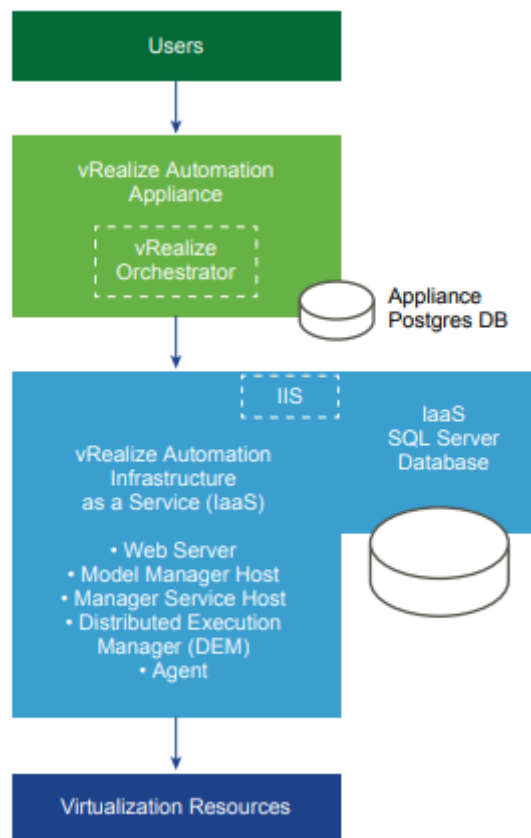
Joonis 2. Serverite peale installeeritavate firmade osakaal. [2]

### 3 Ülevaade arhitektuurist.

Antud peatükis antakse ülevaade järgmiste asjade kohta:

- vRA arhitektuuri komponendid;
- vRA arhitektuuriline ülesehitus;
- Elementide omavahelised seosed.

vRA Arhitektuur. (Joonis 3.)



Joonis 3. Minimal VRA Deployment. [3]

- *vRealize Automation Appliance* – tegemist on põhikomponendiga, mis tuleb *OVF* formaadis ja paigaldatakse *vCenter* serverisse. *vRA* tagab sisse logimise jaoks vajaliku iseteenindusportaali tagamisega, kus toimub masinate tellimine.
- *vRealize Automation Infrastructure as a Service (IaaS)* – Koosneb ühest või mitmest *Windows* serverist, mis töötavad koos, et modelleerida ja provisjoneerida

süsteeme vastavalt kas privaatses, avalikus või pilve infrastruktuuris. Sisaldab paljusid komponente, mis peavad ühendust saama hüperviisoriga. Kõiki toiminguid mida saab IaaS komponentidega teha, avalduvad infrastruktuuri liidese all. [3]

- *Authentication services* –Sisse ehitatud teenus, mis võimaldab ilma lisapistikprogrammita, või teise aplikatsioonita liidestuse *Active Directory*- ga *LDAP*-i kaudu.
- *vRealize Orchestrator* - Tsentraliseeritud/Detsentraliseeritud aplikatsioon, mis võimaldab töövoogude automatiseeritust.

### 3.1 Komponentidest detailsemalt

#### 3.1.1 vRealize Application Appliance

*vRealize Automation Appliance* puhul on tegemist *VMware* poolt automatiseerimist tagava tarkvaraga.

Nimetatud tarkvara koosneb alljärgnevatest elementidest:

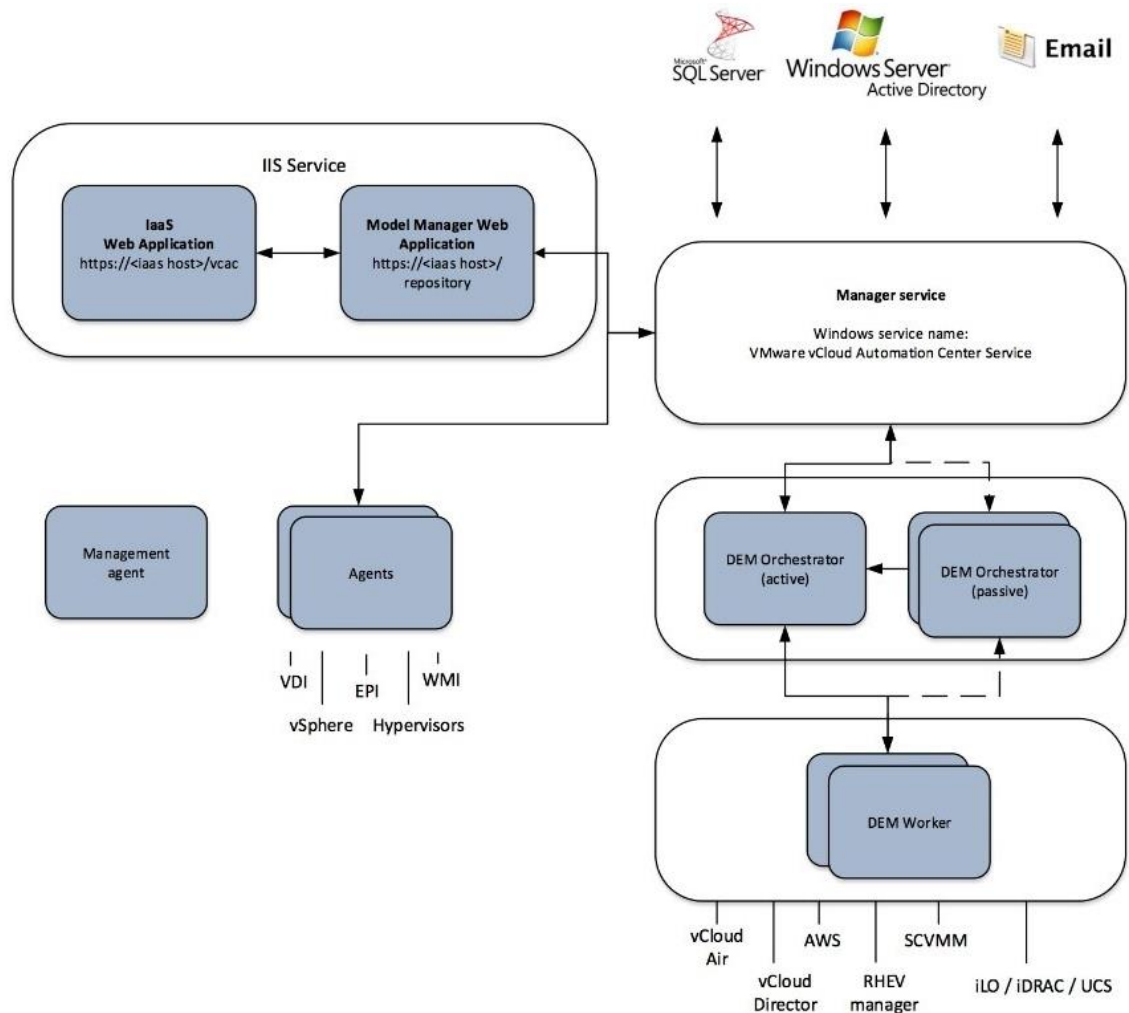
- *vRA* iseteenindus portaal (*SM*) – Tagab virtuaalmasinate tellimise funktsionaalsuse;
- *vRealize Orchestrator* – Võimaldab töövoogude läbi jooksumise funktsionaalsust;
- *vPostgreSQL* – Sisemine andmebaas;
- *tcServer* – Veebiserveri komponent;
- *RabbitMQ* [1] – Erinevate liideste põhine sõnumite vahetuse platvorm.

Lõputöö eesmärgi saavutamiseks seatakse üles minimaalsel kujul(*minimal deployment*) *vRealize Automation Appliance* rakendus. Nimetatud rakenduse skaleeritavus on vähene [3], kuid piisav, et pakkuda uurimisprobleemile lahendust.



### 3.1.2 *IaaS* komponendid

Kõik *IaaS* komponendid tuleb installida *Windows serveril* põhinevale masinale. Alustuseks ülevaatlik pilt (Joonis 4), peale mida hakkab ükshaaval tulema komponentide detailsem kirjeldus.



Joonis 4. *IaaS* komponentide omavaheline suhtlus. [1]

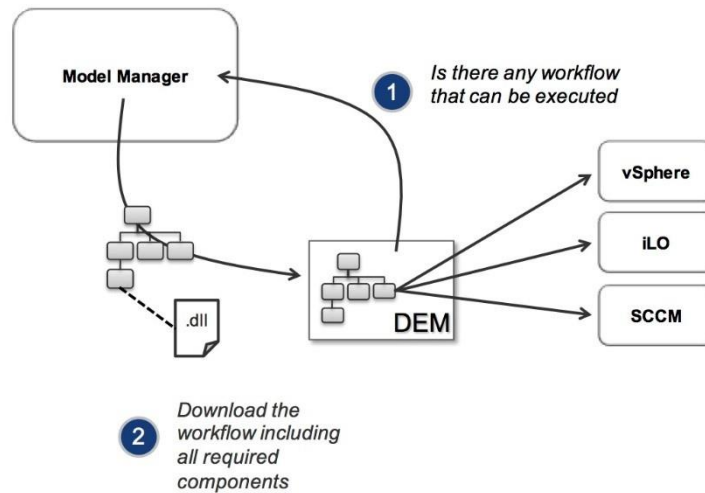
- *IaaS Website* - Tegemist on *IIS* veebiserveri rakendusel põhineva teenusega. Enamus kasutajavaadetest tagab *Linux*i aplikatsioon(*vRA*). *IaaS* komponentide osa, mis tagavad infrastruktuuri administreerimise, on võetud *Windows*i masinast. Lisaks suhtleb see *Model Manager*iga. [1]
- *Model Manager* – Seda võib nimetada *IaaS* komponentide südameks. *Model manager* ühendab ja vahendab infot erinevate komponentide vahel.

Põhilisteks ülesanneteks on:

- Tagada ühendus ning informatsiooni vahetus SQL serveriga;
- Edastada informatsioon töövoogude osas ning vajadusel käivitada töövoogusid erinevates välistes süsteemides. Antud lõputöö mõistes luuakse ühendus *vCenter* serveriga.
- Suhelda *Distributed Execution Manager -idega (DEM)* suhtleb täiendavalt IDE ning veebiliidesega.

Antud elemendi mõistes on kõrgkäideldavus võimalik tagada läbi implementatsiooni *Windowsi* tööjaama masinasse [1].

- *vCloud Automation Center Manager Service* – Selle komponendi puhul on tegemist tähtsa teenusega mis jookseb *Windowsi* masina peal. Tema peamised ülesanded on vahendada suhtlust *Model Manager*, andmebaasi, *vRA* agentide ja *Active Directory* vahel. Selle teenuse puhul pole midagi konfigureraida, lihtsalt peab veenduma, et see töötaks ja oleks administraatori õigused masinal kus jooksevad IaaS komponendid. [1]
- *IaaS MS SQL database* - Tegemist on andmebaasiga kuhu salvestatakse kõik andmed. Kui peaks tekkima soov seda kõrgkäideldavaks teha, siis *VMware* poolt on soovitatud *MS SQL* klaster. Selle paigaldamine toimub *Windowsi* masinale ja installatsiooni käigus tehakse sinna andmebaasi jaoks vajalikud tabelid automaatselt. Kõik *vRA* poolt hallatavate masinate, joonised(*blueprint*), kõik on seal. [1]
- *Distributed Execution Manager (DEM)*- Kuigi töövoogude loogika on hoitud *model manageris*, siis ei ole *model manager* see, kes neid töövooge käivitab. Selleks on *DEM*. *DEM*-id suhtlevad *model Manageriga*,(Joonis 5) saavad teada mis töövood on vaja käima panna ja seejärel käivitada. On kahte liiki *DEM*-e. Ühed mis planeerivad suhtlevad *model manageriga*, saavad sealt töövooge ja ajastavad need õigeaegse täitmise jaoks. Teised mis täidavad neid töövooge vastavalt kas *vSphere* või mõnes muus keskkonnas. [1]



Joonis 5. Model Manager ja DEM suhtlus. [1]

- *vCloud Automation Center management agent* –Vajalik osa Windowsi masinates, selle puudumisel ei suuda *vRA* aplikatsioon üles leida *Windows* server masinat. See agent ühendab osapooled omavahel ja tagab, et *vRA* saaks infot kätte *IaaS* komponentide kohta. *Windowsi* masinas jookseb sellele vastav teenus nimega *VMware vCloud Automation Center management agent*, mis on vajalik ühenduse hoidmiseks.

### 3.1.3 Authentication services

Üks uuemaid ja suuremaid muutusi võrreldes eelmiste versioonidega on aplikatsiooni sisse ehitatud autentimis teenus, mis teeb mugavaks selle liidestamise vastavalt soovile. Selle lõputöö raames kasutatakse domeeni nimega DK1.DEV. Kasutajate tegemine ja nende õiguste jagamine toimub nii *Active Directory*, kui ka *vRA* enda kaudu. Kõik kasutajad ja grupid mis eksisteerivad *Active Directory*-s saab nüüd sünkroniseerida *vRA*-ga. Selle teenuse olemasolul saab kaasata nüüd mitu domeeni ühe aplikatsiooni kohta. Kasutajate sünkroniseerimine käib käsitsi või vastavalt teatud aja tagant, mille saab määrata juba süsteemist seest. Sünkroniseerimine toimub *Active Directory*-st *vRA*-sse. *vRealize Automation*-isse luuakse ainult lokaalsed kontod. [1]

### 3.1.4 vRealize Orchestrator

*vRealize Orchestratori* näol on tegemist automatiseerimist lihtsustava töövahendiga. See on sisse ehitatud *vRA* sisse, kuigi saab üles seada täiesti eraldiseisvana. Rakenduses

jooksevad töövood, mis nõuavad sisendit ja tulemuseks on väljund, mis olenevalt olukorra võib olla virtuaalmasin või siis lihtsalt nimekiri soovitud. *vRA* kasutab *vRO*-d ära *Anything as a Service (XaaS)* jooniste valmistamisel, mis põhinevad täielikult töövoogudel. *Orchestratorit* ei pea ainult kasutama et luua uusi objekte, vaid on võimalik määrata ajaliselt toimuv sündmus kus siis saab muuta virtuaalmasina väärtusi, panna neid kinni, muuta nime ja paljusid asju veel. [1]

### 3.2 Füüsiline serveri pool

Antud alapeatükkis tutvustatakse, millist füüsilist konfiguratsiooni kasutatakse lõputöös pakutava virtuaalse lahenduse majutamiseks.

Infosüsteemi füüsiline ülesehitus:

- Server : *Cisco blade B200 M3*,
- CPU: *Intel Xeon E5-2600*,
- Mälumaht: 380 GB
- Operatsioonisüsteem: *ESXi 6.7 update 2*.
- Keskhalde: *vCenter Server Appliance versioon 6.7 update 2*,

## 4 vRealize automation appliance tööle panemine.

### 4.1 Ülesseadmine

#### 4.1.1 Esialgse info kogumine

Selleks et vRA saaks tööle panna on vaja see importida vCenter serverisse. Importimise ajal küsitav info on:

- Root password
- Hostname – vrealizeautomation.dk1.dev
- Default Gateway – 192.168.59.1
- Domain Name – dk1.dev
- Domain Name servers – 192.168.59.3
- Network1 IP address – 192.168.59.21
- Network1 Netmask – 255.255.255.0

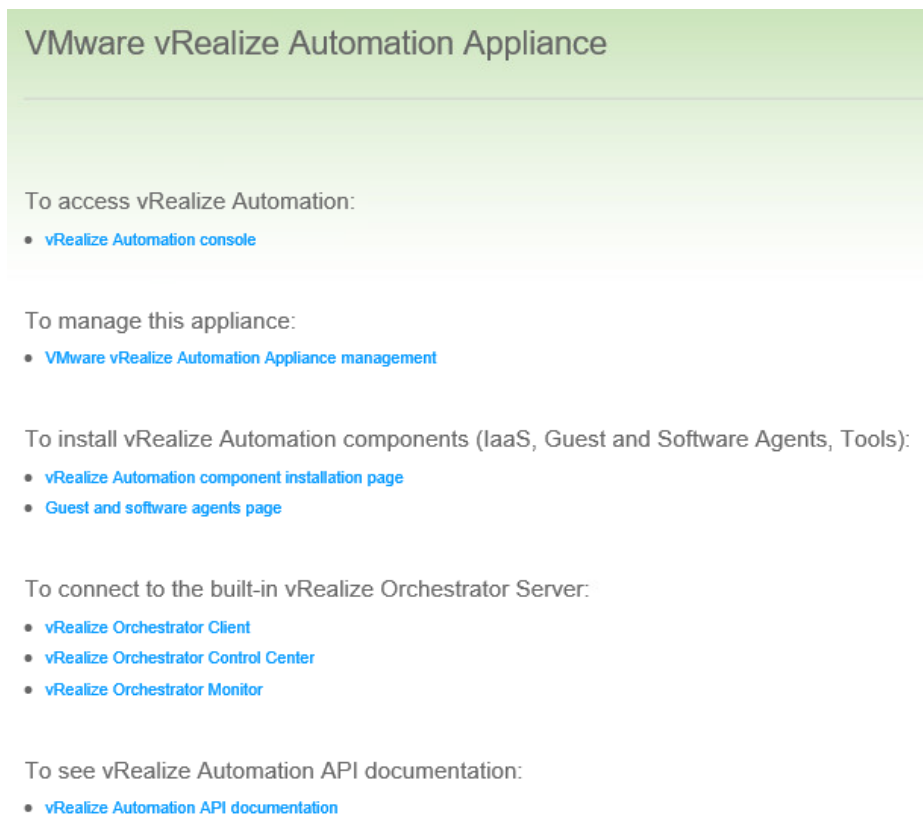
Pärast OVF faili edukat importi ja masina käivitamist, tuleb sisse logida *Windows* server arvutisse ja minna veebilehele [https://\(FQDN või Network1 IP\):5480](https://(FQDN või Network1 IP):5480). Et kasutada *FQDN*-i tuleb see enne defineerida DNS kirjete sekka. Antud lõputöö raames sai *FQDN* nimeks *vrealizeautomation.dk1.dev* Avanevas aknas hakatakse üksahaaval seadistama kõiki komponente. Selle protsessi käigus installeeritakse masinasse *vCloud Automation Center management agent*, mis loob ühenduse vRA-ga.

Seejärel kontrollitakse kas *IIS* server on olemas, *IaaS* komponendid töötavad, *SQL* server on olemas ja sisse logitav administraatori poolt. Esile kerkinud võimalikkude vigade esinemisel tuuakse selgelt välja mis on probleem ning millega tuleb tegeleda. Ülesseadmise ajal on vaja meelde jätta *vCenteri* jaoks määratud *endpoint*, mida hiljem kasutab süsteemis ära et luua ühendus *vCenteriga*. Viimaste asjadena genereeritakse *Self-signed* sertifikaadid.

Masinate hetkeolukorra jäädvustamiseks tuleks teha nendest *snapshotid*. Sellega tagame teatud seisu, kuhu saame alati tagasi pöörduda kui installi ajal midagi katki läheb. Peale edukat installatsiooni peab sisestama veel kehtiva litsentsi ja ongi kõik sellega.

#### 4.1.2 Esimene sisse logimine ja *tenant*-i loomine

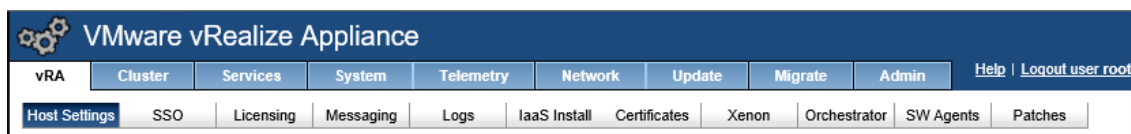
Kui minna <https://vrealizeautomation.dk1.dev/> peale siis avaneb vaade. (Joonis 6)



Joonis 6. Vaade veebilehele minnes.

Järgmiseks on vaja minna *vRealize Automation console* lehele, kus toimub edasine konfiguratsioon.

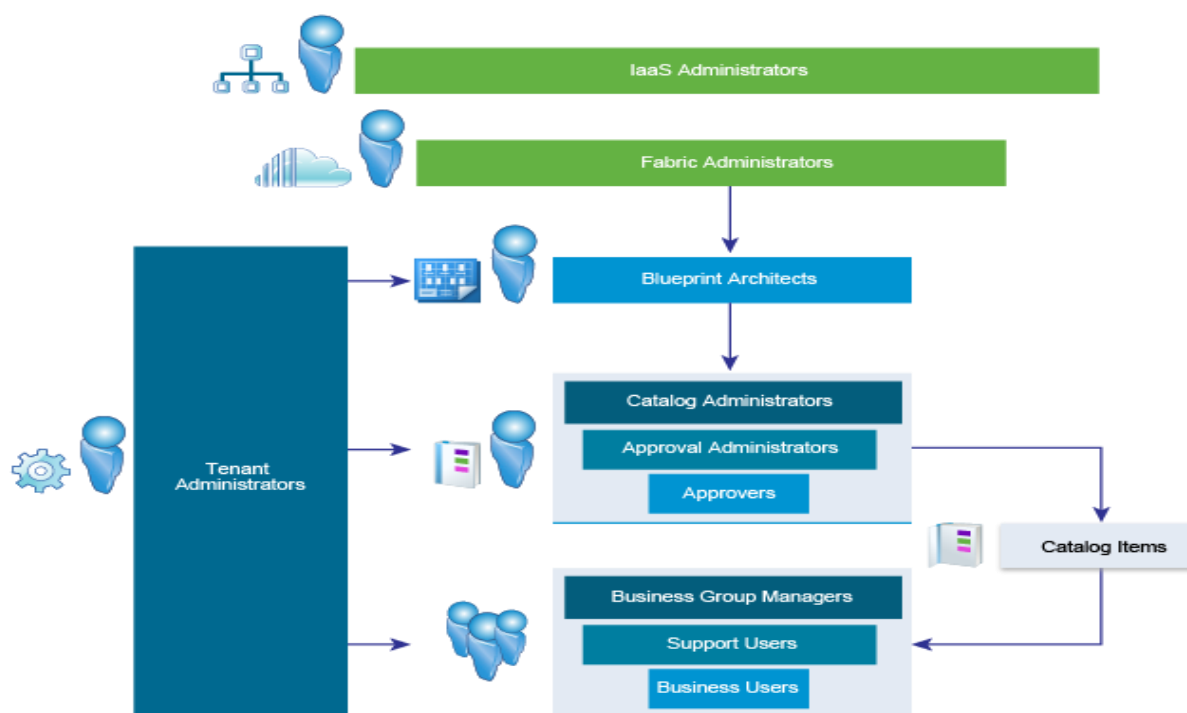
*VMware vRealize Automation Appliance management* – tegemist on kohaga kus toimub vRA seadete muutmine, litsentsi haldus, sertifikaatide haldus ja teised seaded.(Joonis 7)



Joonis 7. VMware vRealize Automation Appliance management settings.

Edasi toimub seadistus *tenanti* lisamise näol. *Tenant* näol on tegemist grupi inimestega, kellel on õigused toimetada antud tarkvara piires [5]. Lisame *tenant* grupi nimega DK1. Tenant grupp saab *vRA*-s eraldi sisselogitava nime. Seega iga loodud tenant grupp on unikaalne. Tenant grupid lisamisel ja sinna kasutaja tegemisel liidestame selle domeeniga. Seega on tegu kui ühe domeeni grupiga. Grupi loomisel palutakse määrata URL, mis määrab edaspidise sisse logimise aadressi.

Järgmiseks tuleb luua kohalik konto, mida kasutame, et sisse logida juba äsja loodud *tenant* gruppi. Kohalik konto tuleb määrata nii *Tenant* administraatoriks, kui ka *IaaS* administraatoriks. *Tenant* gruppide ja *vRA* gruppide kohta käib järgnev joonis. (Joonis 8)



Joonis 8. Tenant rollide ülevaade. [4]

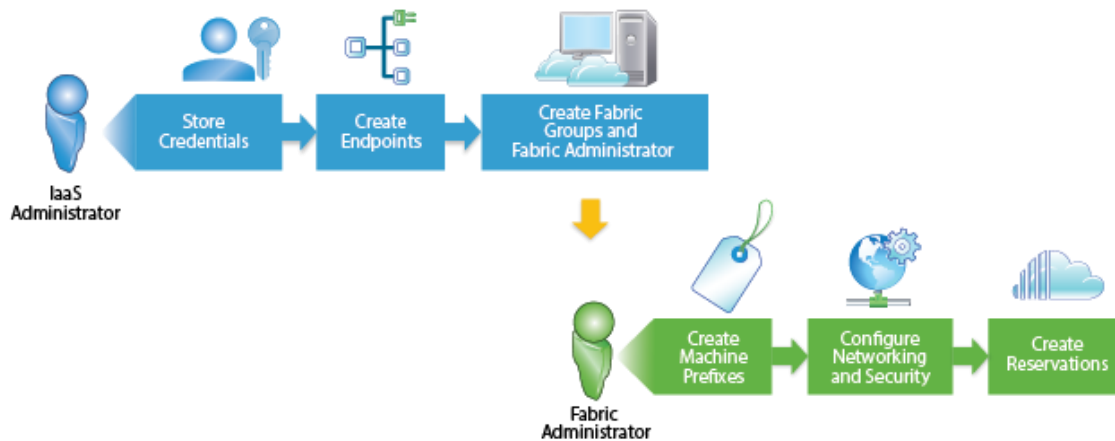
Rollid:

- *IaaS Administrator*- Tema haldab kõikide infrastruktuuri osadega süsteemi tasandil. Ülesanded on igast valdkonnast. võrgundusest, *endpointide* lisamised, logide lugemised, inimeste lisamine õigetesse gruppidesse, administraatorite määramine, et nemad tegeleksid oma valdkondadega juba edasi. (Joonis 9)
- *Tenant administrator*- Administraator kes seab süsteemi üles vastavalt organisatsiooni vajadustele. Nende hoolitseda on kasutajate ja gruppide haldus.

Lisaks sellele tuleb tegeleda ka äriloogika poolega, kinnitades virtuaalmasinate soove ja hoolitsedes ressurside eest.

- *Fabric Administrator* – Tegeleb füüsiliste masinate ja ressursi haldusega selle grupi piires, mis on neile määratud. Nende gruppide sees valmistavad ette reserveeringud. Lisaks sellele hoolitsevad nemad veel masinate eesliidest ja omaduste grupi üle.
- *Blueprint Architect*- Tegemist on isikutega kes teevad valmis jooniseid, mille järgi hakatakse siis kloonima või uuesti tegema masinaid.
- *Catalog Administrator*- loob ja haldab kataloogi majandusega. See tähendab, et loodud joonised mis tehakse, tuleb loogiliselt sorteerida eraldi kaustadesse.
- *Approval Administrator*- Tema tegeleb kooskõlastamise poliitika tegemisega, mis lähevad erinevate kataloogis olevad masinate külge.
- *Approver*- Tegemist on inimesega kes siis vastavalt poliitikale aktsepteerib või lükkab tagasi taotlusi. On kahte liiki taotlusi: sellised mis aktsepteeritakse enne kui masinat hakatakse tegema, ja teine peale masina valmimist.
- *Business group manager*- Tema tegeleb gruppide haldusega. Määrab õigusi vastavalt gruppide kaupa kes saab millele ligi. Nemad saavad taotleda masinaid teiste inimeste eest grupis.
- *Business user*- Tema on tavaline kasutaja kes kasutab iseteenindusportaal masinate taotlemiseks.





Joonis 9. IaaS administrator ja Fabric administrator. [5]

### 4.1.3 Domeeni lisamine aplikatsiooni

Järgmiseks sammuks on logida sisse uuel aadressil äsja loodud admin kasutajaga:

<https://vrealizeautomation.dk1.dev/vcac/org/DK1.DEV>

Seal saame lõpuks määrata *Active Directory* aadressi, et saaks kasutama hakata meie domeeni. Pärast domeeni lisamist leiame *Identity provideri* alt *WorkspaceIDP1*, mis on edukalt ära ühendatud meie *Active Directoriga* üle *LDAP-i*. Seal samas saame ära sünkroniseerida kõik kasutajad kes meie domeenis on. Seejärel oleks vaja logida vana admin kasutajaga uuesti sisse, ja uuesti määrata õigused nüüd domeenist tulevatele kasutajatele.

Kõige õigem ja soovituslik oleks teha iga grupile oma kasutaja, mida saaks pärast edasi kasutada sihtotstarbe pärast, kuid siin lõputöös määrasin ma ühele kasutajale kõik õigused, kes suudab kõiki ülesandeid täita. Lisaks administraatori kasutajale on seal veel paar testkasutajat, kelle kaudu saab demonstreerida, kuidas käib masinate tellimine läbi iseteenindusportaali, kui neil on ainult masinate tellimise õigus.

## 4.2 Ressursi ja gruppide määramine.

### 4.2.1 Endpoint määramine

*Endpoint* on lühidalt öeldes ühendus *vRA* ja *vCenter* serveri vahel. Tänu sellele ta avastab mis hostid on *vCenter* serverisse ühendatud, määramaks kui palju ressursi seal on, mida saaks kasutada, et masinaid proviisjoneerida.

*Endpointi* nimi peab kattuma punktis 4.1.1 mainitud *vCenteri endpoint* nimega.

#### 4.2.2 Fabric group loomine

Kui *vCenteri endpoint* on lisatud, tuleb luua *Fabric group*. See on grupp mis koondab enda alla ressursid. Vastavalt nii mitu hosti on *vCenter-isse* lisatud, saab kõik lisada ühte *Fabric* gruppi, või jagada gruppide vahel laiali, et oleks eri grupid eraldi administraatoritega.

#### 4.2.3 Sünkroniseerimine vCenteriga

Nüüd kui on olemas ressurss mida saame kasutada masinate tegemisel, tuleb see ära sünkroniseerida *vRA* aplikatsiooniga. Selleks tuleb minna *Compute Resource* alla ja sealt valida *host* millega sünkroniseerida. Selle tagajärjel teab nüüd *vRA* mis ressursi saab kasutada.

#### 4.2.4 Machine prefix

Masina ees-liideste loomine määratakse vastavalt grupile mida nad kasutavad. See aitab selgesti määrata mis grupi masinatega tegu on. Teiseks on olemas *custom property* nimega omadus, mille lisamisel kirjutatakse üle esimeses variandis määratud eesliides. *Custom property* kohta on eraldi tutvustav lõik

#### 4.2.5 Network profile

Järgmiseks võtame ette võrgunduse. Masinaid saab muidugi teha ka ilma võrgukonfiguratsioonita. Kui võrgu puhul kui ei täpsusta mis IP läheb mis masinale, siis määratakse sellele *DHCP*-ga automaatse IP. Selle lõputöö üks eesmärkidest oli masinale IP määramine kohe selle tegemise juures, seega tuleb teha üks *Network Profile*.

Selleks tuleb määrata: Nimi, kirjeldus, *subnet mask*, *gateway*, DNS ja lõpuks päris IP vahemikud mida saab see võrk kasutada. Selle lõputöö jaoks lõin võrguprofiili nimega „Serverite vahemik“ ja sinna sisse erinevaid IP vahemikke. IP vahemikke saab luua palju, kuid mitte väljapoole jäävad võrguaadressid. IP-de määramine toimub nii. (Joonis 10)

The screenshot shows the vRealize Automation interface. The top navigation bar includes 'Catalog', 'Deployments', 'Design', 'Inbox', 'Administration', 'Infrastructure', and 'Containers'. The left sidebar lists various management options like 'Recent Events', 'Managed Machines', 'Endpoints', 'Reservations', 'Network Profiles', etc. The main content area is titled 'Edit Network Profile - External' and shows a table of network ranges. The selected range '120-129' is highlighted, and its details are shown in a donut chart below the table.

Name	Description	Start IP	End IP
120-129		192.168.59.120	192.168.59.129
130-139		192.168.59.130	192.168.59.139
Vahemik		192.168.59.150	192.168.59.159
DHCP	DHCP annab ise automaa...	192.168.59.230	192.168.59.240

Details for Selected Range	
Name: 120-129	Allocated: 3
Description:	Unallocated: 7
Start IP: 192.168.59.120	Destroyed: 0
End IP: 192.168.59.129	Expired: 0
Total IP: 10	

Joonis 10. Network range määramine.

*Network range* määramisel tuleb määrata nimi, alguse IP ja lõpu IP. Ülejäänuga tegeleb vRA ise. Uue süsteemi ülespanekul võib kohalikust võrgust väljavõtte võtta CSV formaadis, mis annaks hetkeseisu, millised IP d on vabad millised ei ole. Seejärel selle sisse importides vRA-sse, teab *IPAM* arvestada sellega.

Aplikatsiooni on sisse ehitatud vRA enda teenus nimega *vRealize Automation IPAM*. (Joonis 11) Selle teenuse kaudu saame vaadata vRA seest, millised on kättesaadavad

IP-d. IP-d liigituvad seal kolme kategooriasse: *Allocated*, *Unallocated*, *Destroyed*.

IP Address ^	Machine or External Resource	Last Modified	Status
192.168.59.123	tava-2019-dom	5/12/2019 1:56:30 PM	Allocated
192.168.59.126	CentOS-7-tava	5/14/2019 2:46:58 PM	Allocated
192.168.59.129	Siim-2019	5/14/2019 9:24:28 PM	Allocated
192.168.59.131	tava-win2019nod	5/12/2019 2:44:38 PM	Allocated
192.168.59.139	Winser2019dom	5/14/2019 2:50:07 PM	Allocated

Joonis 11. vRealize Automation IPAM.

#### 4.2.6 Business groups

Kasutajad, kellel on ligipääs iseteenindusportaalile on vaja loogiliselt ära jaotada. Seda teeb *Business Group*. Grupile on nime ja kirjeldust. Seejärel lisame sinna kasutajad. Kasutajaid on mitut sorti. Nagu eelnevalt mainitud on grupil *Manager role*. Tema tegeleb masinate aktsepteerimine või siis tagasi lükkamisega. Teine tähtis roll on *User*. Tema on tavaline masina taotleja kellel siis läheb õnneks või mitte. Lisaks veel saab määrata *Support role* – kes aitab vajadusel kui on probleeme ja *Shared Access role*, on see kui keegi saab teiste tellitud masinaid kontrollida. Edasi minnes saame eelnevalt tehtud masina eesliidese nüüd selle grupiga siduda.

#### 4.2.7 Reservations

Ilma reserveeringuteta pole võimalik läbi *vRealize Automation-i* masinaid teha. Reserveeringu teeme eraldi *vSphere (vCenteri)* jaoks. Alustuseks tuleb määrata nimi, seejärel *business group*, mille just eelnevalt valmis tegime. Järgmises aknas on ressursside tabel, kus on *Compute Resource*, ehk siis hostid mille ressurssi me näeme ja mille peale virtuaalmasinad minema hakkavad. Me saame reserveeringu all määrata maksimaalse ja minimaalse masinate arvu mida tohib teha selle reserveeringu piires. Lisaks masinate arvule on vaja ära määrata mälu ja kettamahu piirangud. See tähendab olemasolevast ressursist kui palju võtame ära ja anname vastavale grupile. Kui mälu, masinate arvud ja kettamahud on määratud, tuleb lisada võrguprofiil. Seal

saab määrata tehtud võrguprofiili Serveri vahemik. Reserveeringute ülevaade on järgmine. (Joonis 12)

Reservations

Review information about the listed reservations. Machine count and memory usage include only machines that are powered on.

Advanced Search

+ New • Edit | Delete

Name ^	Ty...	Te...	Business Group	Pri...	Reserva... Policy	Machi... Total	Machines Allocated (%) (Alloc/Total)	Quota Allocated (%) (Alloc/Res)	Memory Allocated (%) (Alloc/Res)	Storage Allocated (%) (Alloc/Res)
Andmekeskus	vSp...	DK1...	Andmekeskus	1		5	100% (5 of 5)	0% (5 of Unlimited)	22% (22 GB of 100 GB)	35% (176 GB of 500 GB)
Tavakasutaja	vSp...	DK1...	Tavakasutaja	2	Uus	3	100% (3 of 3)	0% (3 of Unlimited)	32% (16 GB of 50 GB)	48% (120 GB of 250 GB)

Joonis 12. Reservations.

Kõik need tegevused mis ma eelnevalt ära kirjeldasin on vajalikud et luua piisav struktuur sellele et toetada edasiste probleemide lahendamist. Tegemist ei ole tarkvaraga mis peale ülesseadmist kohe tööle hakkab.

## 5 Virtuaalmasinate tegemise automatiseerimine.

Selles lõigus hakkame lahendama sissejuhatuses mainitud esimest probleemi. Probleem tuleneb just eelkõige mitte administraatori kiirest tempost, vaid sellest et kõik oleks sama malli järgi tehtud. Masinad oleksid samad. See tagab administraatori seisukohast parema probleemi otsimise.

### 5.1 vCenter mallide tegemine

Selleks, et saaks automatiseerida virtuaalmasinate tegemist on vaja põhja, mille pealt see masin tehakse. Selle jaoks tuleb teha *vCenter* serveris mall. Toetatud on nii Linuxi kui ka Windowsi operatsioonisüsteemid. Kõik algab sellest kui *vCenter*isse luuakse tavaline masin. Temast saab lõpuks mall, aga enne kui mall tehakse on vaja masin ära seadistada. Seadistamise all pean silmas *VMtools-ide* installeerimist ja ühe skripti käima panemine.

Tegemist on spetsiaalselt *vRealize Automation*-i jaoks loodud koodiga. Skripti leiab minnes veebilehele <https://vrealizeautomation.dk1.dev/software/> Valikus on kaks skripti. Üks on mõeldud *Linux*i masinatele, teine *Windows*i tüüpi masinatele. *Linux*i skripti nimi on: `prepare_vra_template_linux.tar.gz` ja *Windows*i skripti nimi `prepare_vra_template_windows.zip`. Skripti käimapanekul paigaldatakse automaatselt *Java Runtime Enviroment* 1.8.0 ja *vmware-vra-software-agent-service\_7.5*. Tänu sellele *API*-le saab masinatega liidestada tarkvara läbi *vRealize Automation*-i.

#### 5.1.1 Linux masina mall tegemine

Praeguse lõputöö käigus olen ma loonud ühe Linux masina põhja. Tegemist on *CentOS* 7 operatsiooni süsteemiga. Peale operatsioonisüsteemi installi ja käivitamist on vaja tööle panna eelnevalt mainitud skript. Eelnevalt mainitud skripti lehele minnes tuleb võtta skripti peal parem klikk ja kopeerida lingi aadress. Lisan väikse illustratiivse pildi milline on installatsiooni kulg.(Joonis 13.)

```
root@CentOS-7-tava ~]# wget https://vrealizeautomation.dk1.dev/software/prepare_vra_template.tar.gz
--2019-05-16 19:18:24-- https://vrealizeautomation.dk1.dev/software/prepare_vra_template.tar.gz
Resolving vrealizeautomation.dk1.dev (vrealizeautomation.dk1.dev)... 192.168.59.21
Connecting to vrealizeautomation.dk1.dev (vrealizeautomation.dk1.dev):192.168.59.21:443... connecte
d.
ERROR: cannot verify vrealizeautomation.dk1.dev's certificate, issued by '/C=EE/O=EDF/OU=IKT/CN=vrea
lizeautomation.dk1.dev':
  Self-signed certificate encountered.
To connect to vrealizeautomation.dk1.dev insecurely, use '--no-check-certificate'.
```

Joonis 13. Skripti tööle panemine.

.Kuna seda sertifikaati ei usaldata, siis peab lõppu määrama `--no-check-certificate`  
Kui sertifikaadist ollakse mööda saanud salvestatakse skript masinasse. Seejärel tuleb see lahti pakkida käsuga : `tar -xvf prepare_vra_mall_linux.tar.gz`. Seejärel tuleb skriptile õigusi lisada, et seda käivitada saaks käsuga: `chmod +x prepare_vra_mall.sh` .Lõpuks tuleb skript tööle panna käsuga `./prepare_vra_mall.sh`. Avanenud skriptis küsitakse *vRealize Automation* aadressi ja *Manager Services* aadressi.

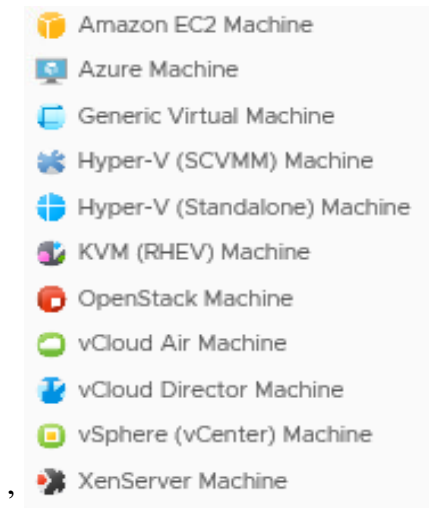
Pärast nende sisestamist küsitakse kas soovid installida *Javat*, mille peale tuleb vastata jah.. Sellega algab installatsioon, mille lõppedes võib masina kinni panna. Järgmisena tuleb *vCenter-is* võtta see masin, teha parem klikk, ja `template > convert to template`. Sellega oleme loonud endale Linuxi, täpsemalt öeldes siis *CentOS 7* masina, mida me saame võtta aluseks kloonimisel.

### 5.1.2 Windowsi masina mall tegemine

Peale Windowsi installi tuleb logida töötavasse masinasse sisse logida. Siinkohal peab ära mainima, et masin peab olema samas võrgus, kus asub vRA aplikatsioon. Järgmiseks tuleb minna <https://vrealizeautomation.dk1.dev/software/> veebilehele ja sealt valida vastav skript. Skript laetakse alla ja lahti pakkimisel avaneb neli faili. Et skripti käivitada peab olema kasutaja olema administraatori õigustes selles masinas. Seal on üks `.bat` fail ja üks `.ps1` fail. Avades `.bat` laiendiga faili algab install. Avanenud aknas küsitakse samu küsimusi mis *Linuxi* malli tegemise puhul. Installi lõppedes masin sulgeda konverteerida malliks.

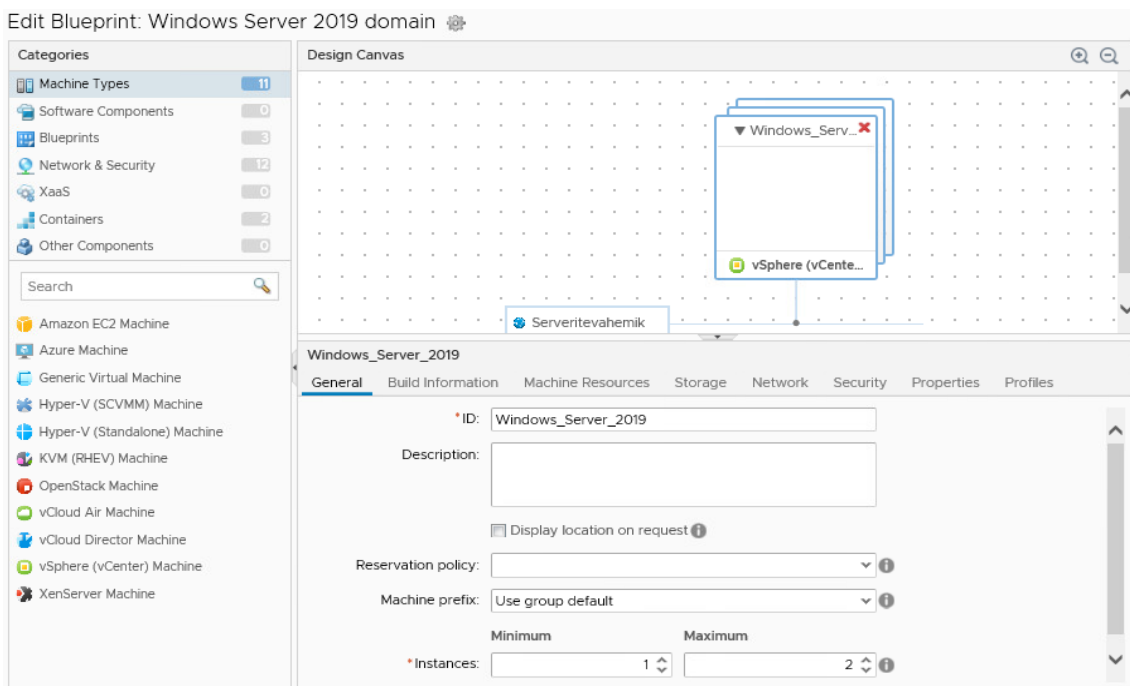
### 5.2 Jooniste tegemine

Masinaid ei saa luua tühja koha pealt. Kõigepealt tuleb minna *Design* lehele kus tuleb luua uus joonis(*blueprint*). Seda avades saab soovitud masinat lohistada lõuendile (*design canvas*) peale, kus hakatakse seda edasi konfigureerima. Valikuid millist masinat valida on päris lai. Valik masinatest (Joonis 14)



Joonis 14. vRA loodavate masinate valik.

Selles lõputöös kasutame *vSphere (vCenter) Machine*, kuna *endpoint*, mis sa varem loodud on just sinna ühendatud.



Joonis 15. Design Canvas.

## 5.2.1 Virtuaalmasina ehitamise valikud

Nagu eelolevalt olevalt pildilt näha võib olen loonud joonise nimega: *Windows Server 2019 domain*. Keskel *Design Canvase* nagu Joonis 15 peal on näha, et on lohistatud *vSphere* masin vasakust äärest pildi keskele. Lisaks virtuaalmasinale on pildile tõmmatud



võrguühendus. Kui keskendume lähemale keskmisele reale siis näeme erinevaid võimalusi mida on võimalik seadistada. Selgitan natuke lahti, mida igas kohas teha saab:

- *General* – Siin määrame selle joonise peal oleva masina nime, lisaks mitu masinat sellest üldse teha võib, alates ühest lõpetades mitmete masinatega. Selle välja juures võib täpselt ära defineerida kui peab tulema teatud arv masinaid, sest tavakasutaja neid sätteid muuta ei saa.
- *Build Information* – Siin määratakse ära kuidas masin luuakse. Erinevate võimaluste seast kasutame meie ära kloonimist. Kuna kasutame kloonimist siis on vaja masinat millest kloonida. Siin tuleb mängu eelnevalt juba valmis tehtud mall mida valides saame sellest täpse koopia. Viimase asjana on väli *Customization spec*, selle juurde jõuame punktis 5.2.3.
- *Machine Resources*- Siin toimub masina protsessorite arvu, mälu mahu ja kettapinna miinimum ja maksimum väärtuste andmine.
- *Storage* – Lisaks olemasolevale ketta partitsioonile saab neid siin juurde lisada. Kõige lihtsam viis on lihtsalt määrata maksimum ketaste arv mis saab sellel masinal olla.
- *Network* – Selles kohas toimub virtuaalmasina ja joonise peale tõmmatud võrgu ühendamise. Valida tuleb millist võrku soovitakse sellele masinale. Ühtlasi tuleb siin ära määrata *custom property* kui on selleks vajadus.
- *Security* - See osa on kõik seotus NSX iga, mida meil ei ole selles töös.
- *Properties* – Seda kohta saab kasutada kui soovitakse masinale mingeid omadusi juurde anda, siis näiteks kas vahetada masina nime.

Kuigi me tõmbame selle võrgu lõuendile, ei tähenda see veel võrguühendust masinaga. Nii võrgul, kui masinal on vaja määrata korrektsed parameetrid et ühendus toimuks. Selle masina juures olen ma lisanud veel spetsiaalse parameetri, ilma milleta ei oskaks vRA IP-d määrata. Täpsemalt lõigus 5.2.2.

## 5.2.2 Custom Properties

*Custom Properties* on omadus mida lisatakse masinatele lisa omaduste saavutamiseks. Eelneva näite puhul oli see *VirtualMachine.Network0.address*, mis annab sellele masinale sellise IP nagu on vormis kirja pandud. *Network0* määrab ära selle, et tegu on esimese võrgukaardiga. See custom property on tegelikult loodud juba varem, eraldi *Property Dictionary* all, kus valideeritakse IP õigsust valemiga [8]. Pildi peal on valem mille järgi seda tehakse (Joonis 16).

Edit Property Definition: Masina IP

Name: VirtualMachine.Network0.Address

\*Label: Masina IP

Visibility: All tenants

Description: Masina IP

Display order: [dropdown]

\*Data type: String

\*Required: Yes

\*Display as: Textbox

Validate user input: ^192\,168\,59\,([2-9])?([1-9])?([0-9])?([0-9])?2([0-3])?([0-9])?40)\$

Enter a regular expression to validate user input.

Test your validator

Joonis 16. Custom property, IP valideerimine.

IP valideerimist on hea kasutada, sest masinale vale IP andmine tähendaks viga ja masinat ei lasta taotleda.

Custom Properties

+ New | Edit | Delete

Name	Value	Encrypted	Overrida...	Show in Request
VirtualMachine.Network0.Address	192.168.59.	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

OK CANCEL

Joonis 17. IP näitamine.

Ilma *custom property* ja *Show in Request* linnukest lisamata, ei näeks IP määramise võimalust. (Joonis 17). Väli *value* võib ette ära täita administraatori poolt, või jätta poolikult täidetud. Linnukese koht *Overridable* tähendab seda, et kui linnuke on tehtud, siis saab kasutaja masina taotlemisel selle väärtuse üle kirjutada.

Teine *custom property* mida oleks hea lisada, on *hostname*. Selle omaduse lisamine kirjutab üle siis eelnevalt määratud grupi eesliidese ja määrab just selle nime, mille taotluses kirja pannakse.

### 5.2.3 VM customization specification

Tegemist on lisaseadistus võimalusi pakkuva profiiliga. See on *vCenter serveris* seadistatud ja peale masina proviisjoneerimist toimub lisa seadistus vastavalt profiilile. Väljas mida on vaja täita on Joonisel 18.

#### Windows Server 2019 domain - Editing

Name and target OS	
Registration information	
Computer name	Name Windows Server 2019 domain
Windows license	OS type Windows
Administrator password	OS options Generate new security ID
Time zone	Registration info Owner name: peeter Organization: DK1
Commands to run once	Computer name Use Virtual Machine name
Network	Product key No product key specified
Workgroup or domain	Server license mode Per Server (Maximum Connections: 5)
Ready to complete	Administrator log in Log in automatically as Administrator
	Automatic logons 1
	Time zone (UTC+02:00) Helsinki, Kyiv, Riga, Sofia, Tallinn, Vilnius
	Network type Standard
	Windows Server domain dk1.dev
	Username dk1\peeter

Joonis 18. Custom specification

## 5.3 Catalog management

Et kasutajad näeksid valminud jooniseid, on need vaja loogiliselt ära määrata erinevatesse gruppidesse. Catalog management jaguneb nelja osasse:

### 5.3.1 Services

Kõik joonised mis me *Design* aknas teeme, on vaja ära kategoriseerida. Seda siis vastavalt kas operatsioonisüsteemidele, osakondadele, isikutele isiklikult mõeldud grupid. Seda aitab teha meil *Services*. Seal me saame luua loogilise grupi kuhu paneme ühesugused

masinad kokku. Praeguse lõputöö raames on tehtud kaks *Service* -it nimega *Windows* ja *Linux*.

### **5.3.2 Catalog Items**

Sinna vaatesse lähevad kõik joonised mis on avalikuks tehtud . Joonised mis esialgu luuakse ei ole veel kõigile avalikud. Need on esialgu mustandid. Avalikuks saab need teha vastav administraator kellel on õigused.

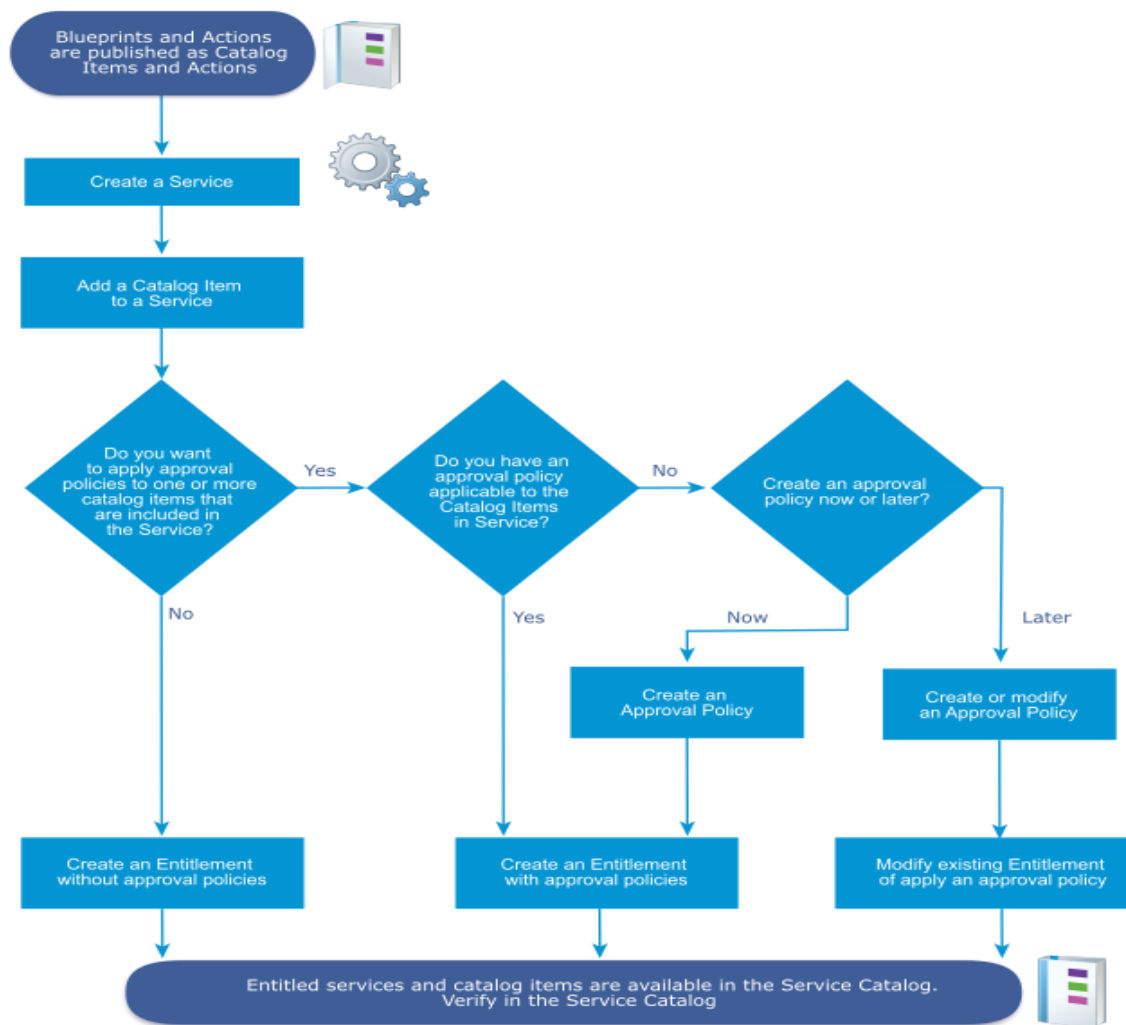
### **5.3.3 Actions**

Siin on üles mainitud kõik võimalikud tegevused mida on võimalik teha virtuaalmasinatega pärast seda kui need on valmis ja kasutajale kättesaadavad. Selle määrab ära *entitlements* grupp, millest on järgmises lõigus juttu. Nende gruppidega määratakse täpsed õigused ja kasutajad sinna gruppi.

### **5.3.4 Entitlements**

Väga oluline lüli kõige automatiseerimise juures. Siin luuakse grupid, kuhu määratakse kasutajad. Igal grupil on omad õigused, mille määrab administraator. Kui grupp on loodud tuleb sinna lisada inimesi. Seda saab teha *Active Directory-s* olevate gruppide või siis üksik indiviidide põhjal. Selle töö raames on tehtud kaks gruppi. Andmekeskus ja tavakasutajad. Ühes olen mina ja kõikide õigustega, teises on üks loodud kasutaja kellele on väga piiratud õigused. Gruppi luues lisatakse *business group*, kus oli ära määratud kindel kogus ressursi. Ühele grupile on võimalik määrata aegumistähtaeg, mis on sobilik kui tegu on projektidega. Peale tähtaegade *business group-i* ja kasutajate määramist tuleb edasi minna järgmisse aknasse, kus on valida nüüd millistele äsja loodud *service-* itele ligipääs anda ja mis õigustega. Grupile saab määrata nii terve *service* õigus, kui ka üksikutele joonistele.

Sellega piirdub *Catalog management* osa. Joonis 19 seletab väikse pildina kuidas asjad omavahel seotud on.

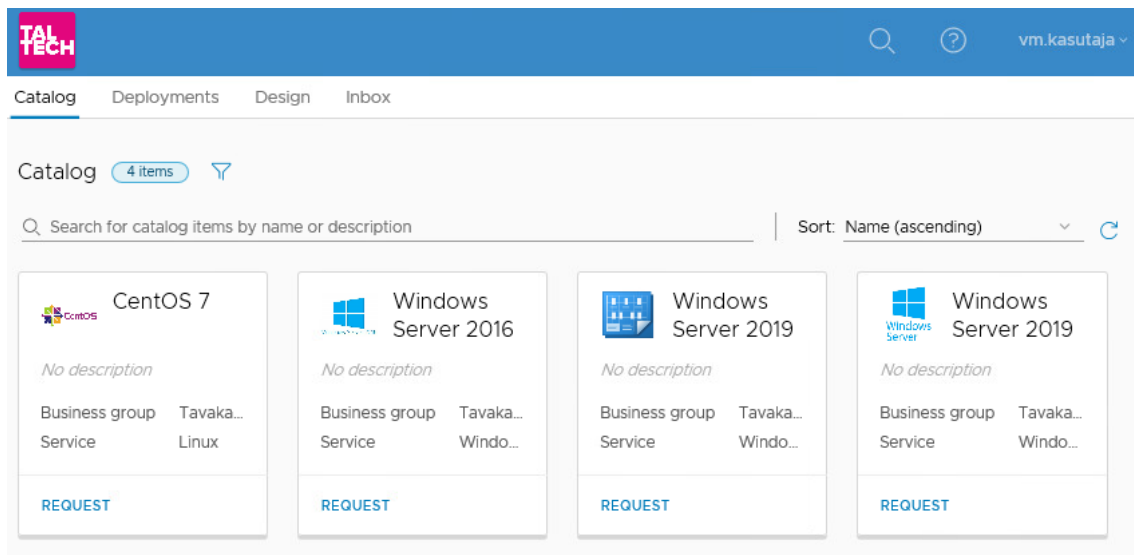


Joonis 19. Kataloogi haldamine.

## 5.4 Service catalog

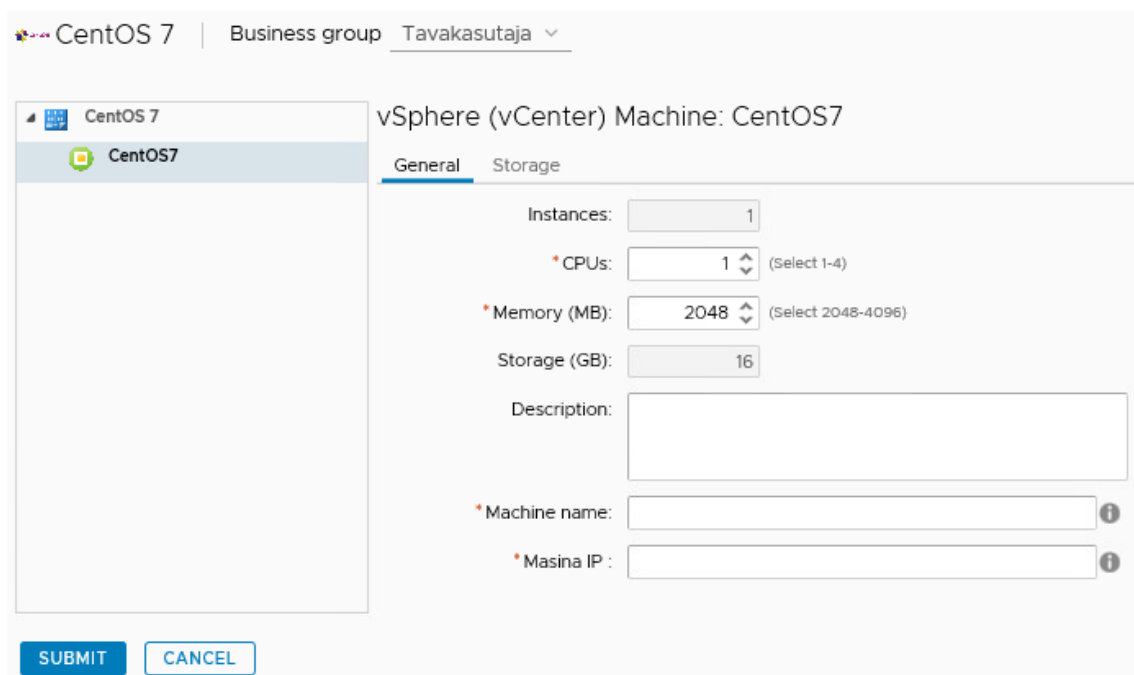
Siin hakkavad olema kasutajale nähtavad joonised mida ta saab taotleda. Need joonised on juba õigetesse kategooriatesse lisatud ja õigused on olemas. Masina taotlemiseks tuleb vajutada *Request*.

Joonis 20 kuvab vaadet mida tavakasutaja näeb sisse logides



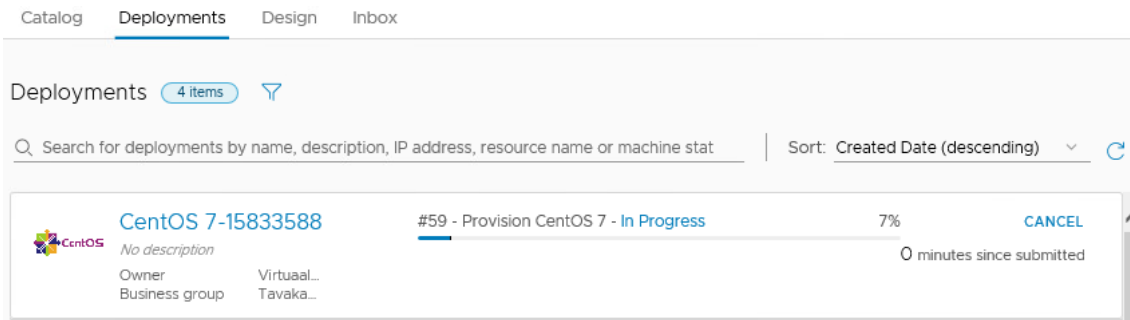
Joonis 20. Service Catalog.

*Request* nuppu peale vajutades tuleb ära täita kõik nõutud väljad mis on ette antud. Praegu valisin välja *CentOS 7* ja vajutasin *request*. Tulemus mis sellega kaasneb on järgmine: (Joonis 21)



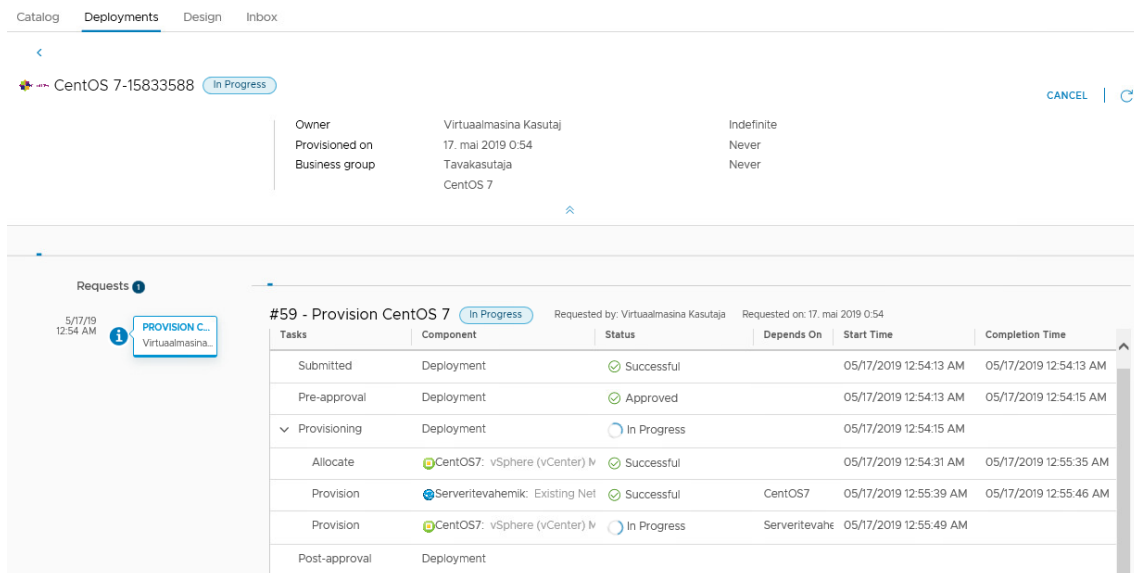
Joonis 21. Masina ressursi täpsustamine.

Peale *submit* nupu vajutamist tuleb selline ekraan: (Joonis 22)



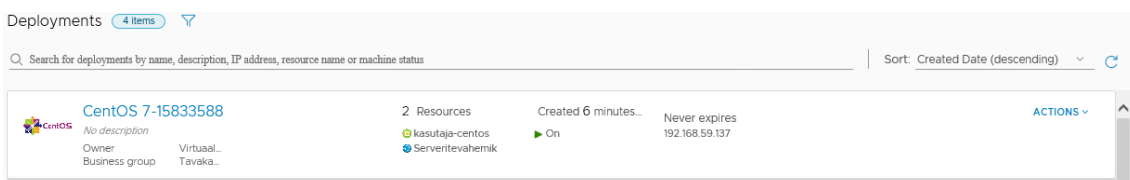
Joonis 22. Masina provisjoneerimine.

Kui vajutada nupule *In Progress*, siis näeb täpsemalt mis seal sees toimub ja kus maal on. (Joonis 23)



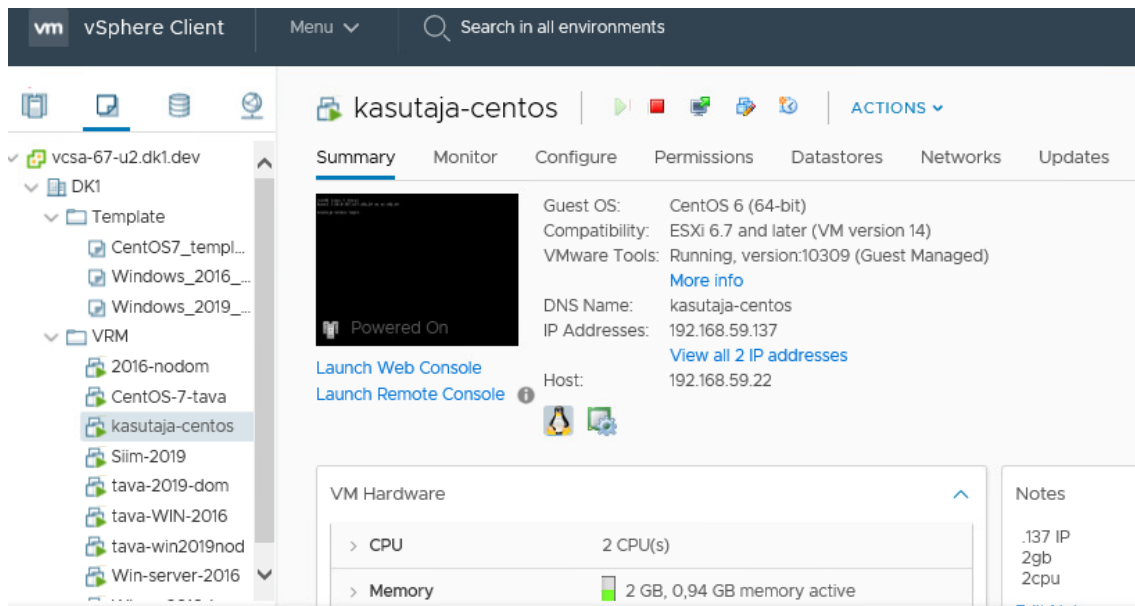
Joonis 23. Masina täpsem provisjoneerimine.

Joonisel 24 on kujutatud eduka masina provisjoneering.



Joonis 24. Edukas deployment.

Masina edukat valmimist kinnitab *vCenter*isse sisse logimine, kus kuvatakse õige masin korrektse IP ja *hostname*-iga. (Joonis 25)



Joonis 25. vCenteris nähtav tellitud masin.

Kokkuvõtvalt selle peatüki kohta: Kõik mured mis sissejuhatuses esimese probleemi kohta sai püstitatud, lahendati ära. Uus masin provisjoneeriti täpselt nende andmetega mis kasutajavormis ette antud. Tähtsad punktid mis tooksin selle peatüki puhul välja: Uute masinate puhul on määratud :

- Hostname, mis ühtlasi toimib kui DNS nimena. Domeenis olles lisatakse see kohe olemas olevate arvutite nimekirja.
- Mälumaht
- Protsessorite arv
- IP aadress

Need oleks kõige tähtsamad asjad mida tuleb vaadata masinaid tehes.



## 6 Snapshotide kustutamine

Administraatoritel on kujunenud mulje, et *snapshot* on üks varundamise võimalikest liikidest. Seda see kindlasti ei ole. [8] Üksikut *snapshot* ei tohiks hoida üle 72 tunni oma produktsiooni keskkonnas [8]. Sellest tulenevalt ka see probleem. Põhjus miks ei tohiks üks *snapshot* olla üle 72 tunni põhjendan punktis 6.2.

### 6.1 Snapshotide tegemine

*VMware* kasutab oma virtuaalmasinate hetketõmmise saamise *snapshotte*. *Snapshotte* saab teha nii töötavast masinast, kui kinnisest masinast. Töötavast masinast *snapshot* tehes on võimalus valida, kas tehakse koos mälua või ilma mälua. Vahe nende vahel on, et mälua koos tehes, saab minna *snapshot*id punkti tagasi reaalajas. ilma masinat kinni panemata. Ilma mälua tehes on vaja masin korra kinni panna ja siis uuesti tööle panna. Mälua ja mälua *snapshot*ides on kettamahu vahe, mälua koos tehes võetakse kõik mälu suurus *snapshot*ti sisse. Töötavast masina *snapshot*id tegemine võib olla natuke ohtlik, kuna talletada on vaja kogu hetkeseis, siis on vaja teatud ajaks peatada kogu masina töö. Kui tegemist on kriitilise masinaga süsteemis, mis ei kannata kõikumisi siis see võib olla suureks probleemiks.

### 6.2 Snapshotide ohtlikkus

*Snapshot*id muutuvad ohtlikuks, kui nad on seisnud pikka aega. Põhjus miks see on, peitub ülesehituses. *Snapshot*ti tehes tehakse uus fail, nii öelda ketas juurde masina konfiguratsioonile. Kõik edaspidi masinas muudetavad failid ei kirjutata enam põhikettale, vaid sellele *snapshot*id kettale. Kui nüüd peaks tagasimineku(*revert*), või kustutamise soov olema, siis kirjutatakse kõik uuel kettal olnud andmed esialgsele tagasi. Tegemist on aga ainult muudatustega, ehk algab väiksest kettamahust, kuid mida kauem on *snapshot* elus seda rohkem kirjutatakse, seda suuremaks see läheb. Sellepärast on teatud ajalimiit soovituslik. Kõige halvem on kui tegemist on andmebaasi virtuaalmasinaga. Selles masinas kirjutatakse väga palju ja see *snapshot* saab suureks. Tagasiminekul või *snapshot*id kustutamisel on vaja suur hulk andmeid jälle liigutada mis võib tekitada probleeme.

### 6.3 Snapshoti eemaldamine

Snapshotide eemaldamine on vaja tööandja poolt paika panna. Snapshoti tegemine on lihtne, kasutaja teeb oma uuenduse ära kuid läheb meelest kustutada. Seepärast võikski olla kindel arv päevi kaua see süsteemis olla võib. Selle lõputöö raames määrame me selleks ajaks 3 päeva. Kõik snapshotid mis on vanemad kui kolm päeva saavad kustutatud. Selleks olen kirjutanud välja väikse koodijupi mis ühendab ennast vCenter serveriga ja otsib üles kõik snapshotiga masinad. Seejärel hangib natuke lisainfot nende kohta ja kuvab ekraanile.(Joonis 27) Oma tööga olen kahjuks kokku puutunud neid kordi kus ei oleks tohtinud ära kustutada, seega praegu kuvatakse vastavat snapshoti mida kustutada on soov.

```
Name                               Port  User
---                               -
vcsa-67-u2.dkl.dev                 443   DK1.DEU\peeter

Kustutamisele kuulub:

Virtuaalmasina_Nimi : tava-2019-dom
Snapshot_Description : See on childsnapshot
Virtuaalmasin_Omanik : UM Kasutaja
Loodud              : 14.05.2019 20:05:36
Päevi_Uana          : 6

Confirm
Are you sure you want to perform this action?
Performing the operation "Removing snapshot." on target "VirtualMachineSnapshot-snapshot-54".
[Y] Yes [A] Yes to All [N] No [L] No to All [S] Suspend [?] Help (default is "y"): _
```

Joonis 26. Väljavõte skriptist.

```

$vCenter= "vcsa-67-u2.dk1.dev"
$vcCredentials = Get-Credential
Connect-VIServer -Server $vCenter -Credential $vcCredentials
$Vanus = 3
$allVMs = Get-VM
$snapshots = $allVMs | Get-Snapshot
$kustutamisele = 0
foreach ($snap in $snapshots) {
    if ($snap.Created -lt (Get-Date).AddDays(-$Vanus) ) {
        Write-Host ("`nKustutamisele kuulub: " ) -
        ForegroundColor Yellow
        $vmName = $snap.VM
        $vmOwnerTag = $vmName | Get-TagAssignment -Category "VM-
        Owner"
        $vmOwner = $vmOwnerTag.Tag.Name
        if ($vmOwner.Length -eq 0) {
            $vmOwner = "Puudub, palun lisada omanik"
        }
        $daysold = (Get-date).AddDays(0) - $snap.Created
        $out = new-object psobject
        $out | add-
        member noteproperty Virutaalmasina_Nimi $snap.vm.name
        $out | add-
        member noteproperty Snapshot_Description $snap.Description
        $out | add-
        member noteproperty Virutaalmasin_Omanik $vmOwner
        $out | add-member noteproperty Loodud $snap.Created
        $out | add-member noteproperty Päevi_Vana $daysold.Days
        Write-Output $out |Format-List
        Remove-Snapshot -Snapshot $snap
    }
}

```

Joonis 27. Snapshottide eemaldamine

Skripti käima pannes leitakse üles ükshaaval snapshotte. Väljavõte kuidas see välja näeb on joonisel 26.

## 7 Kokkuvõte

Kui ma seda lõputööd alustasin, olin ma väga elevel, et selline võimalus on *VMware*-l tekkinud ja ei ole veel väga laialdaselt levinud. Seda üles seades olin ma üllatunud kui keeruline selle ülesseadmine on. Proovima hakates muutus see üpris keerukaks päris kiiresti. Sellest olenevalt arvan, et saavutasin enda püstitatud põhieesmärgi. Leidsin lahenduse kuidas luua automatiseeritud süsteemi administraatoritele.

Esimeseks probleemiks püstitatud virtuaalmasinate loomine administraatori poolt sai lahendust automatiseerimisega. Automatiseerimise tulemusena luuakse virtuaalmasin, kus on küljes administraatori poolt kirjeldatud omadused, mille kasutaja peab ära täitma. Loodud masina atribuutidesse läheb virtuaalmasina tellinud kasutaja nimi.

Probleemi lahendusest lähtuvalt on võimalik väita, et *vRealize Automation Appliance* kasutamine on vastavale asutusele mõistlik, kuid teatud kriteeriumite täitmisel. Taotlemise juures on kõige tähtsam selgelt kehtestatud piirnõuded teenustele. Teiseks on vajalik argumenteeritud ressursi põhjendus. Kolmandaks on oluline esile tuua asutuste jaoks kehtestatud tingimused, et kõik taotlused, mis on üle sätestatud piiri on vajalik kontrollida määratud inimesel.

Teise probleemina välja toodud *snapshotide* kustutamine lahendati antud töö raames skripti loomisega, mis töötles läbi kõik andmekeskuses olevad virtuaalmasinad ning kontrollis, kas nende sees leidub *snapshotte*. Lahendus töötas väga hästi ja kõik üles leitavad *snapshotid* said kustutatud ilma probleemideta.

Lisaks probleemide lahendustele jõudis autor veel ühele arusaamale: on soovitatav asutusesiseselt välja mõelda ning rakendada poliitika kuidas tegeleda *snapshotidega*. See peab olema rangelt reglementeeritud, et administraatorid saaksid enda tööd teha efektiivselt ning kvaliteetselt.

## 8 Kasutatud kirjandus

- [1] „Virtualization,“ StepUP, [Võrgumaterjal]. Available: <https://www.stepupitservices.com/virtualization/>.
- [2] „VMWARE BY FAR THE LARGEST IN THE SERVER VIRTUALISATION MARKET,“ Smartprofile, [Võrgumaterjal]. Available: <https://www.smartprofile.io/analytics-papers/vmware-far-largest-server-virtualisation-market/>. [Kasutatud 20 Mai 2019].
- [3] VMware, „Vrealize automation 7.5 installation,“ [Võrgumaterjal]. Available: <https://docs.vmware.com/en/vRealize-Automation/7.5/vrealize-automation-75-installation-and-configuration.pdf>. [Kasutatud 28 April 2019].
- [4] „Cloudadvisors,“ [Võrgumaterjal]. Available: <https://cloudadvisors.net/vra-buch/kap-2/>. [Kasutatud 14 Mai 2019].
- [5] VMware, „Tenant ülevaade,“ [Võrgumaterjal]. Available: <https://docs.vmware.com/en/VMware-Validated-Design/4.1/com.vmware.vvd.usecases-lcm-deploy.doc/GUID-82D2A8CF-B8E2-4D5F-9142-2A417AFAD084.html>. [Kasutatud 20 Mai 2019].
- [6] VMware, „Tenant Role Overview,“ VMware, [Võrgumaterjal]. Available: <https://docs.vmware.com/en/vRealize-Automation/7.5/com.vmware.vra.prepare.use.doc/GUID-01F9D264-9940-4526-BC2D-B0553F080232.html>. [Kasutatud 13 Mai 2019].
- [7] „Checklist for Configuring IaaS Resources,“ VMware, [Võrgumaterjal]. Available: <https://docs.vmware.com/en/vRealize-Automation/7.5/com.vmware.vra.prepare.use.doc/GUID-88CC4331-0446-4E9D-9798-399AB29991A1.html>. [Kasutatud 12 Mai 2019].
- [8] L. S. Sterling, The Art of Agent-Oriented Modeling, London: The MIT Press, 2009.
- [9] VMware, „vRealize Automation Hardware Specifications and Capacity Maximums,“ [Võrgumaterjal]. Available: <https://docs.vmware.com/en/vRealize-Automation/7.5/com.vmware.vra.install.upgrade.doc/GUID-7AE96C90-4D7F-41B0-8235-8CA8D962259F.html>. [Kasutatud 13 Mai 2019].
- [10] VMware, „Snapshot,“ [Võrgumaterjal]. Available: <https://kb.vmware.com/s/article/1025279>. [Kasutatud 20 Mai 2019].

## Lisa 1 – vRealize Automation Hardware Specifications and Capacity Maximums

Server Role	Components	Required Hardware Specifications	Recommended Hardware Specifications
vRealize Automation Appliance	vRealize Automation Services, vRealize Orchestrator, vRealize Automation Appliance Database	CPU: 4 vCPU	Same as required hardware specifications.
		RAM: 18 GB (See Scalability Considerations for more information.)	
		Disk: 140 GB	
		Network: 1 GB/s	
Infrastructure Core Server	Web site, Manager Service, DEM Orchestrator, DEM Worker, Proxy Agent	CPU: 4 vCPU	Same as required hardware specifications.
		RAM: 8 GB	
		Disk: 40 GB	
		Network: 1 GB/s	
Infrastructure Web Server	Web site	CPU: 2 vCPU	CPU: 2 vCPU
		RAM: 8 GB	RAM: 8 GB
		Disk: 40 GB	Disk: 40 GB
		Network: 1 GB/s	Network: 1 GB/s
Infrastructure Manager Server	Manager Service, DEM Orchestrator	CPU: 2 vCPU	CPU: 2 vCPU*
		RAM: 8 GB	RAM: 8 GB
		Disk: 40 GB	Disk: 40 GB
		Network: 1 GB/s	Network: 1 GB/s
			* 4 vCPU if performing more than 100 concurrent provisions.
Infrastructure Web/Manager Server	Infrastructure Web/Manager Server	CPU: 2 vCPU	CPU: 2 vCPU
		RAM: 8 GB	RAM: 8 GB
		Disk: 40 GB	Disk: 40 GB
		Network: 1 GB/s	Network: 1 GB/s
Infrastructure DEM Server	(One or more) DEM Workers	CPU: 2 vCPU	CPU: 2 vCPU*
		RAM: 8 GB	RAM: 8 GB
		Disk: 40 GB	Disk: 40 GB
		Network: 1 GB/s Per DEM Worker	Network: 1 GB/s Per DEM Worker

			* 4vCPU if performing more than 100 concurrent provisions.
Infrastructure Agent Server	(One or more) Proxy Agent	CPU: 2 vCPU	CPU: 2 vCPU
		RAM: 8 GB	RAM: 8 GB
		Disk: 40 GB	Disk: 40 GB
		Network: 1 GB/s	Network: 1 GB/s
MSSQL Database Server	Infrastructure Database	CPU: 2 vCPU	CPU: 8 vCPU
		RAM: 8 GB	RAM: 16 GB
		Disk: 40 GB	Disk: 80 GB
		Network: 1 GB/s	Network: 1 GB/s
vRealize Business for CloudAppliance	vRealize Business for Cloud Appliance services vRealize Business for CloudDatabase Server	CPU: 2 vCPU	Same as required hardware specifications.
		RAM: 4 GB	
		Disk: 50 GB	
		Network: 1 GB/s	

Tabel 1. vRealize Automation Hardware Specifications [10]

## Lisa 2 – vRealize Automation Resource Capacity Maximums

Parameter		Maximum Value
Tenant		100
vSphere Endpoints		45*
		* Maximum of 25 vCenter agents per server.
Compute Resources		200
Managed Machines		75
Peak Concurrent Request		
	Constant	100
	Bursts	250

Business Groups		5,000 With 10 unique users per business group and with no user being a member of more than 50 business groups.
Reservations		14,000 With 3 reservations per business group.
Blueprints		
	CBP only	6
	CBP + XaaS	8
Catalog Items		
	Across Tenants	4
	In a Single Tenant	6
User/Group Sync with Default 18 GB Memory		
	Number of Users	95
	Number of Groups	20,000 Each group contains 4 users including one level of nesting.
User/Group with Memory Increased to 30 GB		
	Number of Users	100
	Number of Groups	750 Each group contains 4,000 users and each user is in 30 groups.

Tabel 2. vRealize Automation Resource Capacity Maximums [10]