

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Elina Pavlova 185367IAAB

TEHIK konfiguratsioonihalduse täiendamine

Bakalaureusetöö

Juhendaja: Nadežda Furs
MBA

Tallinn 2021

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: Elina Pavlova

17.05.2021

Annotatsioon

Selle lõputöö põhieesmärk on luua optimaalne lahendus andmebaasi ja veebiserverite ja nendega seotud teenuste oleku jälgimiseks.

Selle lahenduse lõi lõputöö autor kasutamiseks TEHIKus. Lahendus on üles ehitatud ITILv2 baasil, kuna see oli kõige sobivam käsitletava ülesande lahendamiseks.

Lahenduse väljatöötamise jagas autor kahte ossa: probleemi lahendamiseks teoreetiline analüüs ja lahenduse praktiline rakendamine.

Selle töö tulemuseks on probleemi lahendamiseks loodud tarkvara mustrite ehk koodi kasutamine, mis aitab jälgida andmebaasi ja veebiserverite olekut, parandab kontrolli nende üle ning aitab ära hoida võimalikke vahejuhtumeid ja probleeme.

Kokkuvõtteks on oluline märkida, et nimetatud probleem on lahendatud, lahendust saab rakendada ka teistes ettevõtetes, samuti on võimalik seda edasi arendada ja teenuste seisundi jälgimiseks välja töötada automatiseerimissüsteem.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 28 leheküljel, 4 peatükki, 3 joonist, 6 tabelit.

Abstract

Complement of TEHIK Configuration Management

The main goal of this thesis is to create an optimal solution for monitoring the status of database and web servers and related services.

This solution was created by the author of the dissertation for use in TEHIK. The solution is built on the basis of ITILv2, since it was the most suitable for solving the task at hand.

The author divided the development of the solution into two parts: theoretical analysis of suitable software for solving the problem and practical implementation of the solution.

The result of this work is the use of software that was created to solve the problem, which helps to monitor the status of database and web servers, improves control over them and helps prevent potential incidents and problems.

In conclusion, it is important to note that the stated problem has been solved, the solution can also be implemented in other enterprises, it will also be possible to further develop this software and develop an automation system to track the status of services.

The thesis is in estonian and contains 28 pages of text, 4 chapters, 3 figures, 6 tables.

Lühendite ja mõistete sõnastik

AB	<i>Andmebaas, Database</i>
CI	<i>Configuration Item, konfiguratsioonielement</i>
CMDB	<i>Configuration Management Database, konfiguratsioonihalduse andmebaas, kuhu on koondatud konfiguratsioonielementide andmed</i>
IKT	<i>Info- ja kommunikatsioonitehnoloogia</i>
ITSM	<i>IT service management, IT-teenuste haldamise</i>
SNMP	<i>Simple Network Management Protocol, Lihtne võrguhalduse protokoll</i>
SSH	<i>Secure Shell, Turvakest</i>
SSL	<i>Secure Sockets Layer, Turvasoklite kiht</i>
WMI	<i>Windows Management Instrumentation</i>

Sisukord

1 Sissejuhatus	9
2 Probleemi kirjeldus ja eesmärk.....	11
2.1 Organisatsiooni tutvustus	11
2.2 Olemasolev olukord.....	12
2.3 Probleem ja eesmärk.....	14
2.4 Autori roll ja ülesande püstitus.....	15
3 Metoodika valik.....	16
3.1 Konfiguratsioonihalduse olemus ITIL v2 järgi	16
3.2 Alternatiivid (Miks Jira Insight?)	19
4 Lahenduse väljatöötamine	23
4.1 Nõuded täiendamisele.....	23
4.1.1 Veebirakenduste instantsid	23
4.1.2 Andmebaasi instantsid.....	24
4.2 Andmete algallikate analüüs.....	25
4.2.1 Veebirakenduste algallikad	26
4.2.2 Andmebaaside algallikad.....	27
4.3 Insight Discovery mustrite loomine ja testimine	31
4.4 Konfiguratsioonielementide ja nende seoste kaardistamine.....	33
4.5 Andmete valideerimine.....	34
4.6 Järeldused	34
5 Edaspidised sammud	36
6 Kokkuvõte	37
Kasutatud kirjandus	38
Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks	40
Lisa 2 – Nginx veebirakenduse serveri muster.....	41
Lisa 3 – Nginx veebirakenduse serveri muster.....	45
Lisa 4 – PostgreSQL AB instantsi muster	46
Lisa 5 – MySQL/MariaDB AB instantsite muster	47

Jooniste loetelu

Joonis 1. TEHIKu struktuur [4].	12
Joonis 2. Kõik haldusprotsessid suhtlevad konfiguratsioonihaldusega [7].	19
Joonis 3. Testitulemused testijate kaupa.	22

Tabelite loetelu

Tabel 1. Projekti MoSCoW kriteeriumid tarkvarade võrdluseks.	20
Tabel 2. Veebirakenduste konfiguratsioonielementide atribuudid.....	23
Tabel 3. Andmebaasi instantsite atribuudid.	24
Tabel 4. Andmebaaside atribuudid.....	25
Tabel 5. Veebirakenduste serverite konfiguratsioonifailide parameetrite loetelu.....	26
Tabel 6. Andmebaasi instantsite konfiguratsioonikirjete algallikate analüüs.	27

1 Sissejuhatus

Konfiguratsioonihaldus on ITIL-i määratletud IT-teenuste haldamise (ITSM) vorm, mis tagab, et IT-süsteemi ressursid, näiteks virtuaalmasinad, tarkvara, protsessorid, mälu, on teada ja usaldusväärne. Mõnikord nimetatakse seda IT-automatiseerimiseks, kuna on üks valdkondadest, kus IT leiab edu aega nõudvate ülesannete automatiseerimisel, eriti konfiguratsioonielementide täpse inventeerimise jaoks konfiguratsioonihalduse andmebaasi (CMDB) värskendamisel.[1] Automatiseerimine on väga oluline IT-meeskondadele, kes üritavad end kursis hoida paljude manuaalülesannetega, mis on tüütud või üha keerulisemad. Kuna TEHIKul on keeruline IT-infrastruktuur, mis aina laieneb, siis tekkis vajadus konfiguratsioonihalduse juurutamisele.

Lõputöö autor tegeleb praktilise uuringu, analüüsi ja potentsiaalse perspektiivse lahenduse loomisega organisatsiooni TEHIKu jaoks. Konfiguratsioonihalduse projekti olemus, mille kallal autor töötab, on teabe kuvamine serverite ning nendel paigaldatud ja töötavate IKT teenuste kohta, sh nende olekute ja vastutajate kohta. Antud lõputöö käsitleb probleemi seotud puuduvate konfiguratsioonielementide kategooriate kaardistamisega, et tagada IKT teenuste läbipaistvust teiste haldusprotsessidele.

Väljatöötatud lahendus aitab tagada kvaliteetsema IKT teenuste osutamiseks läbipaistvust konfiguratsioonihalduse näol ning viia lõppuni organisatsiooni projekti, mis edaspidi aitab organisatsiooni igapäevased tööd kiirendada ja optimeerida süsteemi üldist stabiilsust ning saada koheselt teavet süsteemi ja selle teenuste võimalike probleemide kohta.[2] Samuti aitab lahendus suuresti kaasa intsidendi põhjuse ja juur probleemi algallika tuvastamisele. Selles töös püstitatud probleem on väga oluline, kuna TEHIK töötab otse äriklientidega ja selliseid potentsiaalselt negatiivseid olukordi tuleb hoolikalt jälgida.

Käesolev töö on aktuaalne, sest toode arendajad ei paku universaalset lahendust või lahendus käsitletud probleemile puudub üldsegi. Väljatöötatud töötulemused peaksid olema universaalsed, seega mis tahes organisatsioon, mis soovib juurutada sama

tarkvaralahendust enda konfiguratsioonihalduse protsessi jaoks, saab autori lahendust rakendada.

Töö koosneb neljast peatükist. Esimeses peatükis kirjeldatakse tänast organisatsiooni TEHIK, hetkeolukorda konfiguratsioonihalduse juurutamisest projektis, käsitleva probleemi ning eesmärgi. Samuti käsitletakse peatükis detailsemalt autori rolli ning kirjeldatakse lõputöö skoopt. Teises peatükis esitab lõputöös kasutatavat metoodikat, kirjeldab konfiguratsioonihalduse olemasolu ITIL v2 põhjal ja projekti otsuse protsessi tarkvaralahenduse valiku osas. Kolmandas peatükis esitatakse lahenduse väljatöötamiseks ette võetud sammud, milleks on nõuded konfiguratsioonielementide kaardistamiseks, konfiguratsioonikirjete võimalike algallikate analüüs, koodi loomine ja testimine, konfiguratsioonielementide ning nende seoste kaardistamine mudelisse. Samas peatükis kirjeldatakse saadud andmete valideerimist ning tehtud järeldusi. Viimases ehk viies peatükis tuuakse välja edaspidised plaanitud tööd toodangukeskkonnas.

2 Probleemi kirjeldus ja eesmärk

Antud peatükis antakse ülevaade organisatsioonist ja projekti hetkeolukorrast, selgitatakse põhjalikult probleemi olemust, lõputöö eesmärki ja autori rolli eesmärgi saavutamiseks.

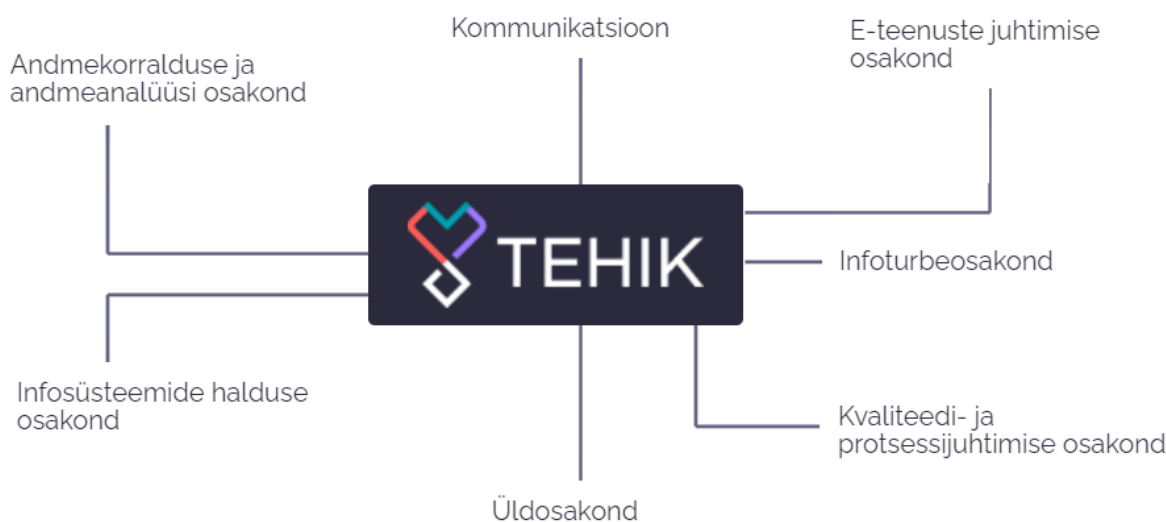
2.1 Organisatsiooni tutvustus

TEHIK ehk Tervise ja Heaolu Infosüsteemide Keskus on info- ja kommunikatsioonitehnoloogia (IKT) kompetentsikeskus, mis pakub ja haldab avaliku sektori IKT-teenuseid töö-, tervise- ja sotsiaalvaldkonnas Sotsiaalministeeriumi valitsusala asutustele: Sotsiaalkindlustusamet, Ravimiamet, Terviseamet, Tervise Arengu Instituut ja Astangu Kutserehabilitatsiooni Keskus, samuti Eesti Haigekassale. TEHIK oli loodud Sotsiaalministeeriumi IKT-osakonna ja E-tervise Sihtasutuse konsolideerimisel 2017.aasta alguses.

"Meie loodud e-teenused säästavad inimeste aega, et nemad saaksid keskenduda olulisele". - TEHIK [3]

TEHIKu peamine ülesanne on tagada Sotsiaalministeeriumi haldusalas infosüsteemide laitmatu töö ning luua turvalisi teenuseid kliendi vajaduste ja eesmärgistatud tellimuse järgi, mis säästavad aega, et oleks võimalik keskenduda rohkem olulisele, st ükski kasutaja ei tunneks, et ta kulutab riigiga suhtlemisel tarbetult aega, ning et tervishoiutöötajad saaksid keskenduda patsientide ravile. Keskus on suundunud teenusepõhisele juhtimisele, ning võtnud kasutusele mõõdikud, mis võimaldavad hinnata ja seeläbi planeerida tegevusi, mis tagavad TEHIKu hallatavate IKT-teenuste kvaliteedi, sh infoturbe. TEHIK loob teenuseid toetudes ITIL teenusehalduse võimalustele. ITILi järgi tegutsemine tagab nii pakutavate IT-teenuste kvaliteedi kui ka võimaldab neid teenuseid hallata. Avaliku sektori oluliste IT-projektide ellu viimine, teiste riigiasutuste nõustamine uute arenduste ja halduse osas ning uute tehnoloogiliste lahenduste loomine on samuti suur roll keskuse töös. [3]

TEHIKus töötab ligi 150 eksperti erinevates valdkondades: e-teenuste ja infosüsteemide haldamise ning arendamise spetsialistid, andmekorralduse ja andmeanalüütika eksperdid, küberturvalisuse eestvedajad, kvaliteedijuhtijad, juristid ja finantsspetsialistid. Igaühel on oma roll teenuste terviklikkuse tagamiseks. [4]



Joonis 1. TEHIKu struktuur [4].

TEHIKu kui organisatsiooni struktuur on väga hästi ja loogiliselt välja töötatud, mis võimaldab kõigil osakondadel vajadusel kiiresti ja tõhusalt koostööd teha.

2.2 Olemasolev olukord

Organisatsioonis TEHIK registreeritakse intsidentide põhjused, samuti ebaõnnestunud muudatuste põhjused. Oli tuvastatud kriitiliste teenuste katkestused seotud muudatustega, mida teostati ebatäpse konfiguratsiooni andmete põhjal. Klient kannab reaalset rahalist kahju katkestuse tagajärjel, kuna väljamakse hilineb. Samuti osad intsidendid on seotud konfiguratsiooni informatsiooni puudumisega. Klient kannab mainekahju kui funktsionaalsus ei tööta ja inimene jääb teenindamata. Lisaks konfiguratsioonihalduse andmete puudumise tõttu on infrastruktuur planeeritud liiga ehk kulukalt, on konfiguratsioonielemente omanikuta. Optimaalne teenuse hind kliendile võib seetõttu kasvada. Välja toodud tagajärjete vältimiseks tekkis vajadus alustada konfiguratsioonihalduse protsessi juurutamisega.

Konfiguratsioonihalduse projekti eesmärk on kaardistada ja hallata kõiki IT teenuse osutamiseks vajalikke konfiguratsioonielemente (CI) ja seoseid tagamaks usaldusväärne

alus teistele IT halduse protsessidele (sh intsidentide, probleemide, muudatuste-, reliisihaldus). Konfiguratsioonihaldus peab võimaldama:

- tagada teenuste omanikele ja juhtkonnale kompaktne ja ajakohane info IKT teenuste toimimise kohta;
- IKT teenuste kvaliteetse ja võimalikult veavaba osutamise kasutajatele;
- inimlike eksimuste vähendamist IT teenuste osutamisel;
- muudatuste halduse, intsidentide ja teenussoovide halduse ning probleemihalduse protsesside toe konfiguratsioonihalduse kaardistuste näol;
- paremini planeerida infrastruktuuri muudatusi;
- vähendada personali liikumisest tulenevaid teabe kadumise riski.

Projekti käigus väljakujunenud ning projekti vajadustele vastav haldusprotsess pole organisatsioonis lõpuni välja töötatud ja juurutatud. Organisatsiooni projektis oli tehtud analüüs ja valikuetpanekud konfiguratsioonihalduse toodete osas ning otsustatud hangitav toode, milleks on Mindville Jira Insight ja temale kuuluv Insight Discovery. Kooskõlastades hanke põhisammud projekti omanikuga oli hangitud TEHIKu konfiguratsioonihalduseks otsustatud toode. Hangitud tarkvaralahendus oli juurutatud testkeskkonnas ja integreeritud Jira testkeskkonnaga. Oli koostatud konfiguratsioonielementide (CI-de) potentsiaalne mudel, sh CI-de skoop ja sügavus, mida jooksvalt täiendatakse. Hetkel konfiguratsioonihalduse protsessi juurutamine organisatsioonis on ootel seni Discovery-le mitte ligipääsetavate konfiguratsioonielementide taga.

Konfiguratsioonihalduse CI-de ning nende atribuutide kaardistus on kõige raskem osa protsessi juurutamise osas. Peale Insight Discovery automaatiseeritud töö käivitamist jälgiti tema võimalusi ja CMDB-sse liikuvaid CI-d. Tuvastati probleemsed konfiguratsioonielementide kategooriad, millele Discovery hetkel ligi ei pääse.

2.3 Probleem ja eesmärk

Analüüsisides hangitud toode võimekust ja keskse CMDB-sse liikuvaid andmeid tuvastati, et praegune konfiguratsioonihalduse protsess vajab täiendusi. On leitud konfiguratsioonielementide kategooriad, millele hangitud toode, ehk Insight Discovery, hetkel ligi ei pääse, sh rakendusserverid, AB instantsid. Nende kaardistamiseks on vaja mitu inimkuud käsitööd. Rakendusserverite ja AB instantside konfiguratsioonielementide puudumine CMDB-s takistab projekti eemärkide täitmist ning läbipaistvust ja tugeva haldusvõimekust teistele IT halduse protsessidele.

Mindville Insight Discovery skaneerib kõik võrguga ühendatud masinad, mis suudavad suhelda SSH, WMI või SNMP protokollide kaudu. Kogutud andmed on automaatselt imporditud CMDB-sse ja on saadaval Jira rakenduses. Hosti või seadme üksikasjaliku informatsiooni ehk konfiguratsioonikirjete kogumiseks vajab Discovery juhiseid, kuidas seda teha.[5] Need juhised on täpsustatud mustritena, mis on tegelikult C# keeles kirjutatud kood. Koodile antakse üks või mitu terminalis kasutatavaid käsku, millega infot kogutakse, ja seejärel töödeldakse saadud tulemuste konfiguratsioonikirjeteks. Insight Discovery on arendajate poolt pakendatud kasutamiseks valmis mustrite komplektiga, kuid mõned nendest ei sobi iga organisatsiooni struktuurile või vajalikud mustrid üldsegi puuduvad.

Lõputöö eesmärk on täiendada konfiguratsioonihalduse protsessi toimimist, mis tõstaks TEHIKu pakutavate IT teenuste kvaliteedi ja veavaba osutamist teenuste kasutajatele. Täienduse tulemiks on keskne konfiguratsioonihalduse andmebaas, milles uued veebirakenduste ja andmebaasi instantside konfiguratsioonielemendid ja konfiguratsioonielementide seosed kajastavad reaalselt olukorda IT teenustest lõputöö käigus välja kujunenud konfiguratsioonihalduse mudelis, ehk Jira Insight objektide skeemis. Discovery teostab igapäevast andmete uuendust uute loodud mustritega, kandes need automaatselt CMDB-sse. Täiendus tagab kiire ülevaate muudatuste mõjust ja vähendab ebaõnnestumiste juhtumeid ehk intsidente.

2.4 Autori roll ja ülesande püstitus

Lõputöö autor organisatsioonis TEHIK täidab püstitatud probleemi lahenduse välja töötamise käigus süsteemiadministraatori ja konfiguratsioonihalduri rolli. Koos teise konfiguratsioonihalduriga analüüsis autor CMDB-sse liikuvaid konfiguratsioonielemente ning nende vastavaid mustreid. Analüüside tulemusel otsustas autor ette võtta puuduvate veebirakenduste ja andmebaaside instantsite kaardistamist. Skoopi on võetud veebirakenduste instantsid, mis töötavad Linux operatsioonisüsteemidel, sh Apache, Nginx, Tomcat, ning andmebaaside instantsid, mis samuti töötavad Linux operatsioonisüsteemil, sh PostgreSQL, MySQL, MariaDB. Skoopi ei kuulu andmebaasi klientide kaardistus, ehk hostide seosed, mis vastava andmebaasiga ühendust loob ja selle andmeid kasutab.

Käesolevas lõputöös viib autor läbi konfiguratsioonikirjete algallikate analüüsi ning vastava eksperimendi Linux terminali käsudega testkeskkonnas, millega Insight Discovery edaspidi konfiguratsioonikirjetele ligi pääseb. Seejärel loob uue mustri veebirakenduste ja andmebaaside instantsidele. Kaardistatud konfiguratsioonielementide ja konfiguratsioonikirjete jaoks täiendab konfiguratsioonihalduse mudeli Jiras ning loob seoseid. Lõpus koos teenuste vastutajatega valideerib saadud andmeid.

Autor on huvitatud konfiguratsioonihalduse täiendamisest, et aidata teenuste konfiguratsioonipildi terviklikkusele ja võimaldada haldusprotsessides määrata täpsemalt seotud konfiguratsioonielemente, kus muudatusi tehakse või intsidente tekib. Samuti jagatud ressursside osas teha parema ülevaate osakaaludest IT teenuse lõikes ning võimaldada kuvada mõju ja seotust teiste IT teenustega. Lisaks võimaldada selgemat vastutuse jagamist ja haldamist, mis on kasuks igapäevatöös nii vastutajatele endile, kui ka teistele.

3 Metoodika valik

Antud lõputöö on praktiline uurimus, mille läbiviimiseks kasutatakse kvalitatiivset lähenemisviisi. Parema tulemuse ja universaalsema lahenduse saavutamiseks uuris autor mitmeid allikaid ning katsetas mitmeid meetodeid. Välja töötatud lahendus võiks sobida ka samalaadse probleemi lahendatavatele organisatsioonidele. Võrdlev analüüs antud probleemi puhul ei ole teostatud, kuna varem pole samalaadse probleemi käsitletud organisatsiooni sees ega väljaspool.

Lõputöö koosneb kahest osast: analüütiline ja praktiline. Analüütilises osas kirjeldab autor konfiguratsioonihalduriga koostatud nõudeid, konfiguratsioonielementidele vastavate konfiguratsioonikirjete algallikaid ning valideerib saadud andmeid. Praktilises osas viib autor läbi eksperimendi Linux terminali käsudega, täiendab konfiguratsioonikirjete algallikate analüüsi põhjal valitud terminali käsud, kirjutab nõuetele vastavaid mustreid igale tarkvarale C# programmeerimiskeelega ning kaardistab uued konfiguratsioonielemendid ja nende seoseid konfiguratsioonihalduse mudelisse.

3.1 Konfiguratsioonihalduse olemus ITIL v2 järgi

ITIL standardite juurutamine on tänapäeval üheks olulisemaks teemaks IT-s. Selleks, et hästi mõista konfiguratsioonihalduse väärtust, on vaja selgelt mõista, mis on ITIL ja mis see pole. Põhimõtteliselt ITIL tähendabki täpselt seda, mida täisnimetus Information Technology Infrastructure Library tähistab – raamatute kogu. Kõik raamatud annavad juhiseid, mis aitavad organisatsioonidel juurutada parimaid IT-tavasid, mis olid õpitud läbi rasket teed. Seal on köidet turvalisuse kohta, üks planeerimise, üks tarkvara varade ja üks rakenduste haldamise kohta [6]. Raamatute kogu kasvab ja areneb jätkuvalt, kuna edukamad tehnikad ja juhised on dokumenteeritud ja kehtestatud selle kohta. Hetkel ITIL-il on 4 versiooni, kuid projekti juhi otsusest kasutatakse projektis konfiguratsioonihalduse ITIL versiooni 2.

ITIL versiooni 2 kahe köide kombinatsioon “Service Support” ja “Service Delivery” kannab nimetust “IT Service Management”. Koos need kaks köidet moodustavad ITIL tuuma – osa, millest ettevõtte alustavad oma juurutamist. Viis protsessi on kirjeldatud teenuse toes: intsidendi-, probleemi-, muudatuse-, reliisi-, konfiguratsioonihaldus (Incident Management, Problem Management, Change Management, Release

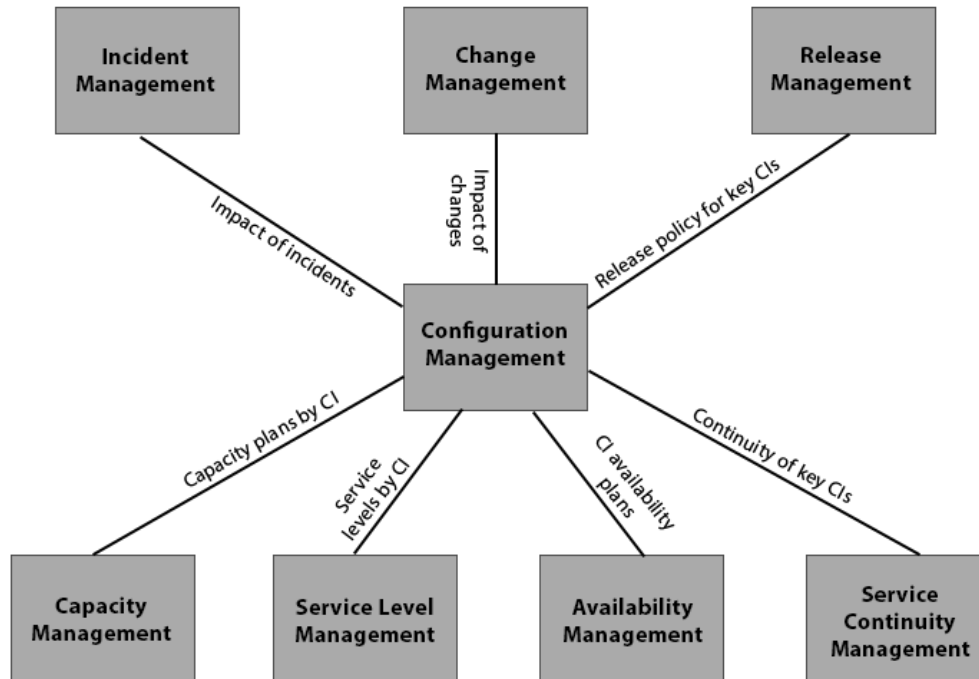
Management, Configuration Management). Teenuse osutamisel on määratletud viis täiendavat protsessi: võimsuse, kättesaadavuse, teenuse järjepidevuse, teenuse taseme, finantshaldus (Capacity Management, Availability Management, Service Continuity Management, Service Level Management, Financial Management). Iga protsess teenuse toes ja teenuse osutamise köietes on seotud konfiguratsioonihaldusega. Selleks, et paremini mõista kuidas, on vaja mõista iga protsessiala kõrgetasemelist eesmärki. Teenuse osutamise protsessid on need, mis mõjutavad otseselt serverite ja tarkvara käitamist. [7]

- Võimsuse haldus (Capacity management) - Eesmärk on mõista, kui palju mahtu (mõõdetuna mälus, ribalaiuses, salvestusruumis ja mitmesugustes muudes seadmetikes) on praegu saadaval, kui palju peab olema tulevikus saadaval ja kuidas oleks kõige parem hallata, et võimsuse puudujääke ei tekiks ja liigset mittevajaliku võimekust ei omandata.
- Kättesaadavuse haldus (Availability management) - Tegeleb IT-infrastruktuuri kättesaadavuse maksimeerimisega trendianalüüsi ja üksikute rikke punktide ennetava kõrvaldamise kaudu.
- Teenuse järjepidevuse haldus (Service continuity management) – Ennetavalt planeerib komponentide katkestusi ja hõlmab kõike, mis on vajalik, et ettevõtte saaks vajalikke IT-teenuseid, hoolimata ühe või mitme IT-komponendi puudumisest.
- Teenuse taseme haldus (Service level management) - määratleb ja jälgib väärtust, mida IT ettevõttele pakub teenuse tasemeks nimetatavate eesmärkide seadmise ja aruandluse kaudu.
- Finantshaldus (Financial management) - tegeleb IT-kulude mõistmisega ja IT-infrastruktuuri asjakohase rahastamisega kõige tõhusamal viisil. [7]

Teenuse tugiprotsessid on keskendunud arvestuse pidamisele, jälgimisele ning protsesside täiustamisele, mis muudavad IT-d edukaks.

- Intsidendihaldus (Incident management) – Jäädvustab teenuse katkestusega seotud üksikasju, edastades need vastavatele osapooltele, et teenus saaks võimalikult kiiresti taastatud.
- Probleemihaldus (Problem management) - Teabeanalüüsifunktsioon, mis määrab kindlaks suure mõju avaldava või sageli korduva katkestuse algpõhjuse ja jälgib seejärel parandusmeetmeid selle algpõhjuse parandamiseks.
- Muudatusehaldus (Change management) - Kriitiline põhimõte, mis kontrollib ja edastab IT-keskkonnas toimuvaid muutusi.
- Reliisihaldus (Release management) - Lähenemisviis muudatuste korraldamisel, et kaasatud ressursse paremini ära kasutada.
- Konfiguratsioonihaldus (Configuration management) - Teadus üldise IT-keskkonna komponentide tuvastamisest ning täpse seisundi jälgimisest ja kontrollimisest iga ajahetkel. [7]

Konfiguratsiooni haldamine on mõeldud vaimse jälgimise koormuse ja sellega kaasnevate vigade eemaldamiseks, mis on aega ja lisapingutuse nõudev. Kõik, mis moodustab IT-keskkonda, tuleks salvestada konfiguratsioonihalduse osana. Kõige tähtsam tuleb jälgida füüsilisi suhteid ja loogilisi sõltuvusi komponentide, alakomponentide, rakenduste, serverite, võrkude ja lugematu osa IT-keskkonna vahel. Need CMDB täpsed ja õigeaegsed andmed tagavad teiste protsesside tõhusa töö ning muid taolisi eeliseid. Neid võtmesõltuvusi hõlbustab konfiguratsiooniüksuste (CI) seostamine muude protsesside põhimõtetega, nagu on kuvatud joonisel 2.



Joonis 2. Kõik haldusprotsessid suhtlevad konfiguratsioonihaldusega [7].

Näiteks, intsidendihalduse meeskond saab probleemi allika kiirelt isoleerida, kuna nad saavad vaadata mõnda serverit ja/või rakendust, mis on maas, ning näha, et need kõik on seotud sama ruuteriga. Muudatusehalduse puhul saab kõrvaldada konflikti, mis tekib heaks kiidetud muudatuste esmapilgul kahe mitte sõltuvate rakendustel, mis tegelikult CMDB andmete järgi töötavad samal andmebaasil, ning esimese muudatuse puhul peaks olema loodud andmebaasist varukoopia, kuid teise puhul peaks andmebaas olema maas.

3.2 Alternatiivid (Miks Jira Insight?)

Projekti alguses konfiguratsioonihalduse meeskonna ees, millesse autor veel ei kuulunud, seisis ülesanne leida konfiguratsioonihalduse toodet, mis sobiks TEHIKu teenuste halduse ökosüsteemiga ja tooks maksimaalse kasu. Algne konfiguratsioonihalduse meeskond viis läbi analüüsi, mille käigus toodi välja järgnevaid potentsiaalseid tarkvara lahendusi: CSM, ServiceNow, Jira Service Desk, Service Desk Plus, SolarWinds Service Desk, Topdesk, Freshservice, SummitAI, SysAid ITSM, Zendesk, SpiceWorks, System Center Service Manager, Alemba Service Manager, InvGate Service Desk, Teamdynamix, Support Works, Atrium CMDB, BMC Helix CMDB, Fossil scm, vScope, I-doit, ChangeGear, Virima, GLPI, Canfigure, Alloy Navigator, DeployHub, NetBrain, RackTables, Aperature, Device42, Mindville Jira Insight. Seejärel iga meeskonna liige

võrdles tarkvarad eelnevalt välja töötatud konfiguratsioonihalduse meeskonna liikmete poolt MoSCoW kriteeriumide alusel, mida on näha tabelis 1. Võrdleja tegi omaette analüüsi allikatest või jagas enda kogemusi.

- M - Kohustuslik kriteerium
- S - Oluline kriteerium. Äriliselt vajalikud nõuded, mis ei ole kriitilise tähtsusega
- C - Võiks olla. Nõuded, mille äriiline kriitilisus on väike
- W - Nõuded, millel ärilist vajadust ei ole [8]

Tabel 1. Projekti MoSCoW kriteeriumid tarkvarade võrdluseks.

Criteria	M (Must have)	S (Should have)	C (Could have)	W (Won't, but nice to have)
<u>Architecture</u>				
On-prem deployment	X			
<u>Functional</u>				
CI-autodiscovery	X			
CI dependancy mapping	X			
CI change autodiscovery and report	X			
Must support high-level service abstraction		X		
<u>Deployment and support</u>				
Must have local support (trusted)	X			
Must provide private demo/onboarding		X		
<u>Traction and reference</u>				

Must have at least 1000 customers		X		
Must be 2+ years on the market	X			
<u>Usability</u>				
Support for diverse consumers of CMDB data		X		
<u>Future</u>				
DevOps and continuous deployment support		X		

Konsultatsiooni käigus oli leitud kolm funktsionaalsuselt sobivat kandidaati TEHIKu konfiguratsioonihalduse tarkvaraks: Mindville Jira Insight, ServiceNow, Freshservice. Mindville Jira Insight-iga oli alustatud lahenduse sobivuse hindamine, ehk test. Teiste toodetega ei õnnestunud testi läbi viia: ServiceNow on-prem rakenduse miinimumtasu osutus liiga kalliks, Freshservice tugi ei avaldanud informatsiooni, millega selguks vastavus vajalikele turbe juhiste.

Mindville Jira Insight testi käigus installeeriti tarkvara TEHIKu Jira keskkonda ja mängiti läbi ühe teenuse CMDB loomist ja haldamist. Seejärel testimismeeskond, kuhu kuulusid süsteemiadministraatorid, infoturbe spetsialist, IT halduse projektijuhid ja konsultant, hindas viite kriteeriumi: strateegiline toote sobivus, funktsionaalsus, kasutatavus, juurutuse lihtsus ja hind. Hindamisel anti punkte 1-10 skaalas arvestades, et kui testija ei osanud anda hinnangut siis oli hinnanguks 0, mida hiljem kaaluti konkreetse hinnangu osatähtsusega. Testitulemusi testijate kaupa saab näha joonisel 3.

Grupp	Kaal	Maks punkte	Konsultant	IT halduse projektijuht 1	Infoturbe spetsialist	Süs. admin 1	Süs. admin 2	IT halduse projektijuht 2	Kesk m.	Standardhälve
Strateegiline Funktsionaalsus	20%	10	5,26	5,13	6,25	7,66	6,15	8,02	6,41	1,10
Kasutatavus	20%	11,3	9,28	8,79	7,1	9,15	5,5	10,9	8,45	1,72
Juurutuse lihtsus	20%	10	9,2	5,74	2,7	7	4,8	5,81	5,88	1,98
Hind	20%	10	4,4	3,6	3,2	5	5	8	4,87	1,55
					ei hinnatud		ei hinnatud			
			10	10		5		9,25	8,56	2,08
Kokku % maksimumist	100%	51,3	38,14	33,26	19,25	33,81	21,45	41,98	31,32	
			74%	65%	38%	66%	42%		61%	

Joonis 3. Testitulemused testijate kaupa.

Kombineeritud tulemuse põhjal selgus, et TEHIKu CMDB vahendiks võib valida Mindville Jira Insight-i. Toode sobib TEHIKu teenuste halduse ökosüsteemi, kuna integreerub täieikult Atlassian Jira tootega ning kasutab tööks Jira infrastruktuuri, s.h. andmebaasi. Samuti toode on saanud demo-kasutusest valdavalt positiivset tagasisidet. Testijad olid valdavalt „ühel meelel“ mida näitab hinnangute vähene hälbimine keskmisest. Kõige madalama skoori sai hinnatud „juurutuse lihtsus“, mille hinnangud hälbisid ka kõige vähem. Üldine arusaam oli, et juurutus saab olema töömahukas ja juurutuse ettevalmistus seega on kogu ettevõtmise õnnestumise seisukohast kriitilise tähtsusega. Kõige enam hälbis kasutatavuse hinnang, sest ei ole tegemist ülimalt intuitiivse tootega, kasutamisel on omad eripärad. Osaliselt saab kasutatavuse ja funktsionaalsuse skoori põhjendada ka sellega, et kui testija ei suutnud/osanud hinnata, siis jättis ta lihtsalt hinnangu andmata. Täiendavalt antud kirjalikud hinnangud olid valdavalt positiivsed.

4 Lahenduse väljatöötamine

Järgnevalt kirjeldab autor lahenduse väljatöötamiseks ette võetud sammud.

Lõputöö käigus kasutatakse organisatsiooni Jira Data Center versioonil 8.13.0 testkeskkonda. Insight Discovery tarkvara on beta versioonil 3.0.2 ja Jira Insight add-on versioonil 8.7.3. Pakutud lahenduse saab kasutada Jira Data Center 8.x versioonidega, ning sellele vastavalt Insight Discovery 3.x versioonidega ja Jira Insight add-on versioonidega alates 8.1. Insight Discovery oli juba eelnevalt seadistatud testkeskkonnas neljal Linux masinal ja neljal erinevatel alamvõrkudel. Insight Discovery 3.x versioonid on veel beta versioonid ning pole soovitatav testimiseks toodangukeskkonnas.

4.1 Nõuded täiendamisele

Projekti arutamise käigus koos konfiguratsioonihalduriga, kes vastutas protsessi toimimise ja juurutamise eest, tekkis vajadus koostada konfiguratsioonihalduse täienduse üldnõuded. Seoses sellega autor koostöös konfiguratsioonihalduriga töötas välja veebirakenduste ja andmebaaside instantsite atribuute, millele võiks Insight Discovery-le ligipääsu luua ja kaardistada konfiguratsioonihalduse mudelisse. Allpool loetletud tabelid representeerivad atribuutide loetelu, millele autor kaardistust mustritega loob.

4.1.1 Veebirakenduste instantsid

Tabelis 2 kirjeldab autor koostatud üldnõuded veebirakenduste konfiguratsioonielementide kohta, mis olid võetud töö skoopi. Tabelis on kaks tulpa: esimeses on atribuutide loetelu, mida autor konfiguratsioonihalduse mudelisse kaardistab, teises on konfiguratsioonikirje kirjeldus, ehk mida ta ennast välja kujutab.

Tabel 2. Veebirakenduste konfiguratsioonielementide atribuudid.

Atribuut	Kirjeldus
Name	Peamine atribuut, mille alusel CMDB mudelis objekti tüübi nimekirjas konfiguratsioonielemendi eristatakse. Väärtuseks saab virtuaalhosti nimi või Ethernet võrguliidese IPv4.
Status	Vastava tarkvara instantsi staatus. Väärtuseks saab "Running" või "Stopped".

Root	Virtuaalse hosti juurkataloog, mida kasutatakse veebirakenduste failide otsimiseks.
Port	Virtuaalse hosti pordi number, mida kasutatakse sissetulevate ühendusteks. Näiteks võib väärtuseks olla 80, 443, 8080.
SSL	Võimalusel kuvab, kas vastav virtuaalne host kasutab SSL protokoll. Väärtuseks on kas „true“ või „false“.
ObjectIdentifier	Insight Discovery poolt genereeritud instantsi konfiguratsioonielemendi hash väärtus.
Referenced Applications	Veebiserveritega seotud tarkvarade loetelu.
Referenced Host	Host, mille peal veebiserver töötab.

Tabelis loetletud atribuudid on veebiserverite konfiguratsioonielementide kategooriale aluseks. Atribuutide nimetused võivad olla omapärased.

4.1.2 Andmebaasi instantsid

Tabelis 3 kirjeldab autor koostatud üldnõuded andmebaaside instantsite ehk konfiguratsioonielementide kohta, mis olid võetud töö skoopi. Tabelis on kaks tulpa: esimeses on atribuutide loetelu, mida autor konfiguratsioonihalduse mudelisse kaardistab, teises on konfiguratsioonikirje kirjeldus, ehk mida ta ennast välja kujutab.

Tabel 3. Andmebaasi instantsite atribuudid.

Atribuut	Kirjeldus
Name	Peamine atribuut, mille alusel konfiguratsiooni mudelis objekti tüübi nimekirjas instantsi eristatakse. Väärtuseks saab FQDN või selle puudumisel hosti nimi.
Status	Vastava tarkvara instantsi staatus. Väärtuseks saab “Running” või “Stopped”.
Service Uptime	Instantsi jooksutamise aeg, kui atribuudi “Status” väärtus on “Running”.
Port	Instantsi pordi number. Väärtuseks saab vaikinisi „5432“ PostgreSQL jaoks [10], „3306“ MySQL ja MariaDB jaoks [9] või muu määratud number.
Instance IP	Instantsi IP number, millega saab andmebaasiga ühenduse luua. Väärtuseks saab kas vaikinisi localhost [9][10] või Ethernet võrguliidese IPv4.

Version	Instantsi tarkvara versiooni number.
ObjectIdentifier	Insight Discovery poolt genereeritud instantsi konfiguratsioonielemendi hash väärtus.
Host Object Hash	Vastava hosti hash väärtus, mille peal instants töötab. On vajalik hostiga seose loomiseks.

Koos instantsitega oleks vaja kaardistada ka nendel olevaid andmebaase kui andmete kogumeid. Tabelis 4 on välja toodud nende atribuudid koos kirjeldusega.

Tabel 4. Andmebaaside atribuudid.

Atribuut	Kirjeldus
Name	Peamine atribuut, mille alusel konfiguratsiooni mudelis objekti tüübi nimekirjas andmebaasi eristatakse. Väärtuseks saab andmebaasi nimi.
Encoding	Tähemärkide kodeering. Näiteks UTF-8.
Size MB	Andmebaasi andmemahd megabaitides.
Description	Kirjeldus andmebaasile, mis võib ka puududa.
Instance	Vastava andmebaasi instantsi konfiguratsioonielement. Selle tuvastamiseks mudelis kasutatakse instantsi atribuudid „Name“ ja „Version“ väärtusi.
ObjectIdentifier	Insight Discovery poolt genereeritud instantsi hash väärtus vastavalt seadele.

Tabelites loetletud atribuudid on andmebaasi ja nende instantsite konfiguratsioonielementide kategooriale aluseks. Atribuutide nimetused võivad olla omapärased.

4.2 Andmete algallikate analüüs

Konfiguratsioonikirjete algallikate tuvastamiseks kasutati ametlike tarkvara dokumentatsioone, foorumeid ning viidi läbi eksperiment käsu ridadega. Autor üritas tekitada universaalsema ligipääsu algandmetele, et oleks kaetud suurem hulk konfiguratsioonielementidest. Analüüsi tulemused kirjeldatakse tabelitena, kus tuuakse välja ainult need atribuudid, millele kaardistust luuakse terminali käsudega, muud

atribuudid kaardistab Insight Discovery ise automaatselt või luuakse koodiga, kasutades eelnevalt kaardistatud andmeid.

Algallikateks pakub autor välja ka konfiguratsioonifaile, mis on kõige täpsem ja efektiivsem meetod konfiguratsioonikirjete kogumiseks. Kuna ligipääs failile on piiratud ja andmeid kätte saada on võimalik ainult lisa administratiivsete turvaõigustega, ehk lisades *sudo* käsu ette, mis võib osutada infoturbe juhustega vastuollu, siis tuli valiku arutada infoturbe osakonnaga ja kinnitust saada. Peale konsultatsiooni infoturbespetsialistiga, sai autor kinnitust, et antud meetod sobib juhul, kui iga *grep* käsu filtri muster on eraldi täpsustatud Insight Discovery sudoers faili ning neid on mõistlik ja hallatav arv. Antud ligipääsud seadistab Uyuni administraator.

4.2.1 Veebirakenduste algallikad

Analüüsi käigus oli tuvastatud, et veebiserverite konfiguratsioonikirjeid, mida kaardistati tabelis 4.1.1, saab kaardistada konfiguratsioonifailidest. Allpool on välja toodud konfiguratsioonifailides olevad parameetrid, millest plaanitakse andmeid kätte saada.

Tabel 5. Veebirakenduste serverite konfiguratsioonifailide parameetrite loetelu.

Atribuut	Nginx	Apache	Tomcat
Name	Parameeter		
	server_name [13]	ServerName [16]	Host name [17]
Root	Parameeter		
	root [14]	DocumentRoot [16]	docBase [17]
Port	Parameeter		
	listen [15]	Virtualhost {port} [16]	port [18]
SSL	Allikas1		
	ssl [19]	SSLEngine [20]	SSLEnabled [21]
	Märkused		
	On virtuaalse hosti väli konfiguratsioonifailis, millel on määratud „on“ või „off“ väärtused. Tomcat puhul on väärtusteks „true“ või „false“. Samuti		

	tuleb märkida, et Nginx ssl olemasolu virtuaalses hostis võib olla ka määratud „listen“ väljal.
--	---

Tabelis loetletud atribuute kaardistatakse iga tuvastatud virtuaalse hosti kohta eraldi.

4.2.2 Andmebaaside algallikad

Kuna MariaDB oli loodud MySQL relatsioonandmebaasi haldussüsteemi põhjal, ühildub MySQL API-ga ja käsudega, siis edaspidi loetakse neid üheks kategooriaks [4]. PostgreSQL puhul on pakutud ka lahendus kui PostgreSQL jookseb Patroni peal. Patroni on klastrihaldur, mida kasutatakse PostgreSQL kõrge kättesaadavusega (high availability) klastrite juurutamise ja hoolduse kohandamiseks ning automatiseerimiseks [6]. Tabel 5 annab ülevaate iga atribuudi kohta, mis olid kaardistatud 4.1.2 peatükis, autori poolt tehtud analüüsis järgneval kujul: olemasolul kirjutatakse välja terminali käsk, kust on võimalik saada vajalikud andmed või selle puudumisel muu allikas, seejärel kirjutatakse vajadusel märkused, kus analüüsitakse lahenduse sobivust ja puudusi, ning lõpuks tuuakse välja analüüsi tulemusena saadud lõplik allikas ja autori poolt täiendatud käsk, mida edaspidi kasutatakse muustris. Allikates sõltuvad muutujad on trakside vahel.

Tabel 6. Andmebaasi instantsite konfiguratsioonikirjete algallikate analüüs.

Atribuut	PostgreSQL	MySQL / MariaDB
Name	Allikas	
	Saadakse juba kaardistatud “Hostname” või “FQDN” väärtusest.	Saadakse juba kaardistatud “Hostname” või “FQDN” väärtusest.
Status	Allikas	
	Saadakse juba kaardistatud instantsi teenuse (application service), nt “postgresql-11”, staatuse väärtusest.	Saadakse juba kaardistatud instantsi teenuse (application service), nt “mysqld” või „mariadb“, staatuse väärtusest.
Service Uptime	Terminali käsk 1 [11]	
	systemctl status {postgresql unit name}	systemctl status {mysql/mariadb unit name}

	Märkused	
	<p>Kuvab “Active” real kuupäeva, millal instants tööle hakkas, ja aja, kaua see töö oli minutite, tundide jne. kaupa. Tuleb täiendada, kasutades <i>awk</i> või <i>grep</i> käsu, et filtreerida välja ainult vajaliku.</p> <p>Pole universaalne lahendus, sest hostil võib puududa <i>systemd</i> pakett.</p>	
	Terminali käsk 2 [11]	
	<code>service {postgresql unit name} status</code>	<code>service {mysql/mariadb unit name} status</code>
	Märkused	
	<p>Kuvab “Active” real kuupäeva, millal instants tööle hakkas, ja aja, kaua see töö oli minutite, tundide jne. kaupa. Tuleb täiendada, kasutades näiteks <i>awk</i> või <i>grep</i> käsu, et filtreerida välja ainult vajaliku.</p> <p>Sobib lahenduseks, kuid on selliseid olukordi, kus antud käsk tagastab ainult staatust töötava protsessi id numbrit.</p>	
	Terminali käsk 3 [27]	
	<code>ps -o etime= -p {pid number}</code>	<code>ps -o etime= -p {pid number}</code>
	Märkused	
	<p>Annab aja, kaua instants oli töö formaadis “päev-tund:minut:sekund”. Tuleb kombineerida teise käsuga, et saada vastava protsessi id numbrit. Näiteks võiks kombineerida terminali käsuga 2.</p> <p>On universaalne ja sobib lahenduseks.</p>	
	Tulemus	
	<p>Analüüsi tulemusena valiti universaalsema käsu, täiendades seda ülaltoodud „Terminali käsk 2“ allikaga ja <i>grep</i> filtriga:</p> <pre>ps -o etime= -p `service {unit name} status grep -i pid grep -Po '[0-9]+'`</pre>	
Instance IP ja Port	Terminali käsk 1 [12]	
	<code>sudo ss -plunt grep 'postgres\ postmaster'</code>	<code>sudo ss -plunt grep 'mysql'</code>
	Märkused	

<p>Annab ip ja pordi numbrit, mida instants kasutab tema peamise protsessi id numbriga ja kasutajaga (PostgreSQL puhul postgres või postmaster kasutaja, MariaDB ja MySQL puhul kasutaja mysql). Tuleb veel täiendada <i>grep</i> ja <i>awk</i> käsudega, et filtreerida välja ainult vajaliku. On oluline saada instantsi protsessi id numbrit <i>grep</i> filtri jaoks, mis tagaks andmete täpsust, kui hostil töötavad mitu instantsi korraga.</p> <p>Sobiks lahenduseks, kuid puuduseks on see, et andmete kaardistamiseks peab instants tööle olema.</p>	
<p>Terminali käsk 2</p>	
<pre>ps aux grep {pid number}</pre>	-
<p>Märkused</p>	
<p>Sisaldab parameetreid, millega PostgreSQL instantsi jooksetati. Käsu tuleb täiendada käsuga, millega saab kätte vajaliku instantsi protsessi id numbrit.</p> <p>Sobiks lahenduseks, kuid enamasti ajast instantsi IP ja pordi numbreid parameetrid seal puuduvad. Puuduseks on samuti see, et andmete kaardistamiseks peab instants tööle olema. Kasuks aga lähivad parameetrid, mis kuvavad PostgreSQL bin ja data kataloogide asukohta: saadud bin kataloogis asub utiliit, millega on võimalik tuvastada versiooni numbrit, data kataloogis asub konfiguratsioonifail (Allikas 1).</p>	
<p>Allikas 1</p>	
<pre>postgresql.conf [22]</pre>	<pre>my.cnf [23]</pre>
<p>Märkused</p>	
<p>On konfiguratsioonifail, kus on määratud instantsi IP „listen_addresses“ väljal PostgreSQL failis ning „bind-address“ MySQL ja MariaDB väljal. Instantsi port on kõigi tarkvarade puhul „port“ väljal.</p> <p>PostgreSQL konfiguratsioonifaili asukoht võib olla mis iganes kataloogis, seega tuleb leida viisi selle tuvastamiseks, näiteks terminali käsk 2. MySQL ja MariaDB konfiguratsioonifail asub /etc/ või /etc/mysql/ kataloogis. MySQL ja MariaDB atribuudid võivad olla ka muu konfiguratsioonifailis, mis on määratud my.cnf konfiguratsioonifaili „!include“ või „!includedir“ välja kaudu.</p> <p>On universaalne ja täpne meetod, sest ei sõltu instantsi staatusest.</p>	
<p>Allikas 2</p>	
<pre>postgresql.yml [6]</pre>	-

	<p>On Patroni poolt kasutatav konfiguratsioonifail PostgreSQL instantsi jooksutamiseks. IP ja pordi numbrit saab väljal „listen“. Asukoht võib olla mistahes kataloogis, seega tuleb leida viisi selle tuvastamiseks.</p>	
	<p>Tulemus</p>	
	<p>Analüüsi lõpus valiti allikaks konfiguratsioonifaile (Allikas 1 ja 2). Lisaks PostgreSQL jaoks alternatiiviks valiti ka terminali käsu 2.</p> <p>Täiendatud terminali käsk 2 PostgreSQL jaoks:</p> <pre>1. ps aux grep `service {unit name} status grep -i pid grep -Po '[0-9]+'` 2>&1 grep -Po '/usr/.*'</pre> <p>Käsk PostgreSQL konfiguratsioonikirjete saamiseks allikast 1:</p> <pre>1. sudo grep '^#\?port\ listen_addresses' {data directory path}/postgresql.conf</pre> <p>Käsk PostgreSQL konfiguratsioonikirjete saamiseks allikast 2:</p> <pre>1. sudo grep -A5 'postgresql:' {yaml template file path} grep 'listen\ data_dir\ bin_dir'</pre> <p>Käsud MySQL ja MariaDB konfiguratsioonikirjete saamiseks allikast 1:</p> <pre>1. sudo grep '^#\?port\ bind-address\ !include' /etc/my.cnf 2. sudo grep '^#\?port\ bind-address\ !include' /etc/mysql/my.cnf</pre>	
<p>Version</p>	<p>Terminali käsk 1</p>	
	<pre>{utility path}/postgres -V [25]</pre>	<pre>mysql -V [24]</pre>
	<p>Märkused</p>	
	<p>On utiliit, mis kuvab kasutava tarkvara versiooni. PostgreSQL puhul tuleb leida utiliidi „postgres“ asukohta, sest asukoht erineb iga versiooniga ja installeerimise meetodist.</p> <p>Sobib lahenduseks ainult PostgreSQL puhul, sest MySQL/MariaDB käesolev utiliit kuvab kliendi versiooni, mis võib serveri versioonist täiesti erinev olla.</p>	
	<p>Terminali käsk 2</p>	
	<pre>-</pre>	<pre>mysql -u {username} -p{password} -e „select version();“ [24]</pre>
	<p>Märkused</p>	
	<p>Käesolev käsk loob ühendust serveriga ja täidab käsu, mis kuvab serveri versiooni.</p>	

	Sobib lahenduseks, kuid töötab ainult instantsi aktiivse staatusega, ehk kui instants töötab.
	Tulemus
	<p>Analüüsi lõppus valiti PostgreSQL jaoks terminali käsu 1. Selleks, et leida utiliidi asukohta loodi kahte käsku:</p> <ul style="list-style-type: none"> • <code>ps aux grep `service {unit name} status grep -i pid grep -Po '[0-9]+'` 2>&1 grep -Po '/usr/.*'</code> • <code>`service status awk '\$1=="Loaded:"{print \$3}' grep -Po '(\\/[a-zA-Z]*[0-9]*[.]*[a-zA-Z]*[0-9]*)*(.service)' while read line; do grep 'ExecStart' „\$line“; done sed -n 's/.*ExecStart=//p' awk {print \$1}'` -V</code> <p>MySQL ja MariaDB jaoks valiti terminali käsu 2.</p>

Andmebaasidel kui andmete kogumitel on eraldi allikad. Nende kaardistamiseks tuleb andmebaasiserveriga luua ühendust. Selleks on tingimus, et andmebaasi instants oleks aktiivne. Välja töötati kahte käsku, esimene on PostgreSQL ja teine MySQL/MariaDB jaoks.

- `PGPASSWORD={password} psql -U {username} -h {host} -p {instance port} -c „\l+“`
- `mysql -u {username} -p{password} -e „SELECT TABLE_SCHEMA AS 'Database name', Round(Sum(DATA_LENGTH + INDEX_LENGTH) / 1024 / 1024, 0) AS 'Size (MB)' FROM information_schema.TABLES GROUP BY TABLE_SCHEMA;“`

4.3 Insight Discovery mustrite loomine ja testimine

Selleks, et alustada mustrite ehitamisega, tuli tutvuda nende nüanssidega. Enda kohandatud muster peab olema UTF-8 kodeeringuga ja “.pat” lõpuga fail, mis peab asuma Discovery “pattern” kataloogis ja määratud Discovery konfiguratsioonifailis <EnabledPatternFiles> sõlmel. Muster peab sisaldama nõutud meetodi “PerformAction”, mida Discovery kutsus välja objekti massiiviga, ja XML-sõlmeid:

- <PatternID> – mistahes unikaalne string;
- <ProcessType> – protsessi tüüp, millega muster andmeid kogub. Linux operatsioonisüsteemi puhul on “SSHExecute”;

- <PatternType> – Discovery objekti tüüp, mille eest muster vastutab. Antud töö puhul on see kogu aeg “Application”;
- <OrderNr> – mustri järjekorranumber. Iga objekti tüübil on oma järjekord;
- <ApplicationName> – tarkvara nimi, mille kohta andmeid kogutakse;
- <Processing> – funktsioon, mis sisaldab C# klassi, et töödelda käsu tulemusi Discovery objektiks, kasutades PatternExec klassi PerformAction meetodi. [5]

Andmebaaside kui andmete kogumite kaardistamiseks on oluline Insight Discovery-le liidese kaudu määrata kasutaja mandaadid andmebaasi serveriga ühenduse loomiseks, mis olid eelnevalt administraatorite poolt seadistatud serveritel. Selleks on vaja valida mandaadile tarkvara nime, mis oli määratud vastava mustri <ApplicationName> sõlmel, sh postgresql, mysql.

Lisaks on kindlasti vaja muuta ka olemasoleva “Linux_ApplicationServices” nimega mustris <PatternType> sõlme väärtust “Application” peale. Autori poolt loodud mustrid sõltuvad konfiguratsioonielementidest saadud sellest mustrist ning on väga oluline, et selle mustri järjekorranumber sõlmel <OrderNr> oleks väiksem uute omadest. Kui seda muudatust ei tehta ja <PatternType> jääb “ApplicationService”, siis Insight Discovery ei saa koguda nõutud konfiguratsioonielemente, sest “ApplicationService” tüübiga mustrid kasutab ta peale “Application” omadest. Teostatud muudatus midagi muud ei mõjuta kui järjekorranumbrit.

Mustrite testimiseks tuleb anda Insight Discovery kasutajale administratiivseid turvaõigused *sudo grep* käsude kasutamiseks iga hostil.

Lisa 3 annab ülevaadet autori poolt loodud mustrist Nginx veebiserverite kaardistamiseks. Virtuaalsete hostide seadistused võivad asuda üks-kõik millises konfiguratsioonifailis. Selleks, et tuvastada kasutatavaid konfiguratsioonifaile, on määratud mustris kaks käsku:

- `nginx -v` – kuvab parameetriga „--conf-path“ peamise Nginx konfiguratsioonifaili. [26]

- `grep 'include' {konfiguratsioonifail}` – sisaldab veebiserverite konfiguratsioonifailide asukohta. Traksides asuv sõltuv muutuja on peamine konfiguratsioonifail, mis oli saadud ülalpool käsust.

Peale failide tuvastamist muster sisetab neid järgmise käsule. Käesolev käsk kuvab vajalikud andmed kaardistamiseks:

- ```
awk '/server *{/{c=1; print;next} c&&/{c++} c&&/{c--} c' {failide
loetelu} | grep -Po
'^([^\#]*((server)\s*{${}})|(listen|server_name|ssl|root|return)\s+)).
*'
```

Lisa 4 annab ülevaadet PostgreSQL andmebaasi instantsite ja nendel paiknevate andmebaaside mustri koodist. Alguses püüakse käsuga „`ps aux | grep `service {unit name} status | grep -i pid | grep -Po '[0-9]+'` 2>&1 | grep -Po '/usr/.*'`“ kätte saada aktiivse andmebaasi instantsi parameetreid, sh peamise data kataloogi. Ebaõnnestustumisel kasutatakse alternatiivsed käsud. Kui peamine data kataloog on teada, siis edasi otsitakse seal asuva konfiguratsioonifailis pordi ja IP numbreid. Peale instantsi konfiguratsioonikirjete tuvastamist, kaardistatakse ka temal paiknevaid andmebaase, kui andmete kogumeid.

MySQL ja MariaDB jaoks oli loodud ühine muster, mis on välja toodud Lisa 5. Kõige pealt luuakse ühendust serveriga ja kaardistatakse tema versiooni. Seejärel üritab Discovery saada `/etc/my.cnf` või `/etc/mysql/my.cnf` asukohadega konfiguratsioonifailis pordi ja IP numbreid, samuti kontrollitakse ka alternatiivsete konfiguratsioonifailide olemasolu „`!include`“ väljal. Seejärel kaardistatakse instantsi andmebaase.

#### 4.4 Konfiguratsioonielementide ja nende seoste kaardistamine

Konfiguratsioonihalduse mudeli täiendamiseks uute konfiguratsioonielementidega tuleb eelnevalt luua nendele objekti tüübid (Object types). Vastavatele objekti tüüpidele tuleb Jira Insight skeemis seadistada tühja impordi, mida Insight nimetab kui „Object type mapping“, ilma atribuudideta. Objekti tüüpide impordi seadete täiendamiseks uute atribuutidega tuleb eelnevalt teostada andmete sünkroniseerimist ühe hostiga, kus on kaardistatud meile vajalikud andmed vastavate atribuutidega. Selliselt Insight salvestab uued atribuudid CMDB-sse, mida nimetab „data locator“-iks ja mida me muustrites

määrasime. Lisa 2 annab detailse ülevaade, kuidas oli täiendatud konfiguratsioonihalduse mudel ning seadistatud uued objekti tüübid, sh veebirakendused, andmebaasi ja nende instantsid. Iga joonise ülalpool on kuvatud peatükis 4.1 kaardistatud atribuudid, seejärel on kuvatud vastavate atribuutidele impordi seadistus ehk „Object type mapping“. Andmebaasi instantsitele on ka eraldi tehtud hosti ja tarkvara objekti tüüpides seose seadistus.

## **4.5 Andmete valideerimine**

Projekti skooopi valiti kahte teenust, mille põhjal autor valideeris saadud andmeid, sest kõigi teenuste andmete valideerimine on aeganõudev. Koostöös teenuste vastutajatega ehk süsteemiadministraatoritega tegi autor ülevaadet konfiguratsioonihalduse mudelis vastavate IT teenuste konfiguratsioonielementide kohta. Kuna tegu oli testkeskkonnaga ja veebirakenduste konfiguratsioonielemendid oli piiratud arv, siis mõni atribuut võis selle tõttu puududa, kuid praeguse seisuga ei tekita see mingit kahju. Samuti puudub oluline informatsioon IT teenuste poolt kasutatavate andmebaaside kohta, mida autor käesoleva töö ülesannete skooopi ei võtnud, kuid kindlasti on see prioriteedis projekti lähimas tulevikus.

## **4.6 Järeldused**

Lõputöö käigus olid saavutatud järgmised tulemused:

- Oli täiustatud Insight Discovery võimekus tänu loodud uute mustritega, sh Nginx rakenduse ja PostgreSQL, MySQL, MariaDB AB instantsite mustrid.
- Uued mustrid on kasutusel testkeskkonnas ning kaardistavad aktuaalset ja ajaloolist infot seni ligipääsetavate hostide osas. Lisaks sobivad lahenduseks teistele ettevõtetele.
- Oli täiendatud TEHIKu testkeskkonna konfiguratsioonihalduse mudel uute konfiguratsioonielementide kategooriatega, mis kajastab reaalselt olukorda ligipääsetavate IT teenuste PostgreSQL, MySQL, MariaDB andmebaasidest ja Nginx veebiserveritest.

- Lähimas tulevikus tagatud usaldusväärsem alus teistele IT halduse protsessidele – on lihtsam näha suuremat pilti, selgem ülevaade infosüsteemi komponentidest, abiks muudatuste tegemisel, ülevaade kasutatavast tarkvarast ja versioonidest, abiks infosüsteemide planeerimisel.

Lõputöö käigus selgus, et implementeerida mõned mustrid testkeskkonnas, sh Apache ja Tomcat, osutus liiga keeruliseks piiratud arvu andmetetest ja testimise võimalusest. Seetõttu autori arvamusel on otstarbekam kasutada näidiseks toodangu keskkonnas olevaid konfiguratsioonifailide sisu. Selleks tuleb küberturbe osakonnaga kooskõlastada ligipääsude saamist autorile ja andmete kasutamiseks. Samuti, mis puudutab MySQL ja MariaDB andmebaaside „Encoding“ ja „Description“ atribuute, siis nende kaardistamise võimalust autor ei suutnud välja töötada. See tähendab, et konfiguratsioonihaldaja poolt esitatud andmebaaside kaardistamisele nõuded polnud täielikult, kuid suurem osa ikkagi täidetud.

Seni välja töötatud konfiguratsioonihalduse mudel ja autori poolt konfiguratsioonihalduse mudelis tehtud täiendused osutub aluseks tootmiskeskkonnas tulevase mudeli jaoks.

## 5 Edaspidised sammud

Konfiguratsioonihalduse täiendamiseks on vaja kaardistada veel puudu olevaid konfiguratsioonielemendid. Konfiguratsioonihalduse protsessi juurutamise järgmine samm on autori poolt loodud lahendust integreerida toodangu keskkonna Insight Discovery-le ning importida konfiguratsioonihalduse mudeli toodangu Jira-sse. Seejärel tuleb monitoorida CMDB-sse liikuvaid andmeid mudelis ning neid valideerida koos teenuste vastutajatega. Puuduvate konfiguratsioonielementide kategooriate kaardistamiseks kirjutatakse jooksvalt mustreid juurde. On vajalik ka üle kontrollida hostide saadavust Insight Discovery-le alamvõrkude kaupa.

Eelnevalt mainitud muudatuste, intsidentide ja teenussoovide halduse ning probleemihalduse protsesside toe tagamiseks ning personali liikumisest tulenevaid teabe kadumise riski vähendamiseks tuleb leida viisi kaardistatud Insight Discovery poolt instantse seostada olemasoleva teenuste portfelliga, kuhu on koondatud teenuste vastutajad, ehk süsteemiadministraatorid, mis tagaks selgemat vastutuse jagamist ja haldamist iga töötajale igapäevatöös. Samuti on plaanis luua monitooringu CMDB alusel, mis saadaks välja intsidentide teavitusi vastutajatele ja tõstaks teenuste osutamise kvaliteedi.

## 6 Kokkuvõte

Käsitatud töö eesmärk oli täiendada konfiguratsioonihalduse protsessi veebirakenduste ja andmebaasi konfiguratsioonielemendi kategooriatega organisatsioonis TEHIK. Konfiguratsioonihaldus on vajalik tööriist keeruliste tarkvarasüsteemide haldamiseks. Konfiguratsioonihalduse puudumine või mitte täieliku juurutamise tõttu võivad kaasneda tõsised probleemid süsteemi töökindluse, tööaja ja süsteemi laiendamise võimega.

Soovitud eesmärgi saavutamiseks olid kirjutatud mustrid organisatsiooni TEHIK konfiguratsioonihalduse tarkvarale, milleks on Insight Discovery. Uute konfiguratsioonielementide kategooriatele olid loodud ja seadistatud konfiguratsioonihalduse mudelis vastavad objekti tüübid, kus kuvatakse CMDB-sse kaardistatud andmeid. Antud ülesannete täitmise jooksul tegeles töö autor suures koguses andmeanalüüsiga, eksperimendiga Linux terminali käsudega ning mustrite testimisega. Lahenduse välja töötamisest kuni rakendamiseni, suutis töö autor saavutada märkimisväärseid tulemusi, ehk kaardistada nõutele vastavaid konfiguratsioonikirjed. Täitmata aga jäid Apache ja Tomcat veebirakenduste kaardistamiseks püstitatud ülesanded, mis tuleneb vajaliku andmete ja korrektse testimise võimaluse puudusest testkeskkonnas.

Tänu lõputöös autori poolt välja töötatud lahendusele konfiguratsioonihalduse protsessile organisatsioon TEHIK suudab lähimas tulevikus hakkata vähendada enda kui ka klientide kulusid ja suurendada kindlustunnet tõusnud andmekvaliteediga. Samuti suurendada süsteemi üldist stabiilsust, vähendades intsidentide arvu ja muudatuste mõjut.

## Kasutatud kirjandus

- [1] „What Is Configuration Management and Why Is It Important?“, [Võrgumaterjal]. Available: <https://www.upguard.com/blog/5-configuration-management-boss>. [Kasutatud 14 04 2021].
- [2] „Why Configuration Management should be applied to your business“, Javelin Essentials, [Võrgumaterjal]. Available: <https://www.javelin-tech.com/3d/people/why-configuration-management/>. [Kasutatud 14 04 2021].
- [3] „Avaleht“, TEHIK, [Võrgumaterjal]. Available: <https://www.tehik.ee/>. [Kasutatud 03 04 2021].
- [4] „Meist“, TEHIK, [Võrgumaterjal]. Available: <https://www.tehik.ee/meist>. [Kasutatud 03 04 2021].
- [5] „Data collected by Discovery“, [Võrgumaterjal]. Available: <https://documentation.mindville.com/display/IDV229/Data+collected+by+Discovery>. [Kasutatud 10 03 2021].
- [6] „ITIL V2 to V3 Comparison“, [Võrgumaterjal]. Available: <https://www.itsm.info/ITIL%20V2%20to%20V3%20Comparison.pdf>. [Kasutatud 20 04 2021].
- [7] L. Klosterboer, Implementing ITIL Configuration Management, Upper Saddle River (N.J.): IBM Press/Pearson, 2008.
- [8] „Prioritization using MoSCoW“, [Võrgumaterjal]. Available: [https://cs.anu.edu.au/courses/comp3120/local\\_docs/readings/Prioritization\\_using\\_MoSCoW\\_AllAboutAgile.pdf](https://cs.anu.edu.au/courses/comp3120/local_docs/readings/Prioritization_using_MoSCoW_AllAboutAgile.pdf). [Kasutatud 21 04 2021].
- [9] „MySQL Workbench Manual“, [Võrgumaterjal]. Available: <https://dev.mysql.com/doc/workbench/en/wb-mysql-connections-methods-standard.html>. [Kasutatud 25 04 2021].
- [10] „PostgreSQL Documentation: 11“, [Võrgumaterjal]. Available: <https://www.postgresql.org/docs/11/app-postgres.html>. [Kasutatud 02 03 2021].
- [11] „Managing MySQL Server with systemd“, [Võrgumaterjal]. Available: <https://dev.mysql.com/doc/refman/5.7/en/using-systemd.html>. [Kasutatud 12 04 2021].
- [12] [Võrgumaterjal]. Available: <https://stackoverflow.com/questions/49381124/what-are-the-differences-between-lsof-and-netstat-on-linux>. [Kasutatud 20 04 2021].
- [13] [Võrgumaterjal]. Available: [http://nginx.org/en/docs/http/server\\_names.html](http://nginx.org/en/docs/http/server_names.html). [Kasutatud 10 05 2021].
- [14] „NGINX | Serving Static Content“, [Võrgumaterjal]. Available: <https://docs.nginx.com/nginx/admin-guide/web-server/serving-static-content/>. [Kasutatud 10 05 2021].
- [15] „How nginx processes a request“, [Võrgumaterjal]. Available: [http://nginx.org/en/docs/http/request\\_processing.html](http://nginx.org/en/docs/http/request_processing.html). [Kasutatud 10 05 2021].
- [16] „Apache Core Features“, [Võrgumaterjal]. Available: <https://httpd.apache.org/docs/current/mod/core.html>. [Kasutatud 10 05 2021].
- [17] „Apache Tomcat 8 Configuration Reference“, [Võrgumaterjal]. Available: <https://tomcat.apache.org/tomcat-8.0-doc/config/host.html>. [Kasutatud 10 05 2021].
- [18] „Changing Tomcat HTTP Port to 80“, [Võrgumaterjal]. Available: <https://www.baeldung.com/tomcat-change-port>. [Kasutatud 10 05 2021].
- [19] „Configuring HTTPS servers“, [Võrgumaterjal]. Available: [http://nginx.org/en/docs/http/configuring\\_https\\_servers.html](http://nginx.org/en/docs/http/configuring_https_servers.html). [Kasutatud 10 05 2021].

- [20] „Apache Module mod\_ssl,“ [Võrgumaterjal]. Available: [https://httpd.apache.org/docs/current/mod/mod\\_ssl.html#sslengine](https://httpd.apache.org/docs/current/mod/mod_ssl.html#sslengine). [Kasutatud 10 05 2021].
- [21] „SSL/TLS Configuration HOW-TO,“ [Võrgumaterjal]. Available: <https://tomcat.apache.org/tomcat-7.0-doc/ssl-howto.html>. [Kasutatud 10 05 2021].
- [22] [Võrgumaterjal]. Available: <https://www.postgresql.org/docs/8.3/app-postgres.html>. [Kasutatud 10 05 2021].
- [23] [Võrgumaterjal]. Available: <https://dev.mysql.com/doc/mysql-port-reference/en/mysql-ports-reference-tables.html>. [Kasutatud 10 05 2021].
- [24] [Võrgumaterjal]. Available: <https://stackoverflow.com/questions/8987679/how-to-retrieve-the-current-version-of-a-mysql-database-management-system-dbms>. [Kasutatud 10 05 2021].
- [25] [Võrgumaterjal]. Available: <https://stackoverflow.com/questions/13733719/postgresql-which-version-of-postgresql-am-i-running>. [Kasutatud 10 05 2021].
- [26] „How To Find Out Nginx Version using command line,“ [Võrgumaterjal]. Available: <https://www.cyberciti.biz/faq/unix-linux-bsd-display-nginx-version/>. [Kasutatud 10 05 2021].
- [27] [Võrgumaterjal]. Available: <https://stackoverflow.com/questions/13919209/parse-format-ps-eo-etime-in-minutes?rq=1>. [Kasutatud 09 05 2021].

## **Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks<sup>1</sup>**

Mina, Elina Pavlova

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose „TEHIK konfiguratsioonihalduse täiendamine“, mille juhendaja on Nadežda Furs
  - 1.1. reprodutseerimiseks lõputöö säilitamise ja elektroonse avaldamise eesmärgil, sh Tallinna Tehnikaülikooli raamatukogu digikogusse lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
  - 1.2. üldsusele kättesaadavaks tegemiseks Tallinna Tehnikaülikooli veebikeskkonna kaudu, sealhulgas Tallinna Tehnikaülikooli raamatukogu digikogu kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.
2. Olen teadlik, et käesoleva lihtlitsentsi punktis 1 nimetatud õigused jäävad alles ka autorile.
3. Kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest ning muudest õigusaktidest tulenevaid õigusi.

17.05.2021

---

<sup>1</sup> Lihtlitsents ei kehti juurdepääsupiirangu kehtivuse ajal vastavalt üliõpilase taotlusele lõputööle juurdepääsupiirangu kehtestamiseks, mis on allkirjastatud teaduskonna dekaani poolt, välja arvatud ülikooli õigus lõputööd reprodutseerida üksnes säilitamise eesmärgil. Kui lõputöö on loonud kaks või enam isikut oma ühise loomingulise tegevusega ning lõputöö kaas- või ühisautor(id) ei ole andnud lõputööd kaitsvale üliõpilasele kindlaksmääratud tähtajaks nõusolekut lõputöö reprodutseerimiseks ja avalikustamiseks vastavalt lihtlitsentsi punktidele 1.1. ja 1.2, siis lihtlitsents nimetatud tähtaja jooksul ei kehti.



## Lisa 2 – Nginx veebirakenduse serveri muster

| Veebiserver    |                         | Create Object          |         | Objects               | Attributes       | Graph | Object Type |
|----------------|-------------------------|------------------------|---------|-----------------------|------------------|-------|-------------|
| No description |                         |                        |         |                       |                  |       |             |
|                |                         |                        |         |                       |                  |       |             |
| Id             | Name                    | Description            | Type    | Type Value            | Additional Value |       |             |
| 16135          | Key                     |                        | Default | Text                  |                  |       |             |
| 16136          | Name                    | The name of the object | Default | Text                  | ⚙️               |       |             |
| 16137          | Created                 |                        | Default | DateTime              |                  |       |             |
| 16138          | Updated                 |                        | Default | DateTime              |                  |       |             |
| 16215          | Status                  |                        | Status  |                       | ⚙️               |       |             |
| 16257          | Root                    |                        | Default | Text                  | ⚙️               |       |             |
| 16258          | Port                    |                        | Default | Text                  | ⚙️               |       |             |
| 16357          | SSL                     |                        | Default | Boolean               | ⚙️               |       |             |
| 16358          | ObjectIdentifier        |                        | Default | Text                  | ⚙️               |       |             |
| 16359          | Referenced Applications |                        | Object  | Installed application | Application      | ⚙️    |             |
| 16360          | Referenced Host         |                        | Object  | Server                | Host             | ⚙️    |             |

Default Text

| Veebiserver                                |                | ENABLED |  |
|--------------------------------------------|----------------|---------|--|
| <b>Selector</b>                            | APPLICATION    |         |  |
| <b>Filter data source by IQL</b>           | No Value       |         |  |
| <b>Missing objects</b>                     | Update         |         |  |
| <b>Attribute to edit</b>                   | Status         |         |  |
| <b>New value of attribute to edit</b>      | Not Found      |         |  |
| <b>Threshold Type</b>                      | Days           |         |  |
| <b>Threshold Number</b>                    | 2              |         |  |
| <b>Missing objects outbound references</b> | Remove         |         |  |
| <b>Empty Values</b>                        | Ignore         |         |  |
| <b>Unknown Values</b>                      | Use default    |         |  |
| <b>Match Identifier</b>                    | Case sensitive |         |  |

| Identifier                          | Data Locator            | Insight Attribute       | Object mapping (IQL)                                     | Description |
|-------------------------------------|-------------------------|-------------------------|----------------------------------------------------------|-------------|
| <input checked="" type="checkbox"/> | Web Name                | Name                    |                                                          | ⚙️          |
| <input type="checkbox"/>            | Port                    | Port                    |                                                          | ⚙️          |
| <input type="checkbox"/>            | Status                  | Status                  |                                                          | ⚙️          |
| <input checked="" type="checkbox"/> | Root                    | Root                    |                                                          | ⚙️          |
| <input checked="" type="checkbox"/> | Object Hash             | ObjectIdentifier        |                                                          | ⚙️          |
| <input type="checkbox"/>            | Referenced Applications | Referenced Applications | "ObjectIdentifier" IN (\${Referenced Applications\${0}}) | ⚙️          |
| <input type="checkbox"/>            | Hosts Connected         | Referenced Host         | "ObjectIdentifier" IN (\${Hosts Connected\${0}})         | ⚙️          |
| <input type="checkbox"/>            | SSL                     | SSL                     |                                                          | ⚙️          |

**Instants** Create Object Objects Attributes Graph Object Type

*No description*

| Id    | Name             | Description            | Type    | Type Value | Additional Value |
|-------|------------------|------------------------|---------|------------|------------------|
| 16135 | Key              |                        | Default | Text       |                  |
| 16136 | Name             | The name of the object | Default | Text       | ⚙️               |
| 16137 | Created          |                        | Default | DateTime   |                  |
| 16138 | Updated          |                        | Default | DateTime   |                  |
| 16215 | Status           |                        | Status  |            | ⚙️               |
| 16350 | Instance IP      |                        | Default | Text       | ⚙️               |
| 16351 | Port             |                        | Default | Text       | ⚙️               |
| 16245 | Service Uptime   |                        | Default | Text       | ⚙️               |
| 16246 | Version          |                        | Default | Text       | ⚙️               |
| 16247 | ObjectIdentifier |                        | Default | Text       | ⚙️               |
| 16248 | Host Object Hash |                        | Default | Text       | ⚙️               |

Default Text Add

**Instants** ENABLED ⚙️

**Selector** DATABASE

**Filter data source by IQL** *No Value*

**Missing objects** Update

**Attribute to edit** Status

**New value of attribute to edit** Not Found

**Threshold Type** Days

**Threshold Number** 2

**Missing objects outbound references** Remove

**Empty Values** Use default

**Unknown Values** Use default

**Match Identifier** Case sensitive

| Identifier                          | Data Locator     | Insight Attribute | Object mapping (IQL) | Description |
|-------------------------------------|------------------|-------------------|----------------------|-------------|
| <input checked="" type="checkbox"/> | Server Name      | Name              |                      | ⚙️          |
| <input type="checkbox"/>            | Status           | Status            |                      | ⚙️          |
| <input type="checkbox"/>            | Instance Name    | Instance IP       |                      | ⚙️          |
| <input type="checkbox"/>            | Port             | Port              |                      | ⚙️          |
| <input type="checkbox"/>            | Uptime           | Service Uptime    |                      | ⚙️          |
| <input type="checkbox"/>            | Host Object Hash | Host Object Hash  |                      | ⚙️          |
| <input type="checkbox"/>            | Object Hash      | ObjectIdentifier  |                      | ⚙️          |
| <input checked="" type="checkbox"/> | Version          | Version           |                      | ⚙️          |

Add

**Installed application**  
No description

**Create Object** Objects **Attributes** Graph Object Type ▾

|  |       |                   |        |          |                              |  |
|--|-------|-------------------|--------|----------|------------------------------|--|
|  | 15875 | Database Instance | Object | Instants | Referenced Database Instance |  |
|--|-------|-------------------|--------|----------|------------------------------|--|

| Identifier               | Data Locator                                          | Insight Attribute | Object mapping (IQL)                                                                                       | Description |
|--------------------------|-------------------------------------------------------|-------------------|------------------------------------------------------------------------------------------------------------|-------------|
| <input type="checkbox"/> | <b>Referenced Databases</b><br><b>Hosts Connected</b> | Database Instance | "ObjectIdentifier" IN (\${Referenced Databases\${0}}) AND "Host Object Hash" IN (\${Hosts Connected\${0}}) |             |

**Server**  
No description

**Create Object** Objects **Attributes** Graph Object Type ▾

|  |       |                     |        |          |                              |  |
|--|-------|---------------------|--------|----------|------------------------------|--|
|  | 16240 | Andmebaasi Instants | Object | Instants | Referenced Database Instance |  |
|--|-------|---------------------|--------|----------|------------------------------|--|

| Identifier               | Data Locator       | Insight Attribute   | Object mapping (IQL)                 | Description |
|--------------------------|--------------------|---------------------|--------------------------------------|-------------|
| <input type="checkbox"/> | <b>Object Hash</b> | Andmebaasi Instants | "Host Object Hash" = \${Object Hash} |             |

| Id    | Name             | Description            | Type    | Type Value | Additional Value                |
|-------|------------------|------------------------|---------|------------|---------------------------------|
| 16135 | Key              |                        | Default | Text       |                                 |
| 16136 | Name             | The name of the object | Default | Text       | ⚙️                              |
| 16137 | Created          |                        | Default | DateTime   |                                 |
| 16138 | Updated          |                        | Default | DateTime   |                                 |
| 16215 | Status           |                        | Status  |            | ⚙️                              |
| 16251 | Description      |                        | Default | Text       | ⚙️                              |
| 16252 | Size MB          |                        | Default | Text       | ⚙️                              |
| 16253 | Encoding         |                        | Default | Text       | ⚙️                              |
| 16254 | Instance         |                        | Object  | Instances  | Referenced Database Instance ⚙️ |
| 16255 | ObjectIdentifier |                        | Default | Text       | ⚙️                              |

Default ▾ Text ▾ Add

**Selector** DATABASE  
**Filter data source by IQL** No Value  
**Missing objects** Remove  
**Threshold Type** Days  
**Threshold Number** 1  
**Empty Values** Use default  
**Unknown Values** Use default  
**Match Identifier** Case sensitive

| Identifier                          | Data Locator        | Insight Attribute | Object mapping (IQL)                                    | Description |
|-------------------------------------|---------------------|-------------------|---------------------------------------------------------|-------------|
| <input checked="" type="checkbox"/> | Name                | Name              |                                                         | ⚙️          |
| <input type="checkbox"/>            | Description         | Description       |                                                         | ⚙️          |
| <input type="checkbox"/>            | Size                | Size MB           |                                                         | ⚙️          |
| <input type="checkbox"/>            | Encoding            | Encoding          |                                                         | ⚙️          |
| <input checked="" type="checkbox"/> | Server Name Version | Instance          | "Name" = \${Server Name} AND<br>"Version" = \${Version} | ⚙️          |
| <input type="checkbox"/>            | Object Hash         | ObjectIdentifier  |                                                         | ⚙️          |

▾  ▾   Add

## Lisa 3 – Nginx veebirakenduse serveri muster

Allpool on ülevaade ainult osa koodist. Tervet koodi saab uurida ning vajadusel alla laadida käesoleva lingi kaudu: [https://github.com/elaina0/TEHIK-Insight-Discovery-patterns/blob/main/Linux\\_Application\\_Nginx-TEHIK.pat](https://github.com/elaina0/TEHIK-Insight-Discovery-patterns/blob/main/Linux_Application_Nginx-TEHIK.pat)

```
<?xml version="1.0" encoding="utf-8"?>
<!--
@ Atlassian
-->
<ScanPattern xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
 <Version>1.2.0</Version>
 <PatternID>TEHIK-Linux-3-nginx</PatternID>
 <OrderNr>250</OrderNr>
 <ProcessType>SSHExecute</ProcessType>
 <PatternType>Application</PatternType>
 <ApplicationName>nginx</ApplicationName>
 <Processing>
 <![CDATA[
 using System;
 using System.Collections.Generic;
 using Insight.Discovery.Tools;
 using Insight.Discovery.InfoClasses;
 using Insight.Discovery.ProviderClasses;
 using Insight.Discovery.InfoClasses.CommandResult.ResultTypes;

 namespace Insight.Discovery {
 public class PatternExec {
 public void PerformAction(object[] parameters)
 {
 try
 {
 HostInfo host = ((HostInfo)parameters[2]);
 SSHProvider sshProvider = (SSHProvider)parameters[1];
 SSHExecuteResult sshExecuteResult;
 string internalCommand;
 string commandResult;

 // Find Nginx service
 ApplicationServiceInfo nginxService = host.ApplicationServices.Find(o =>
o.Name.Contains("nginx"));

 if (nginxService != null) {
 // Find all server Nginx applications
 List<ApplicationInfo> nginxApp = host.Applications.FindAll(o =>
o.Name.ToLower().Contains("nginx") && o.Description.ToLower().Contains("server"));
 // List for configuration files path
 List<string> Paths = new List<string>();

 // Find Nginx main configuration path
 internalCommand = "nginx -V";
 sshExecuteResult =
(SSHExecuteResult)sshProvider.ExecuteCommand(internalCommand);
 commandResult = sshExecuteResult;
 sshExecuteResult.LogResult();

 if (!string.IsNullOrEmpty(commandResult)) {
 string[] line = commandResult.Split('\n');

 for (int x = 0; x < line.Length; x++) {
 if (line[x].Contains("conf-path=")) {
 int start = line[x].IndexOf("conf-path=") + 10;

```

```

 int end = line[x].IndexOf(" ", start);
 Paths.Add(line[x].Substring(start, end - start));
 }
}
}

```

## Lisa 4 – PostgreSQL AB instantsi muster

Allpool on ülevaade ainult osa koodist. Tervet koodi saab uurida ning vajadusel alla laadida käesoleva lingi kaudu: [https://github.com/elaina0/TEHIK-Insight-Discovery-patterns/blob/main/Linux\\_Application\\_PostgreSQL-TEHIK.pat](https://github.com/elaina0/TEHIK-Insight-Discovery-patterns/blob/main/Linux_Application_PostgreSQL-TEHIK.pat)

```

<?xml version="1.0" encoding="utf-8"?>
<!--
@ Mindville
-->
<ScanPattern xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
 <Version>1.2.0</Version>
 <PatternID>TEHIK-Linux-1-postgreSQL</PatternID>
 <OrderNr>205</OrderNr>
 <ProcessType>SSHExecute</ProcessType>
 <PatternType>Application</PatternType>
 <ApplicationName>postgresql</ApplicationName>
 <Processing><![CDATA[
 using System;
 using System.Collections.Generic;
 using System.Linq;
 using Insight.Discovery.Tools;
 using Insight.Discovery.InfoClasses;
 using Insight.Discovery.ProviderClasses;
 using Insight.Discovery.InfoClasses.CommandResult.ResultTypes;

 namespace Insight.Discovery {
 public class PatternExec {
 public void PerformAction(object[] parameters)
 {
 try
 {
 HostInfo host = ((HostInfo)parameters[2]);
 SSHProvider sshProvider = (SSHProvider)parameters[1];
 SSHExecuteResult sshExecuteResult;
 string internalCommand;
 string commandResult;
 // Variable to check whether we need to use additional code for Patroni
 bool isPatroni = false;

 if (!host.ApplicationServices.IsNullOrEmpty())
 {
 var ServiceNames = new List<string>();

 // Does postgresql run on Patroni?
 ApplicationInfo patroni = host.Applications.Find(o => o.Name ==
("patroni") && o.Description.ToLower().Contains("postgresql"));
 if (patroni != null) {

```

```

 ServiceNames.Add(host.ApplicationServices.Find(o => o.Name ==
"patroni").Name);
 isPatroni = true;
 }
 else {
 List<ApplicationServiceInfo> Services =
host.ApplicationServices.FindAll(o => o.Name.Contains("postgresql-") || o.Name ==
"postgresql");
 // Get every service name in string and add them to ServiceNames
list
 foreach (ApplicationServiceInfo i in Services) {
ServiceNames.Add(i.Name); }
 }

 // Find information for each instance in ServiceNames
for (int i = 0; i < ServiceNames.Count; i++)
{
 Server InstanceInfo = new Server() { data_dir = "", bin_dir = "",
version = "", ip = "", port = "", status = "", uptime = ""};
 string[] lines;
 string[] lines2;
 // Variable to check whether localhost is allowed to connect to
DB.
 bool nolocalhost = false;

```

## Lisa 5 – MySQL/MariaDB AB instantsite muster

Allpool on ülevaade ainult osa koodist. Tervet koodi saab uurida ning vajadusel alla laadida käesoleva lingi kaudu: [https://github.com/elaina0/TEHIK-Insight-Discovery-patterns/blob/main/Linux\\_Application\\_mySQL-TEHIK.pat](https://github.com/elaina0/TEHIK-Insight-Discovery-patterns/blob/main/Linux_Application_mySQL-TEHIK.pat)

```

<?xml version="1.0" encoding="utf-8"?>
<!--
© Mindville
-->
<ScanPattern xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
 <Version>1.1.0</Version>
 <PatternID>TEHIK-Linux-2-mysql</PatternID>
 <OrderNr>206</OrderNr>
 <ProcessType>SSHExecute</ProcessType>
 <PatternType>Application</PatternType>
 <ApplicationName>mysql</ApplicationName>
 <Processing><![CDATA[
 using System;
 using System.Collections.Generic;
 using Insight.Discovery.Tools;
 using Insight.Discovery.InfoClasses;
 using Insight.Discovery.ProviderClasses;
 using Insight.Discovery.InfoClasses.CommandResult.ResultTypes;

 namespace Insight.Discovery {
 public class PatternExec {
 public void PerformAction(object[] parameters)
 {
 try

```

```

 {
 HostInfo host = ((HostInfo)parameters[2]);
 SSHProvider sshProvider = (SSHProvider)parameters[1];
 SSHExecuteResult sshExecuteResult;
 string internalCommand;
 string commandResult;

 if (!host.ApplicationServices.IsNullOrEmpty())
 {
 Server InstanceInfo = new Server() { config = "", version = "", ip =
"", port = "", status = "", uptime = ""};
 ApplicationServiceInfo Service;

 // Running MariaDB/MySQL instance version
 internalCommand = @"mysql -u $$login$$ -p$$password$$ -e ""select
version();""";
 sshExecuteResult =
(SSHExecuteResult)sshProvider.ExecuteCommand(internalCommand);
 commandResult = sshExecuteResult;
 sshExecuteResult.LogResult();

 if (!string.IsNullOrEmpty(commandResult) &&
commandResult.Contains("version")) {
 if (commandResult.Contains("--+\r\n")) // result containing frame
 {
 string[] lines = commandResult.Replace("\r", "").Split('\n');

 for (int i = 1; i < lines.Length; i++)
 {
 if (!string.IsNullOrEmpty(lines[i]) &&
!lines[i].Contains("version") && lines[i].Contains("|"))
 {
 InstanceInfo.version = lines[i].Replace("|",
"".Trim());
 }
 }
 }
 else // without frames
 {
 InstanceInfo.version = commandResult.Split('\n')[1].Trim();
 }

 if (InstanceInfo.version.Contains("-Maria")) {
 InstanceInfo.version = InstanceInfo.version.Split("-")[0];
 Service = host.ApplicationServices.Find(o =>
o.Name.ToLower().Contains("mariadb"));
 }
 else {
 Service = host.ApplicationServices.Find(o =>
o.Name.ToLower().Contains("mysql"));
 }
 }
 }
 }

```