



VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

FUNDAMENTINIŲ MOKSLŲ FAKULTETAS

INFORMACINIŲ SISTEMŲ KATEDRA

**DUOMENŲ ATSARGINĖS KOPIJOS/SIEM**

Namų darbas nr. 4

Atliko: ITSfm-22 grupės studentas Aurimas Šakalys

Tikrino: lektorius Vitalijus Gurčinas

Vilnius, 2022

# Turinys

<b>1</b>	<b>Duomenų atsruginės kopijos</b>	<b>3</b>
1.1	Programinės įrangos pasirinkimas	3
1.2	Programinės įrangos diegimas ir konfigravimas	3
1.3	Užduoties reikalavimų įgyvendinimas	5
<b>2</b>	<b>SIEM</b>	<b>5</b>
2.1	Programinės įrangos pasirinkimas	5
2.2	Programinės įrangos diegimas ir konfigravimas	6
2.3	Užduoties reikalavimų įgyvendinimas	7

## Iliustracijų sąrašas

1	<i>borg</i> repozitorijose saugomi archyvai	5
2	<i>Wazuh</i> įrankio komponentinė diagrama	6
3	Žurnalinių įrašų agregacijos konfigracija	7
4	Saugos įvykių nurodytose direktorijose agregacijos konfigracija	7
5	<i>CIS</i> patikros rezultatai <i>wazuhAgent1</i> agentui	7

## Ištraukų sąrašas

1	SSH rakto apribojimas vienai komandai	3
2	<i>Borgmatic</i> konfigracija	3
3	<i>cron</i> įrašas, kas minutę kviečiantis <i>Borgmatic</i>	4

# 1. Duomenų atsarginės kopijos

## 1.1. Programinės įrangos pasirinkimas

Atsarginėms duomenų kopijoms daryti pasirinkau du įrankius - *borg* ir *Borgmatic*.

*borg* įrankis skirtas daryti atsargines duomenų kopijas. Įrankis veikia *serverio-kliento* principu, kur serveris atlieka duomenų saugojimo užduočių vykdymą, o klientas - deduplikaciją, šifravimą, suspaudimą ir pan.

*Borgmatic* įrankis skirtas gerokai palengvinti darbą su *borg* atsarginių kopijų darymo įrankiu. Šis įrankis suteikia vartotojui paprastą vartotojo sąsają ir konfigūraciją. Įrankį galima sukonfigūruoti įvairiapusiškai, įjungiant ar išjungiant funkcionalumą, kurį suteikia *borg* įrankis.

## 1.2. Programinės įrangos diegimas ir konfigūravimas

Virtualios mašinos kuriamos naudojant *Vagrant* įrankį, o pačios mašinos yra automatiškai sukonfigūruojamos naudojant *Ansible* pjesėmis.

Norint įgyvendinti pirmąjį namų darbo reikalavimą, nusprendžiau atlikti kopijų darymą naudojant **3-1-2** strategiją. Ši strategija įgyvendinta lokaliai sukuriant dvi virtualias mašinas, į kurias bus talpinamos atsarginės kopijos. Nors praktine prasme abi *borg* serverio virtualios mašinos egzistuoja toje pačioje vietoje, šias virtualias mašinas galima būtų paleisti ir sukonfigūruoti debesų kompiuterijos infrastruktūroje, taip abu atsarginių kopijų serverius patalpinant skirtinguose vietose. Taip **3-1-2** strategija būtų įgyvendinama praktiškai.

Prieš pradedant atsarginių kopijų sistemos diegimą, klientinėje mašinoje privaloma sugeneruoti SSL raktų porą, kuri bus naudojama autentifikuojantis siunčiant duomenis į atsarginių kopijų serverį. Kiekvieno kliento raktas turi būti įdiegtas į atsarginių kopijų darymo serverius. Diegiant viešąjį kliento raktą, šis raktas yra apribojamas *tik* atsarginių kopijų darymo tikslams pridėdant šį apribojimą matomą Ištrauka 1. Tokiu būdu užtikrinama, kad SSL raktų pora negalės būti panaudota kitiems tikslams serveryje. Tolimesni žingsniai apibūdinami tariant, jog tarp serverio ir kliento galima užmegzti saugų SSH ryšį.

**Ištrauka 1:** SSH rakto apribojimas vienai komandai

```
1 command="borg serve --restrict-to-path /var/local/backups/backup.\n  borg",restrict MIIDDjCCAfY...
```

Serverinėse mašinose paprasčiausiai yra įdiegiama *borg* programinė įranga, kuri leis atlikti kopijų darymą klientinei mašinai.

Klientinėje mašinoje, įdiegiami *borg* ir *Borgmatic* įrankiai. *Borgmatic* įrankis yra sukonfigūruojamas naudojant konfigūracinį failą matomą Ištrauka 2.

**Ištrauka 2:** *Borgmatic* konfigūracija

```
1 location:\n2   source_directories:
```

```

3      - /home
4      - /etc
5      - /var/log/syslog*
6  repositories:
7      - ssh://vagrant@192.168.0.202/var/local/backups/backup.
        borg
8      - ssh://vagrant@192.168.0.203/var/local/backups/backup.
        borg
9      - /var/local/backups/backup.borg
10
11 storage:
12     encryption_passphrase: "encryption_password"
13     ssh_command: ssh -i /home/vagrant/.ssh/id_rsa -o
        StrictHostKeyChecking=accept-new
14
15 retention:
16     keep_minutely: 60
17     keep_daily: 7
18     keep_monthly: 6
19     keep_yearly: 1

```

Konfiguracija nurodo:

- **source\_directories** - kurioms direktorijoms/failams norime sudaryti atsarginių duomenų kopijas;
  - Šiuo atveju sudarysime direktorių/failų */home*, */etc* ir */var/log/syslog\** atsargines kopijas.
- **repositories** - į kurias *borg* repozitorijas norėtume talpinti atsargines kopijas;
  - Šiuo atveju, dvi atsarginės kopijos talpinamos į du atsarginių kopijų serverius, viena talpinama lokaliai.
- **storage** - nurodo šifravimo slaptažodį ir komandą, naudojamą prisijungti prie atsarginių kopijų serverių;
- **retention** - nurodo kiek, kokio tipo atsarginių kopijų archyvų paliksime.<sup>1</sup>

Atlikus įrankio konfigūraciją telieka įdiegti *cron* įrašą (Ištrauka 3), kuris kas tam tikrą laiką pakviestų *Borgmatic* įrankį, taip atliekant duomenų kopijų sudarymą ir išsaugojimą.

**Ištrauka 3:** *cron* įrašas, kas minutę kviečiantis *Borgmatic*

```

1  */1 * * * * root PATH=$PATH:/usr/bin:/usr/local/bin /usr/
    local/bin/borgmatic --verbosity -1 --syslog-verbosity 1

```

<sup>1</sup>Norėčiau atkreipti dėmesį, jog visu 100% nesuprantu, kokia metodologija yra išlaikomi deduplikuoti duomenys po senų archyvų pašalinimo. *borg* įrankis savo dokumentacijoje šio mechanizmo neaprašo.

```
vagrant@ubuntu-focal:~$ sudo borgmatic list
ssh://vagrant@192.168.0.202/var/local/backups/backup.borg: Listing archives
ubuntu-focal-2022-12-10T14:20:05.940152 Sat, 2022-12-10 14:20:07 [04f361a5807bbb302625e87d54ebfd20bc
f07cd5ce15ceb1a35ec956c6adb1ba]
ubuntu-focal-2022-12-10T15:11:05.884029 Sat, 2022-12-10 15:11:07 [4f5fa35459c14bc7cec5904ef7740e84b3
7b3c483967e118d5b3352030cef3e7]
ubuntu-focal-2022-12-12T18:25:26.362540 Mon, 2022-12-12 18:25:27 [ab62b89fb32eda1829df734496676445cc
fe2a1d0ac7836d5646f7fbb6b8a878]
ubuntu-focal-2022-12-12T18:26:05.515293 Mon, 2022-12-12 18:26:06 [aa5c361fd3f27a1331ee4e804265fcf274
fb7c958d9a31839aa195c211633fe9]
ssh://vagrant@192.168.0.203/var/local/backups/backup.borg: Listing archives
ubuntu-focal-2022-12-10T14:20:15.450906 Sat, 2022-12-10 14:20:16 [5e6d3b1b983ec36173e5f2626a93768330
4b7cd6a98e4d91260a07f17fda05d5]
ubuntu-focal-2022-12-10T15:11:13.033037 Sat, 2022-12-10 15:11:14 [80210a2da4bbcd3c72b09c560c1fb2aa11
138115f7a64d7b3843fa9c185bdfba]
ubuntu-focal-2022-12-12T18:25:33.034875 Mon, 2022-12-12 18:25:34 [097d50a2bc94956eaa6a2c8fc600d3ea1c
2884da1b80372d09218b6599f4c90a]
ubuntu-focal-2022-12-12T18:26:12.820958 Mon, 2022-12-12 18:26:14 [9371025b05672fc933e23073be8037a2ae
4be680e947c0899aa0c1a154720ba3]
/var/local/backups/backup.borg: Listing archives
ubuntu-focal-2022-12-10T14:20:22.500129 Sat, 2022-12-10 14:20:22 [a33f1839a21c1b5794481c1d8e9cc59e28
23c4adce32a3a5d9d4673c11b411e0]
ubuntu-focal-2022-12-10T15:11:17.994367 Sat, 2022-12-10 15:11:18 [21587514cf2ed612878fe6ab907ae4bec3
33d29ba37da9296425e1a318166ac6]
ubuntu-focal-2022-12-12T18:25:23.923728 Mon, 2022-12-12 18:25:24 [95060d046b37d34422efad9d833552e1bb
2406f0026397094b2698e54d602ffb]
ubuntu-focal-2022-12-12T18:26:17.543431 Mon, 2022-12-12 18:26:17 [c54f0049aa44c9754e0407f4354f8d1f81
805f3f525edb61908c26cbd2b7e1c7]
```

1 pav. *borg* repozitorijose saugomi archyvai

### 1.3. Užduoties reikalavimų įgyvendinimas

Atsarginės kopijos atliekamos daromos naudojant **3-1-2** strategiją, taip įgyvendinant pirmąjį reikalavimą. Kadangi *borg* įrankis yra deduplikuojantis pagal nutylėjimą ir yra padaromos tik pakeistų ar naujų failų atsarginės kopijos, o duomenys yra šifruojami, yra įgyvendinami antras ir penktasis reikalavimai. *borg* įrankis sudaro šifruotas ir inkrementalias atsargines kopijas, taip apsisaugant nuo išpirkos atakų ir įgyvendinant ketvirtą reikalavimą.

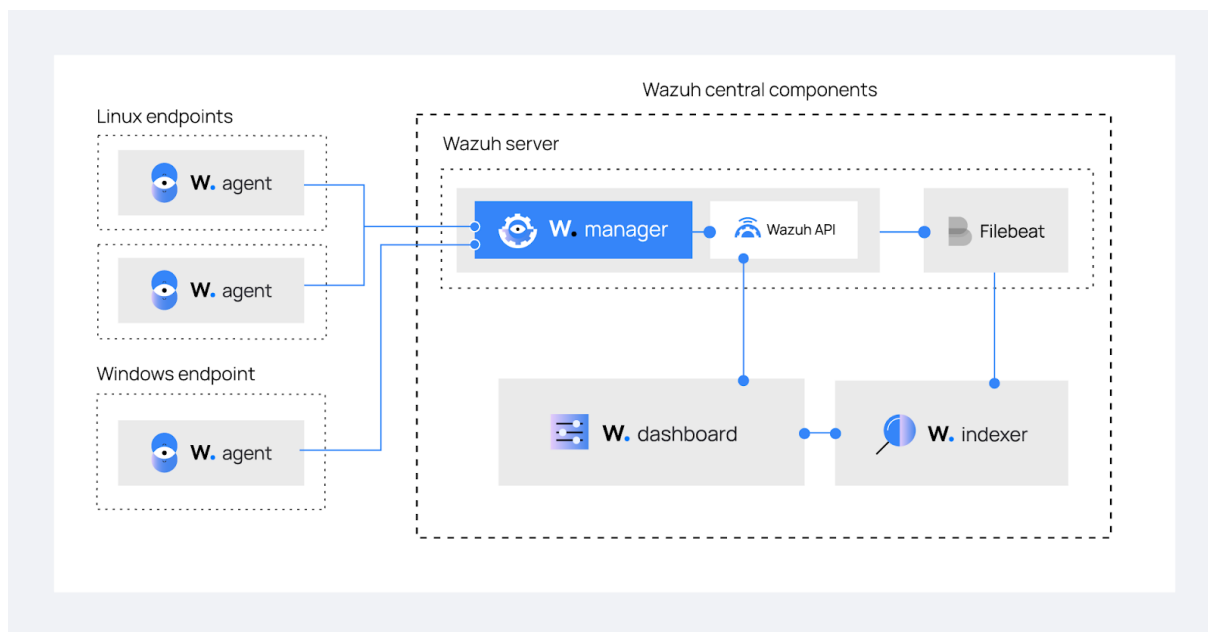
Trečiasis reikalavimas įgyvendinamas kiek sudetingiau. Kadangi yra paliekami dienos (septynios), savaitės (keturios) ir mėnesio (šešios) atsarginės kopijos, duomenis galima atkurti naudojant juos, tačiau šie nebūtinai gali vadintis pilnomis atsarginių duomenų kopijomis. Norint pilnai atitikti reikalavimą, kiekvienos savaitės pilnam archyvui tektų sukurti po *borg* repozitoriją. Tai nėra patogu, tačiau tai užtikrintų šimtaprocentinį reikalavimo įgyvendinimą.

Atsarginės duomenų kopijos yra daromos automatiškai, todėl įgyvendinamas ir papildomas reikalavimas.

## 2. SIEM

### 2.1. Programinės įrangos pasirinkimas

SIEM sistemos įgyvendinimui pasirinkau *Wazuh* įrankį dėl paprasto diegimo ir naudojimo.



2 pav. Wazuh įrankio komponentinė diagrama

*Wazuh* įrankis susideda iš kelių komponentų. *Wazuh agent* yra komponentas, atsakingas už saugos įvykių registravimą ir žurnalinių įrašų stebėjimą klientinėje mašinoje. Šis komunikuoja su *Wazuh manager* komponentu, kuris perduoda gautus duomenis iš klientinių mašin indeksatoriui. Besinaudojant indeksatoriaus duomenimis *Wazuh dashboard* vartotojo sąsajos komponentas atvaizduoja šiuos duomenis įvarus tipo diagramose ir pan.

## 2.2. Programinės įrangos diegimas ir konfigūravimas

Virtualios mašinos kuriamos naudojant *Vagrant* įrankį, o pačios mašinos yra automatiškai sukonfigūruojamos naudojant *Ansible* pjesėmis.

Naudojamos dvi virtualios mašinos, viena - *Wazuh* serveris, kita - *Wazuh Ubuntu Linux* klientas.

Diegimas vyksta keliais etapais. Pirmojo etapo metu yra sugeneruojami *Wazuh* serverio sertifikatai, šie eksportuojami panaudojimui klientinėse mašinose. Šiuos sugeneravus, įdiegiami serverio komponentai, indeksatorius ir *Wazuh dashboard*, kartu ir *Wazuh manager* su *Filebeat*.

Sudiegus serverį, galima pradėti konfigūruoti klientines mašinas, į jas diegiant *Wazuh agent*. Visi veiksmai atliekami naudojantis *Wazuh* suteikiamomis *Ansible* pjesėmis ir rolėmis.

Konfigūracija kuri sudiegama naudojantis pateikiamomis pjesėmis yra tvarkinga ir atitinkanti didžiąją dalį užduoties reikalavimų. Figure 3 galime matyti jog sukonfigūruojami *syslog* tipo žurnalinių įrašų nuskaitymas iš nurodytų failų. Šie įvykiai yra išsaugomi *Wazuh* serveryje ir juos galima pamatyti *Wazuh dashboard*. Figure 4 galime matyti saugos įvykių generavimo konfigūraciją, šiose direktorijose atlikti veiksmai sugeneruotu įvykius, kurie būtų patalpinti serveryje.

*Wazuh* nesuteikia funkcionalumo automatiškai ištrinti žurnalinius įrašus ar įvykius, tačiau tai galima atlikti naudojant *cronjob* funkcionalumą.

```

<localfile>
  <log_format>syslog</log_format>
  <location>/var/log/auth.log</location>
</localfile>

<localfile>
  <log_format>syslog</log_format>
  <location>/var/log/syslog</location>
</localfile>

<localfile>
  <log_format>syslog</log_format>
  <location>/var/log/dpkg.log</location>
</localfile>

<localfile>
  <log_format>syslog</log_format>
  <location>/var/log/kern.log</location>
</localfile>

```

3 pav. Žurnalinių įrašų agregacijos konfigūracija

```

<syscheck>
  <disabled>no</disabled>
  <frequency>43200</frequency>
  <scan_on_start>yes</scan_on_start>
  <!-- Directories to check (perform all possible verifications) -->
  <directories >/etc,/usr/bin,/usr/sbin</directories>
  <directories >/bin,/sbin,/boot</directories>
  <directories check_all="yes" report_changes="yes" realtime="yes"/>/home/vagrant</directories>

```

4 pav. Saugos įvykių nurodytose direktorijose agregacijos konfigūracija

Sistema taip pat atlieka didelį kiekį įvairaus tipo patikrų, kaip *CIS* ar pažeidžiamumų patikros.

Wazuh - Modules / wazuhAgent1 / Security configuration assessment

Inventory Events wazuhAgent1 (001)

< CIS benchmark for Ubuntu Linux 20.04 LTS > Export formatted Refresh

Pass	Fail	Not applicable	Score	End scan
68	103	20	39%	Dec 12, 2022 @ 22:06:52.000

Filter or search

ID ↑	Title	Target	Result
19000	Ensure mounting of cramfs filesystems is disabled.	Command: modprobe -n -v cramfs	Failed
19001	Ensure mounting of freevxfs filesystems is disabled.	Command: /sbin/modprobe -n -v freevxfs	Failed
19002	Ensure mounting of jffs2 filesystems is disabled.	Command: /sbin/modprobe -n -v jffs2	Failed

5 pav. CIS patikros rezultatai wazuhAgent1 agentui

## 2.3. Užduoties reikalavimų įgyvendinimas

Įdiegta Wazuh SIEM sistema nuskaity žurnalinius įrašus, generuoja saugumo įvykius ir pranešimus. Taip įgyvendinami pirmi trys užduoties reikalavimai. Penktasis reikalavimas gali būti įgyvendinamas naudojant *crontab* funkcionalumą serveryje.