

Grupp 1: Pilve keskkonnad

Pilve keskkondi on neli: public clouds, private clouds, hybrid clouds ja multiclouds.

Mis on neil sarnast?

Iga cloud võtab kokku, koondab ja jagab skaleeritavaid andmetöötlusressursse üle võrgu. Iga pilvetüüp võimaldab ka pilvandmetöötlust. Igasugustele pilvedele saab lisada ka virtualiseerimis- ja automatiseerimistarkvara, et saada lisavõimalusi või suurendada tõhusust.

Mis on neil erinevat?

Erinevused nende vahel olid kunagi ainult asukoha ja omandiõiguse järgi kergesti määratletavad, kuid tänapäeval on neid erinevusi veidi rohkem.

Public clouds

Public cloud-id on pilvekeskkonnad, mis on tavaliselt loodud IT-infrastruktuurist, mis ei kuulu lõppkasutajale. Mõned suurimad avaliku pilveteenuse pakkujad on Alibaba Cloud, Amazon Web Services (AWS), Google Cloud ja Microsoft Azure.

Traditsioonilised avalikud pilved töötasid alati väljaspool tööruume, kuid tänapäeva avalikud pilveteenuse pakkujad on hakanud pilveteenuseid pakkuma klientide kohapealsetes andmekeskustes. See on muutnud asukoha ja omandi eristamise aegunuks.

Kõik pilved muutuvad avalikeks pilvedeks, kui keskkonnad on jaotatud ja mitmele inimesele ümber jaotatud.

Private clouds

Private cloud-id on lõdvalt määratletud kui pilvekeskkonnad, mis on pühendatud ainult ühele lõppkasutajale või rühmale, kus keskkond töötab tavaliselt selle kasutaja või rühma tule müüri taga. Kõik pilved muutuvad privaatlilvedeks, kui nende aluseks olev IT-infrastruktuur on pühendatud ühele kliendile, kellel on täiesti isoleeritud juurdepääs.

Kuid private cloud-e ei pea enam hankima kohapealsest IT-infrastruktuurist. Organisatsioonid loovad praegu private cloud-e renditud, tarnijale kuuluvatele andmekeskustele, mis asuvad väljaspool äriruume, mistõttu kõik asukoha- ja omandireeglid on muudetud aegunuteks.

Hybrid clouds

Hybrid cloud on näiliselt üks IT-keskkond, mis on loodud mitmest keskkonnast, mis on ühendatud LAN-ide, WAN-ide, VPN-ide ja/või API-de kaudu. Iga IT-süsteem muutub hübriidpilveks, kui rakendused saavad liikuda mitmesse erinevasse, kuid ühendatud keskkonda, sisse ja välja.

Hybrid cloud-i omadused on keerulised ja võivad erineda olenevalt kellelt sa küsid. Näiteks võib hybrid cloud sisaldada järgmist:

1. Vähemalt üht private cloud ja vähemalt üht public cloud
2. Kaht või enam private cloud-i
3. Kaht või enam public cloud-i
4. Jne

Multiclouds

Multicloud-id on pilvepõhine lähenemine, mis koosneb enam kui ühest pilveteenusest rohkem kui ühelt pilveteenuse pakkujalt – avalikult või privaatsetl. Kõik hybrid cloud-id on multicloud-id, kuid mitte kõik multicloud-id pole hybrid cloud-id. Multicloud-id muutuvad hübriidpilvedeks, kui mitmed pilved on ühendatud mingi integratsiooni.

Multicloud keskkond võib eksisteerida sihilikult või kogemata. Mõlemal juhul on mitme pilve olemasolu muutumas üha tavalisemaks ettevõtetes, mis soovivad suurendada turvalisust ja jõudlust laiendatud keskkondade portfelli kaudu.

NB!

<https://github.com/NeptuneN/PythonREST>

Tahtsin näidata, et ma vähemalt **üritasin** ka teoreetilise ülesande kõrvalt teha seda praktilist osa, kus pidi mingi REST teenuse pilve keskkonda panema. Ma üritasin kasutada Pythonid, FaunaDB-d, ning oma keskkonnaks DigitalOceanit. Kahjuks, ma ei jõudnud kaugemale kui POST päringud, mis ei tahtnud läbi Command Line-l mul töödada kui kasuta "curl" käsklust. Mul tekib "NoneType error" POST-i käskluse käima laskmisel, ning seal mul sai ka mõistus otsa. Kui soovite, siis mul on mingisugune juhend kasutuseks **lokaalses keskkonnas, mitte pilves** läbi Windowsi, kuid mitte Linuxi.

Vaja läheb Python 3.x.x ja Pip-i.

1.pip install flask

2.pip install faunadb

3.flask run (IP aadress peaks olema 127.0.0.1:8080) **VÕI** python main.py (IP aadressi tuleb järgida Command Lineist)

4.Kasutaja tekitamine - POST päring Command Lineis:

```
curl -i -d '{"user":"string", "password": "string"}' -H 'Content-Type: application/json' -X POST http://127.0.0.1:8080/signup
```

5.Kasutaja autentimine – POST päring Command Lineis:

```
curl -i -d '{"user":"string", "password": "string"}' -H 'Content-Type: application/json' -X POST http://127.0.0.1:8080/login
```

GET, PATCH ja DELETE päringuteni ma ei jõudnud kahjuks.