

1 LABORATORINIS DARBAS

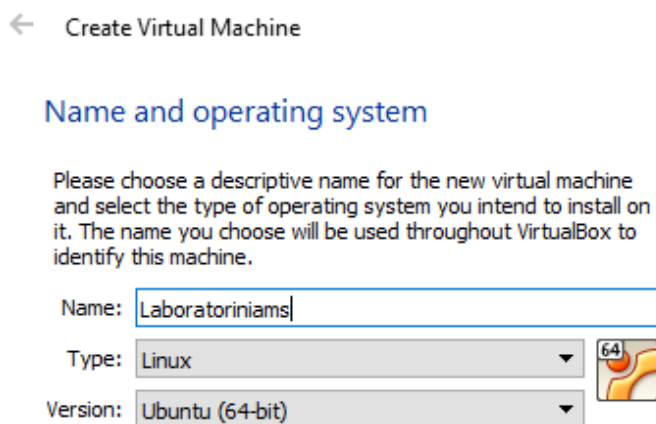
Atliko: Artūras Jonkus, EKSfm-16

1. Pasirinktos atviro kodo virtualios mašinos įrašymas

Pasirinkta virtualios mašinos programinė įranga – *Oracle VirtualBox*. Senesnės nei 4.0 programos versijos buvo aprašomos *VirtualBox Personal Use and Evaluation* (PUEL) licencija, vėlesnės aprašomos GPLv2 (*General Public License*, v2) licencija. Ši licencija leidžia neapribotai naudoti programinę įrangą ir jos kodą bei iš to naudojimo uždirbti, tačiau bet kokia nauja programa, kurioje panaudotas šia licencija saugomas kodas, negali būti aprašomi kitokio tipo licencija. Tokia licencija apsaugo, kad kodas nebus panaudotas uždaro kodo sistemose (komerciniuose projektuose), tuo pačiu kiekvienas galės prisidėti prie kodo tobulinimo. Tokia licencija padėjo augti Linux ir kitų atviro kodo operacinių sistemų populiarumui – jų branduoliui (angl. *kernel*) suteikiama būtent ši arba naujesnė, trečioji licencijos versija (GPLv3), turinti papildomų saugiklių, tačiau iš esmės suteikianti panašias apsaugas.

Pati programinė įranga parsisiunčiama iš <https://www.virtualbox.org/wiki/Downloads>. Čia pasirenkama, kokioje operacinėje sistemoje ši programa bus naudojama (mano atveju, *Windows*). Parsisiųsta programinė įranga įrašoma tiesiog sekant nurodymus ekrane.

Prieš įrašinėjant bet kokią operacinę sistemą reikia sukurti virtualią mašiną ir sukonfigūruoti jos parametrus. Tam reikia spausti *Machine->Add*. Atsivėrusiame lange (1.1 paveikslas) pasirinkti vardą (*Name*), tipą (*Type*) ir versiją (*Version*). Paspaudus *Next* bus pereinama prie kitų langų, kuriuose paeiliui reikia pasirinkti operatyvios atminties dydį (pasirinkta 2048 MB), toliau keletas langų skiriami kietojo disko aprašymui – šiuo atveju išskiriamas naujas virtualus diskas (VDI tipo), kurio dydis alokuojamas dinamiškai – padidėjus virtualioje mašinoje saugojamam duomenų kiekiui padidėja ir šio virtualaus disko dydis pagrindinėje sistemoje (*Windows OS*).



1.1 pav. Virtualios mašinos vardo, tipo ir versijos nustatymai

Aukščiau paminėtos konfigūracijos galimybės yra būtinos sukurti virtualią mašiną, prieš įrašinėjant operacinę sistemą joje taip pat galima spustelti *Settings* mygtuką ir ten susikonfigūruoti papildomus parametrus – įjungti vaizdo skaičiavimų spartinimą, apibūdinti USB įrenginius, sukurti tarp skirtingų operacinių sistemų pasiekiamus bendrus aplankus ir t.t.

2. Atviro kodo operacinės sistemos diegimas virtualioje mašinoje

Į įrašytą virtualią mašiną jau galima diegti operacinę sistemą. Pasirinkta operacinė sistema – *Ubuntu*, kurios branduolį iš esmės sudaro *Linux* branduolys (angl. *kernel*), kuris bene visas licencijuojamas jau minėtomis GPL tipo licencijomis. Grafinių sąsajų pasirinkimas nemažas, tačiau paprastai su *Ubuntu* siūloma GNOME grafinė sąsaja.

Du kartus paspaudus ant virtualios mašinos ikonos pradiniam puslapyje bus parodytas langas, jog nėra sistemos, kurią būtų galima užkrauti (*No bootable medium found*). Tai reiškia, kad nėra pasirinktas joks virtualus diskas, iš kurio būtų galima įrašyti operacinę sistemą (OS). Tam reikia pasirinkti *Devices->Optical Drives->Choose disk image* ir atsidariusiame lange surasti virtualaus disko failą, kurio plėtinys paprastai yra .iso. Vėliau, paspaudus *Machine->Reset*, virtuali mašina persikraus ir startuodama pradės operacinės sistemos diegimą. Diegimas pakankamai intuityvus, atsidarančiuose languose reikia nurodyti, kad OS įrašoma į diską ištrinant šio disko (virtualaus) duomenis, vėliau pasirinkti, ar norima tuo pačiu įdiegti atnaujinimus, taip pat nustatyti naudojimosi kalbą, laiko juostą bei vartotojo vardą ir slaptažodį. Pabaigus diegimą pakanka išimti virtualų diską iš virtualaus diskasukio, pasirenkant *Devices->Optical Drives->Remove disk from virtual drive*. Dabar šia operacine sistema galima naudotis pilnavertiškai – pradėti diegti vartotojui naudingas, tačiau dar neįdiegtas programas.

3. Lokalaus git serverio diegimas operacinėje sistemoje

Paskutinis laboratorinio darbo žingsnis – lokalaus *git* serverio diegimas. Git sistemos paprastai aprašomos LGPL (*Lesser Global Public License*) licencija, reiškiančia, kad taikomi GPL licencijoms galiojantys reikalavimai, tačiau tik panaudojamam duomenų blokui. Pavyzdžiui, jeigu tokia licencija pritaikyta bibliotekai, pavyzdžiui, komunikacijai su modemu, tai galutinė programa, kurioje panaudota ši biblioteka, gali turėti kitokią licenciją ir būti uždaro kodo, tačiau redagavus biblioteką pati biblioteka negalės pakeisti licencijos tipo. Kitaip sakant, atlikti pokyčiai turi būti prieinami atvirai.

Pirmiausia reikia įdiegti pačią *git* programinę įrangą. Tai galima padaryti terminalo lange parašius *sudo apt-get update && sudo apt-get install git-core*. Vėliau reikalinga sukurti atskirą vartotoją, kuris bus naudojamas kaip vietinis serveris. Tai gali būti jau esamas vartotojas, tačiau šiuo atveju sukuriamas naujas vartotojas *git*. Sukūrimui naudojama *sudo adduser git* komanda. Čia paprašoma parinkti slaptažodį, šiuo atveju jis – *git*, kaip ir vartotojo vardas. Visus kitus parametrus

galima praleisti. Prie sukurto vartotojo failų galima prieiti surinkus *sudo su* bei suvedus esamo vartotojo (to, kuris prisijungęs dabar) slaptažodį. Nuo šiol *sudo* vartoti nereikia ir netgi negalima, kadangi vartotojas *git* nėra įtrauktas į *sudoers* grupę, todėl neturi leidimų naudotis šios funkcijos privilegijomis.

Po minėtų veiksmų namų aplanke */home/git/* sukuriamas aplankas *Projektai*. Tam reikalinga *mkdir Projektai*. Tam, kad šiame aplanke būtų galima inicijuoti *git* direktoriją, naudojama *cd Projektai && git --bare init*. Tai visi reikalingi žingsniai iš šio vartotojo pusės. Grįžti atgal galima su *exit && sudo su arturas* (*arturas* - studento vardas).

Tam, kad būtų galima patikrinti, ar galima skelbti failus, reikia kažkur sukurti aplanką ir jame sukurti bent vieną failą. Tai atlieka *mkdir [projekto aplanko vieta] && cd [projekto aplanko vieta] && touch [failo vardas]* (skliaustuose nurodytos vertės priklauso nuo vartotojo pasirinkimo). Pavyzdžiui, *mkdir Projektas1 && cd Projektas1 && touch README*. Atsiradusiame aplanke taip reikia sukurti *git* aplanką, su *git init* komanda. Sukurtas *README* failas gali būti pridėtas *git add README* komanda. Pirmą kartą kreipiantis sistema paprastai paprašo autentifikavimo teikiant failus versijavimui, todėl prieš tai reikėtų nurodyti elektroninį paštą ir vartotojo vardą. Jiems aprašyti atitinkamai reikalingos *git config --global user.email [vartotojo elektroninis paštas]* ir *git config --global user.name [vartotojo vardas]* komandos. Po šių komandų galima teikti failus versijavimui su *git commit -m "[pageidaujama žinutė]"*.

Failas pateiktas versijavimui, tačiau dar neatiduotas lokaliai serveriui. Apskritai, tarp dviejų sukurtų aplankų dar kol kas nėra ryšio, jį reikia sukurti. Tam reikalinga *ssh* (*secure shell*) komunikaciją inicijuojanti programa *OpenSSH* (įrašoma su *sudo apt-get update && sudo apt-get install openssh-server*), po šios komandos sėkmingo įvykdymo reikėtų kviesti *git remote add origin git@localhost:Projektai1*, taip nurodant, kad pastarasis aplankas bus vieta, kur serveryje bus saugomi pakeitimai. Galiausiai pakeitimus pateikti serveriui galima su *git push origin master* bei sutinkant su visais pasirodančiais pasirinkimais. Suvedus *git status* turėtų pasirodyti užrašas, kad visi pakeitimai buvo atlikti ir duomenys yra sinchronizuoti (*nothing to commit, working directory is clean*). Tai reiškia, kad vietinis serveris yra sukurtas ir veikia tinkamai.