KAUNO TECHNOLOGIJOS UNIVERSITETAS

INFORMATIKOS FAKULTETAS

KOMPIUTERIŲ KATEDRA

Mantas Žirgulis

Virtualios aplinkos saugos sistemos prototipas

Magistro darbas

Darbo vadovas

doc. dr. J. Toldinas

Kaunas, 2012

KAUNO TECHNOLOGIJOS UNIVERSITETAS

INFORMATIKOS FAKULTETAS

KOMPIUTERIŲ KATEDRA

Mantas Žirgulis

Virtualios aplinkos saugos sistemos prototipas

Magistro darbas

Recenzentas
doc. dr. G. Ziberkas
2012-05-

Vadovas
doc. dr. J. Toldinas
2012-05-

Atliko
IFN-0/3 gr. stud.
Mantas Žirgulis
2012-05-24

Kaunas, 2012

TURINYS

`

ĮVADAS

Idėja paleisti keletą operacinių sistemų viename kompiuteryje ar serveryje nėra nauja. Ši mintis pristatyta dar 1960 metais, tačiau tuo metu didelio susidomėjimo nesukėlė. Virtualizacija suteikia galimybę veikti kelioms operacinės sistemos viename kompiuteryje, taip efektyviai išnaudojant kompiuterio resursus ir techninę įrangą, todėl, didėjant kompiuterių galingumui, ši technologija įgavo naują pagreitį [1].

Virtualizacija daro didžiulę įtaką šiandienos IT pasauliui. Tai technologija, kuri suskirsto fizinį kompiuterį į keletą iš dalies arba visiškai izoliuotų mašinų, vadinamų virtualiosiomis mašinomis (svečio mašinomis). Virtualios mašinos naudoja pagrindinio (angl. host) kompiuterio resursus, tačiau kiekviena turi savo atskirą operacinę sistemą ir programinę įrangą. Tai sudaro iliuziją, tarsi virtualių mašinų procesai vyksta fiziniame kompiuteryje, bet iš tikrųjų jos dalijasi pagrindinės mašinos fizine įranga. Programinė įranga, kuri leidžia svečio operacinėms sistemoms naudoti fizinio kompiuterio aparatinę įrangą, vadinama hipervizoriumi arba valdymo programa. Hipervizorius diegiamas tarp pagrindinės mašinos operacinės sistemos ir virtualios aplinkos [9].

Ne virtualioje aplinkoje veikiančios programos gali matyti viena kitą, ir, kai kuriais atvejais, netgi komunikuoti viena su kita. Virtualioje aplinkoje programos, veikiančios vienoje svečio mašinoje, yra izoliuotos nuo programų, veikiančių kitoje svečių mašinoje. Izoliacijos laipsnis turėtų būti pakankamai didelis, kad vienos virtualios mašinos pažeidžiamumai neturėtų įtakos nei kitai virtualiai mašinai, nei pagrindinei operacinei sistemai.

Saugumo atžvilgiu virtualizuojamas kompiuteris niekuo nesiskiria nuo kompiuterio, kuris nėra virtualizuojamas. Virtuali aplinka yra pažeidžiama visų tradicinių atakų, kurios paplitusios įprastoje aplinkoje. Tačiau virtualizuotos aplinkos atveju, kai pagrindinėje operacinėje sistemoje veikia keletas virtualių mašinų, situacija yra sudėtingesnė. Galimų saugumo spragų yra daugiau, nes virtualioje aplinkoje reikia apsaugoti daugiau sistemų. Šioje aplinkoje yra daugiau galimų įėjimo taškų, daugiau operacinių sistemų (kurioms nuolat reikia diegti naujinimus), daugiau sujungimo taškų. Kenkėjai ir programišiai aktyviai kuria naujas kenkėjiškas programas virtualiai aplinkai. Viena žinomiausių atakų yra „rootkit“ injekcija. Tai ataka, kurios metu „rootkit“ tipo kenkėjiška programa slapta įsibrauna į operacinę sistemą ar jos branduolį ir perima kompiuterio ar serverio valdymą [5]. Ši ataka yra tik viena iš daugelio galimų išpuolių prieš virtualią aplinką.

`

1. VIRTUALIOS APLINKOS PAŽEIDŽIAMUMAI

Virtualizacija - tai technologija, leidžianti veikti kelioms operacinėms sistemoms tuo pat metu tame pačiame kompiuteryje. Ji pristatyta 1960-ųjų viduryje kartu su IBM tarnybinėmis stotis (angl. mainframes). Ilgą laiką nepopuliari virtualizacija išgyveno renesansą 1990-ųjų pabaigoje kartu su Disco ir VMware komercine sėkme. Su aparatinės įrangos programomis, skirtomis palaikyti pilną virtualizaciją šiuolaikiniuose x86 procesoriuose, pasirodė ir naujos virtualios aplinkos [6]. Tipiškoje virtualizacijoje pridedamas programinės įrangos abstraktusis sluoksnis - hipervizorius, kuris įterpiamas tarp aparatinės įrangos ir pagrindinės operacinės sistemos. Funkcionuodamas tarp virtualių prietaisų ir platformos fizinių įrenginių, virtualizacijos sluoksnis palengvina dalijimąsi ištekliais ir atsieja svečio operacines sistemas nuo aparatinės įrangos.

1.1. Virtualių aplinkų klasifikavimas

Virtualizacijos esmė

Dažniausiai kompiuterinė aplinka nepakankamai išnaudoja fizinių serverių gebą. Virtualizacijos technologija atveria nepanaudotus pajėgumus ir leidžia maksimalizuoti procesorių, atmintį, diską ir valdiklius kiekviename fiziniame įrenginyje. Užuot pirkus brangius serverius neišnaudojant jų pajėgumų, galima sukurti naują virtualią mašiną ir taip išvengti tuščiosios eigos ir serverių priežiūros išlaidų. Toliau pateikiami pagrindiniai virtualizacijos privalumai [3]:

- Resursų naudojimas. Virtualių serverių pagalba galima maksimaliai išnaudoti fizinius įrenginius, todėl atsiperka lėšos, išleistos aparatinei įrangai.
- Valdymas. Galima automatizuoti resursų valdymą. VM gali būti sukurtos ir sukonfigūruotos automatiškai. Visas VM, esančias virtualiame serveryje, galima centralizuotai valdyti pagrindinės operacinės sistemos pagalba.
- Apjungimas. Apjungta skirtinga programinė įranga gali veikti mažesniame fizinių komponentų kiekyje. Galima apjungti tiek serverius ir saugyklas, tiek ištisas sistemas, duomenis, duomenų bazes, tinklus bei asmeninius kompiuterius. Apjungimas sudaro sąlygas efektyvesniam veikimui ir taip sumažina kaštus, kurie skirti resursų įsigijimui.
- Energijos sunaudojimas. Didelių duomenų centrų aprūpinimas elektra tapo pakankamai sudėtingas. Serverių darbui reikalinga elektros energija kainuoja daugiau nei pačių serverių įsigijimas, atsižvelgiant į jų gyvavimo laiką. Resursų apjungimas reikalauja mažiau aparatinės įrangos tai pačiai užduočiai atlikti, taigi tuo pačiu reikia ir mažiau elektros energijos.

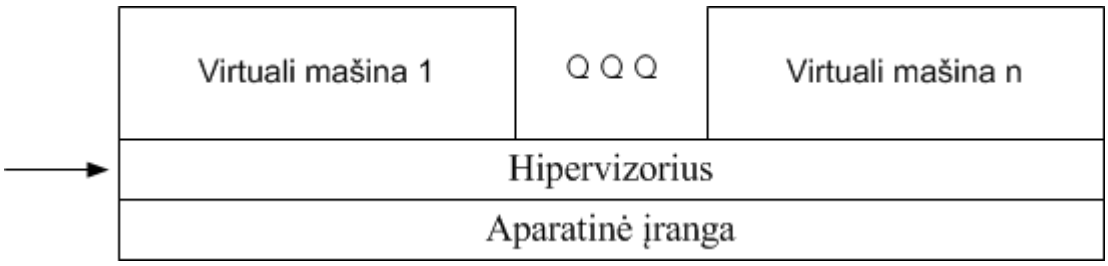
`

- Reikia mažiau vietos. Yra keliami tam tikri saugumo reikalavimai duomenų centrų patalpoms, dėl kurių jų įrengimas ir išlaikymas yra labai brangūs. Pasinaudojus virtualizacija atsiranda galimybė tą patį darbą atlikti su mažesniu aparatinės įrangos kiekiu. Tokiu būdu yra sutaupoma pinigų ne tik dėl mažesnio aparatinės įrangos poreikio, bet ir dėl mažesnio patalpų poreikio.
- Gyvas perkėlimas. Tradicinės sistemos yra pririštos prie serverio ar darbastalio aparatinės įrangos, o serverių gyvavimo trukmė paprastai yra nuo trejų iki penkerių metų, todėl pasenus serverio techninei įrangai duomenys turi būti perkelti į naują platformą, o programos turi būti pertvarkomos naujoje aplinkoje. Virtualioje aplinkoje perkėlimas nesudaro jokių problemų.

Hipervizoriaus tipai

Virtualioje aplinkoje labiausiai privilegijuotas komponentas vadinamas hipervizoriumi. Hipervizoriaus funkcionalumas yra panašus į OS branduolį - atsiribojama nuo pagrindinės aparatinės įrangos platformos ir izoliuojami virš jo veikiantys komponentai. Taip sudaroma iliuzija, kad virtuali mašina veikia tarsi atskiras realus kompiuteris. Virtualizacijoje hipervizoriai skirstomi į dvi grupes – I tipo ir II tipo [2]. 1 ir 2 paveikslai iliustruoja kiekvieną hipervizoriaus tipą. Yra didelių skirtumų tarp I ir II tipų hipervizorių ir kiekvienas iš jų turi konkrečius panaudojimo atvejus.

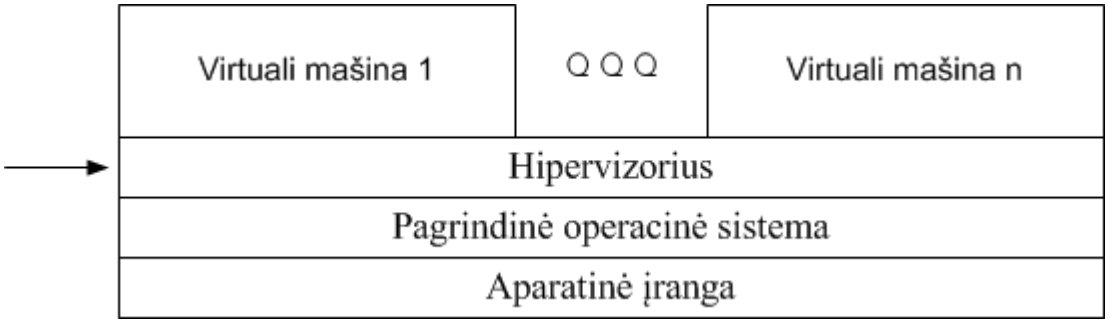
I tipo gimtasis (angl. native, bare-metal) hipervizorius diegiamas tiesiai virš aparatinės įrangos (1 pav.). Šis modelis yra klasikinė virtualių mašinų architektūros realizacija, nes hipervizorius naudoja aparatūrinę įrangą be tarpininkų. I tipo hipervizoriaus konfigūracija labiausiai paplitusi serverių virtualizacijoje.



Pav. 1 I tipo gimtasis Hipervizorius

II tipo (angl. hosted) hipervizorius dirba virš pagrindinės operacinės sistemos (2 pav.). Šio tipo hipervizorius veikia standartinėje operacinėje sistemoje, kaip ir kitos standartinės programos. Pagrindine operacine sistema gali būti bet kuri įprasta OS - Windows, Linux ar kt. II tipo hipervizoriaus konfigūracija labiausiai paplitusi tarnybinių stočių virtualizacijoje.

`



Pav. 2 II tipo Hipervizorius

I tipo hipervizorius valdo virtualių mašinų operacijas ir turi tiesioginę sąsaja su pagrindine aparatine įranga, todėl garantuoja greitesnį veikimą bei didesnį stabilumą. Šio tipo hipervizorius gerai tinka dideliems duomenų centrams, nes paprastai diegiamas su papildomomis funkcijomis, tokiomis kaip: išteklių valdymas, lankstus priėjimas bei padidintas saugumas. Administratoriai gali centralizuotai valdyti šio tipo hipervizorių, o tai yra labai svarbu esant dideliam skaičiui virtualių mašinų. Populiariausi I tipo hipervizoriai yra: VMware ESX ir ESXi, Microsoft Hyper-V, Citrix Systems XenServer.

II tipo hipervizorius valdo virtualias mašinas ir turi sąsają su pagrindine operacine sistema, kuri padeda turėti prieigą prie aparatinės įrangos. Šio tipo hipervizorius geriau suderinamas su aparatine įranga, nes pagrindinė operacinė sistema yra atsakinga už įrenginių tvarkykles, o ne hipervizorius. Šio tipo hipervizorius dažniausiai naudojamas darbastalių virtualizacijoje, kur suteikiama galimybė įdiegti kelias operacines sistemas. Populiariausi II tipo hipervizoriai yra: VMware Workstation, Server, Player ir Fusion, Oracle VirtualBox, Microsoft Virtual PC, Parallels Desktop.

1.1.1. Hipervizoriaus programinės įrangos pažeidžiamumai

Hipervizorius yra naujas sluoksnis tarp pagrindinės OS ir virtualios aplinkos, todėl atsiranda naujų galimybių kenkėjiškoms atakoms. Be to, pats hipevizorius irgi yra programa, todėl, kaip ir visai programinei įrangai, jam būdingos visos tradicinės programinės įrangos klaidos ir saugumo pažeidžiamumai [7][14].

Virtualizuotoje aplinkoje svečio operacinė sistema veikia kaip tradicinė operacinė sistema, kuri valdo įvestį, išvestį bei tinklo srautus, nepaisant to, kad ji yra kontroliuojama hipervizoriaus. Tuo tarpu hipervizorius turi aukštą sistemos kontrolės lygį ne tik virtualioje mašinoje, bet ir pagrindinėje operacinėje sistemoje. Specializuota virtualizacijai, o ypatingai hipervizoriui, kenkėjiška „rootkit“ tipo programinė įranga yra labai populiari įsilaužėlių tarpe. „Rootkit“ tipo piktavališka programinė įranga yra maža programa, kuri slapta įsibrauna į operacinę sistemą ar hipervizorių ir perima kompiuterio ar tarnybinės stoties valdymą. Ši

`

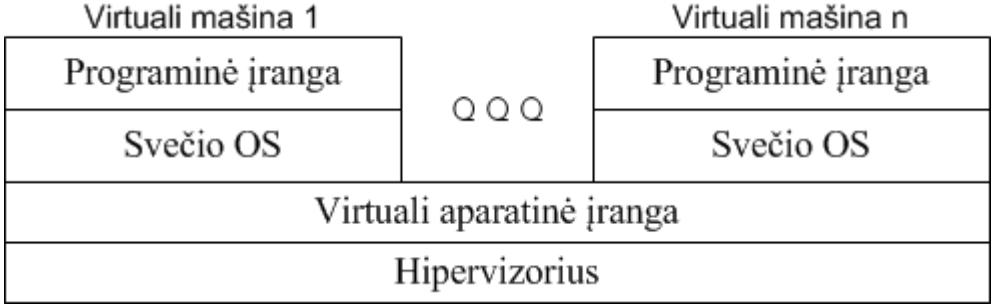
piktavališka programa dengia daug paslėptos veiklos – paslepia kenkėjiškus operacinės sistemos procesus, suteikia administratoriaus arba sistemos privilegijas ir kitaip išnaudoja hipervizoriaus pažeidžiamumus [3].

Dėl pagrindinės operacinės sistemos privilegijuotos pozicijos virtualios mašinos atžvilgiu, netinkamai sukonfigūruota VM gali leisti programinei įrangai visiškai apeiti virtualią aplinką bei administratoriaus teisėmis gauti pilną priėjimą prie fizinės mašinos. To pasekmė būtų „apeitas“ hipervizorius ir sugriautas virtualios aplinkos saugumo mechanizmas. Blogai apsaugotas ir sukonfigūruotas hipervizorius yra didelė, visos virtualios aplinkos, saugumo spraga. Nuo išorės įtakos apsaugota virtuali mašina gali būti tiesiogiai modifikuota hipervizoriaus pagalba. Todėl hipervizoriaus saugumui turi būti skiriamas ypač didelis dėmesys.

Virtualizacijos tipai

Bandydamos spręsti su kompiuteriais susijusias problemas, organizacijos pasirenka įvairių formų virtualizaciją. Virtualizacijos tipai yra skirtingi įvairiose tam skirtose platformose. Dažniausiai naudojamos virtualizacijos formos - aparatinės įrangos, operacinės sistemos bei sparčiai plintanti aplikacijų virtualizacija [11].

Virtualizuojant aparatinę įrangą (3 pav.) hipervizorius sukuria virtualios aparatinės įrangos sluoksnį, kurį svečio operacinė sistema mato kaip savo aparatinę įrangą. Svečio operacinė sistema sąveikauja su virtualia įranga, kuri veikia kaip fizinio kompiuterio įranga. Pagal virtualizuotos aparatinės įrangos nustatymus svečio operacinė sistema įdiegiama į virtualią mašiną. Po įdiegimo svečio OS virtualioje mašinoje veikia tarsi atskirame kompiuteryje, o vartotojas gali įdiegti ir paleisti programas svečio OS taip pat kaip ir fiziniame kompiuteryje.

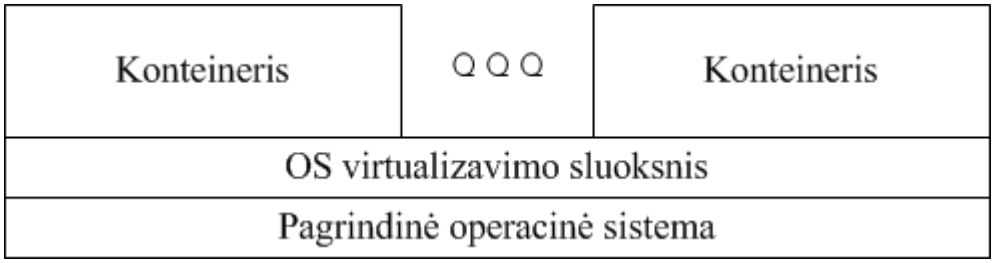


Pav. 3 Aparatinės įrangos virtualizacija

Operacinės sistemos virtualizacijoje (4 pav.) kuriami konteineriai, kurie naudojasi pagrindine aparatine įranga per hipervizorių ir pagrindinę operacinę sistemą. OS konteineris nėra visiškas operacinės sistemos analogas. Hipervizorius yra atsakingas už pagrindinės sistemos išteklių, reikalingų konteineriams, valdymą. Tokio tipo virtualią mašiną sukuria

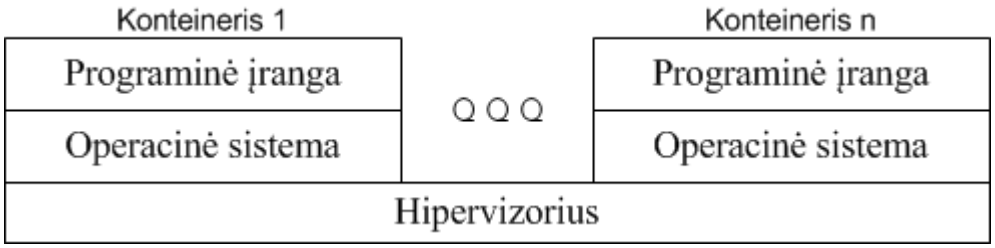
`

Parallels Virtuozzo technologija. Šio tipo virtualizacija yra naudingiausia tuomet, kai organizacijos nori atskirti skirtingas serverio programas, naudojančias to paties tipo operacinę sistemą virtualiose mašinose, kad būtų galima izoliuoti gedimus ir palaikyti saugumą.



Pav. 4 Operacinės sistemos virtualizavimas

Aplikacijų virtualizacija (5 pav.) uždaro aplikacijas į konteinerį kartu su sistemos failais, tiesogiai susijusiais su aplikacija. Ši virtualizacijos forma leidžia naudoti programas daugelyje kompiuterių ir izoliuoja vienos programos sisteminius nustatymus nuo kitos programos nustatymų.

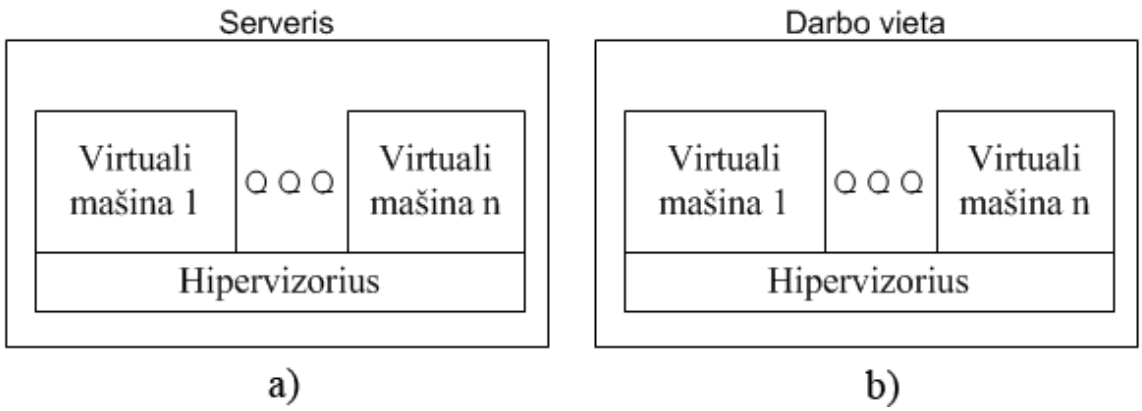


Pav. 5 Taikomųjų programų virtualizacija

Platformų virtualizavimas

Be hipervizoriaus ir virtualizacijos tipo pasirinkimo, svarbu teisingai pasirinti ir virtualizuojamo kompiuterio tipą. Šiandien organizacijoms būdingiausia virtualizuoti stacionarus kompiuterius, siekdamos sumažinti tiesioginių ir netiesioginių išlaidų sumą TCO (angl. total cost of ownership), supaprastinti stalinių kompiuterių valdymą ir padaryti darbastalio aplinką prieinamą iš bet kurios vietos ir bet kuriuo metu. Kaip teigia J. Robertsas ir J. Yaconą [8], 85% informacinių technologijų sprendimų tiekėjų jau 2003 metais rekomendavo ar planavo rekomenduoti serverio virtualizaciją savo klientams. Serverio virtualizacija (6a pav.) suteikia galimybę sujungti kelis serverius naudojant galingas ir naujesnes serverių technologijas. Tokiu būdu organizacijos gali sumažinti fizinių serverių kiekį, taip sumažinant erdvės poreikius ir energijos suvartojimą.

`



Pav. 6 a) serverio virtualizacija b) darbo vietos virtualizacija

Be to, virtualizuojant serverį, hipervizorius izoliuoja kiekvieno serverio virtualią mašiną, taip užkirsdamas kelią virtualioms mašinoms valdyti vienai kitos konfigūracijas, procesus ar kitas charakteristikas. „TechTarget“ atliktas tyrimas parodė, jog 2011 metais didžiausi serverių virtualizacijos rinkos lyderiai buvo VMware ir Microsoft [26]. Serverių virtualizacijos pradininkų VMware siūlomą platformą vSphere rinkosi 76% organizacijų, o Microsoft Hyper-V ir jo pradininką Virtual Server rinkosi 13% organizacijų. Tyrimas rodo, jog kitos serverių virtualizacijai skirtos platformos buvo ne tokios populiarios (pvz. Citrix ir jų paltforma XenServer, Red Hat siūloma platforma KVM bei Oracle, kurie remiasi atviro kodo XEN hypervizoriumi).

Nors darbo vietų virtualizacija panaši į serverio virtualizaciją, darbo vietų virtualizacija leidžia virtualizuoti darbastalio operacines sistemas. Skiriami du pagrindiniai darbo vietų virtualizacijos variantai: kompiuterizuotų darbo vietų virtualizacija (6b pav.) ir darbo vietų virtualizacija, naudojant virtualią darbo vietų infrastruktūrą VDI (Virtual Desktop Infrastructure).

Darbo vietų virtualizacijos atveju, platforma leidžia emuliuoti vieną ar daugiau virtualių mašinų kompiuterizuotoje darbo vietoje. Virtuali mašina gali turėti prieigą prie kompiuterizuotos darbo vietos išteklių per hipervizorių. Kompiuterizuotą darbo vietą galima virtualizuoti naudojant įvairias virtualizacijos platformas - VMware Workstation, Microsoft Virtual PC, Parallel Workstations ir kt.

1.1.2. Pagrindinio kompiuterio programinės įrangos pažeidžiamumai

Virtualizuotos sistemos saugumo spragos yra beveik tokios pačios kaip ir ne virtualizuotos, tik prisideda dar ir atskirtos tik virtualiai sistemai galimos rizikos bei kenkėjiškos atakos [12][10]. Jeigu pažeidžiama bent viena virtuali mašina, saugumo pažeidžiamumas gali būti išnaudotas atakuojant ir kitas VM ar pagrindinę operacinę sistemą. Žemiau paminėta keletas bendrųjų spragų, kuriomis pasižymi virtuali aplinka.

`

Ryšys tarp virtualių mašinų ir pagrindinės operacinės sistemos

Vienas pagrindinių virtualizacijos privalumų - atskyrimas. Šis privalumas gali tapti grėsme šiai aplinkai, jei nėra tinkamai naudojamas. Atskyrimas virtualioje aplinkoje turi būti atidžiai konfigūruojamas ir palaikomas, užtikrinant, kad vienoje virtualioje mašinoje veikiančios programos neturės prieigos prie kitoje virtualioje mašinoje veikiančių programų. Atskyrimas turi būti pakankamai tvirtas, kad įsilaužus į vieną virtualią mašiną nepavyktų pasiekti kitų virtualių mašinų ar pagrindinės operacinės sistemos.

Virtualių mašinų bendrai naudojama mainų sritis yra naudinga funkcija, kuri leidžia persiųsti duomenis tarp virtualių mašinų ir pagrindinės operacinės sistemos. Tačiau minėtąją funkciją taip pat galima traktuoti ir kaip šliuzą persiųsti duomenims tarp bendradarbiaujančių kenkėjiškų programų virtualiose mašinose. Kenkėjiškais tikslais ši mainų sritis naudojama duomenims iš pagrindinės operacinės sistemos pavogti arba į ją slapta įkelti [7].

Kai kurios virtualizacijos sistemos vengia atskyrimo, kad vienai operacinei sistemai sukurtos programos galėtų veikti kitoje operacinėje sistemoje, tačiau šis sprendimas visiškai sunaikina saugos barjerus abiejose operacinėse sistemose. Tokioje sistemoje, kurioje pagrindinė operacinė sistema nėra atskirta nuo virtualių mašinų, VM turi neribotą prieigą prie pagrindinio kompiuterio išteklių, tokių kaip failų sistema ar tinklo įrenginiai. Taip pagrindinės operacinės sistemos failų sistema yra pažeidžiama.

Virtualių mašinų išsilaisvinimas

Virtualios mašinos yra taip sukurtos, kad vienoje virtualioje mašinoje veikianti programa negalėtų stebėti ar bendrauti su kitose virtualiose mašinose ar pagrindinėje operacinėje sistemoje veikiančiomis programomis. Realybėje organizacijos sugriovus atskyrimą lankstus konfigūravimas, tenkinantis organizacijos poreikius, padaro sistemas pažeidžiamas [4].

Atskyrimo atakos gali būti virtualios mašinos išsilaisvinimas. Ši ataka yra vienas blogiausių scenarijų, galinčių įvykti, jeigu pažeidžiamas atskyrimą tarp pagrindinės operacinės sistemos ir virtualių mašinų. Virtualios mašinos išsilaisvinimo atveju VM viduje veikianti programa gali visiškai apeiti virtualųjį hipervizoriaus sluoksnį ir gauti prieigą prie pagrindinės operacinės sistemos. Kadangi pagrindinė OS dirba administratoriaus teisėmis, virtuali mašina, įgijusi prieigą prie pagrindinės operacinės sistemos, tuo pačiu įgauna administratoriaus teises ir praktiškai išsilaisvina iš virtualiai mašinai taikomų teisių apribojimo. Šios atakos rezultate virtualios aplinkos saugos sistema visiškai sugriaunama.

`

Virtualių mašinų stebėjimas iš pagrindinės operacinės sistemos

Pagrindinė operacinė sistema virtualioje aplinkoje yra valdymo priemonė, kuri suteikia galimybę stebėti bei valdyti virtualias mašinas. Dėl šios priežasties būtina daug labiau saugoti pagrindinę operacine sistemą nei atskiras virtualias mašinas. Skirtingos virtualizacijos technologijos suteikia skirtingas priemones veikiančioms virtualioms mašinoms valdyti.

Galimi būdai, kaip pagrindinė operacinė sistema gali daryti įtaką virtualioms mašinoms:

- Pagrindinė operacinė sistema gali paleisti, išjungti, sustabdyti ir paleisti iš naujo virtualias mašinas.
- Pagrindinė operacinė sistema gali stebėti ir keisti virtualioms mašinoms pasiekiamus išteklius.
- Suteikus pakankamai teisių, pagrindinė operacinė sistema gali stebėti virtualiose mašinose veikiančias programas.
- Pagrindinė operacinė sistema gali peržiūrėti, kopijuoti ir keisti virtualioms mašinoms priskirtus virtualius diskus.

Visas tinklo srautas į ir iš virtualių mašinų keliauja per pagrindinę operacinę sistemą, todėl ši gali stebėti visų savo virtualių mašinų tinklo srautus. Pažeidus pagrindinės operacinės sistemos saugumą, visų virtualių mašinų saugumas tampa nieko vertas.

Virtualios mašinos stebėjimas iš kitos virtualios mašinos

Jeigu viena virtuali mašina gali be trukdžių stebėti kitos virtualios mašinos išteklius - tai laikoma saugumo grėsme. Naujieji procesoriai turi integruotą atminties apsaugos funkciją, kuria naudojasi už atminties atskyrimą atsakingas hipervizorius. Ši funkcija neleidžia vienai virtualiai mašinai matyti kitos virtualios mašinos atminties išteklių.

Kalbant apie tinklo srautą, atskyrimas visiškai priklauso nuo virtualios aplinkos tinklo sąrankos. Jei pagrindinis kompiuteris su svečio kompiuteriu yra sujungtas per dedikuotą fizinį kanalą, mažai tikėtina, kad svečio kompiuteris galėtų šnipinėti pagrindiniam kompiuteriui siunčiamus paketus ir atvirkščiai. Tačiau realybėje virtualios mašinos prie pagrindinio kompiuterio jungiamos per virtualų koncentratorių arba virtualų komutatorių. Taip svečio kompiuteris gali šnipinėti tinklu siunčiamus paketus arba, dar blogiau, „užnuodyti“ ARP (angl. address resolution protocol) ir nukreipti kito svečio kompiuterio siunčiamus arba priimamus paketus.

Aptarnavimo perkrovos ataka

Virtualių mašinų architektūroje svečio kompiuteris ir pagrindinis kompiuteris bendrai naudoja fizinius išteklius, tokius kaip CPU, atminties diskas ir tinklo ištekliai. Todėl svečias

`

gali įgyvendinti aptarnavimo perkrovos ataką prieš kitus svečio kompiuterius toje pačioje sistemoje. Aptarnavimo perkrovos ataką virtualioje aplinkoje galima apibūdinti kaip ataką, kurios metu svečias užvaldo visus galimus sistemos išteklius. Tokiu būdu sistema atsisako aptarnauti kitų svečių siunčiamas išteklių užklausas, nes joms nebelieka išteklių.

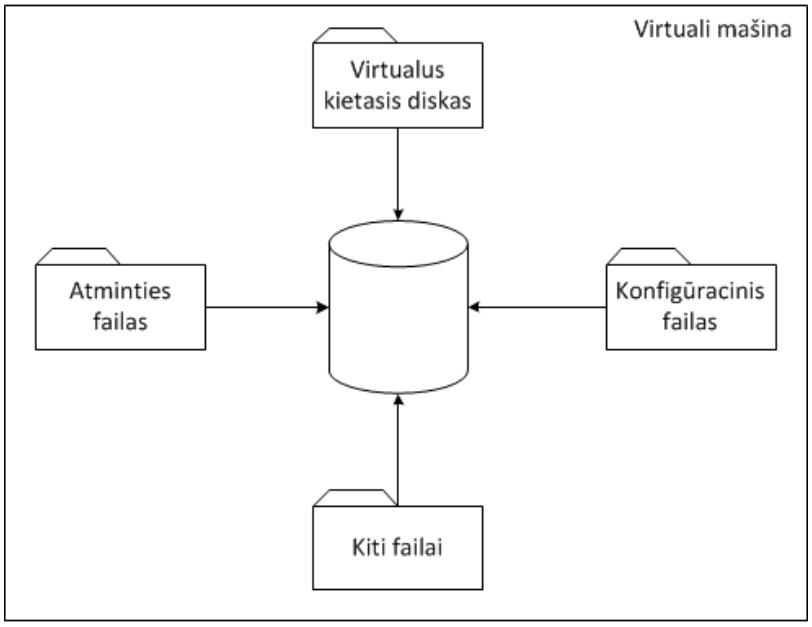
Geriausias būdas neleisti svečiui naudoti visų išteklių - riboti skiriamus išteklius. Dėl šios priežasties virtualizacijos technologija turi būti tinkamai konfigūruota, neleidžiant vienam svečiui sunaudoti visų galimų išteklių. Taip užkertamas kelias aptarnavimo perkrovos atakai.

Svečio ataka prieš svečią

Kaip jau buvo minėta, svarbiau apsaugoti pagrindinį kompiuterį nei virtualias mašinas. Jei užpuolikas įgyja administratorius teises pagrindinėje operacinėje sistemoje, tikėtina, kad jis galės įsilaužti ir į visas virtualias mašinas. Tai vadinama svečio ataka prieš svečią, nes užpuolikas gali peršokti iš vienos virtualios mašinos į kitą, jei pamatinė saugos sistema yra sunaikinta.

1.2. Virtualių mašinų failų struktūra ir jų saugos problemos

Virtuali mašina – tai programa, kurioje, lyg fiziniame kompiuteryje, instaliuojama operacinė sistema bei programinė įranga. Fiziškai virtuali mašina yra tik failų rinkinys, kurį sudaro konfigūracinis failas, virtualus kietojo disko failas, atminties failas, tarnybinių įrašų failas ir kt. Ši struktūra pavaizduota 7 paveiksle.



Pav. 7 Virtualios mašinos failo struktūra

`

TechTarget“ atliktas tyrimas parodė, jog 2011 metais populiariausios serverių virtualizacijos platformos buvo vSphere ir Hyper-V. 1 lentelėje pateikiami svarbiausi šių platformų failų plėtiniai [25].

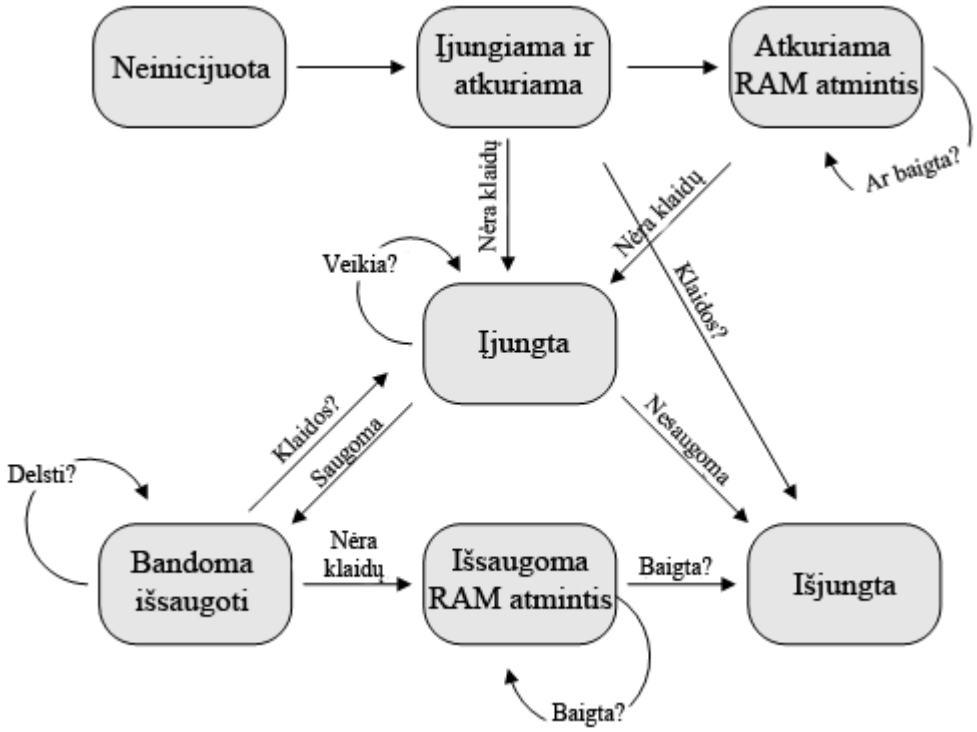
Lentelė 1. Populiariausių platformų plėtiniai.

Failo plėtinys		**Paskirtis**	**Aprašymas**
Hyper-V	**vSphere**		
.vhd	.vmdk	virtualus kietasis diskas	Virtualiame kietajame diske diegiama svečio operacinė sistema bei saugomi virtualios mašinos duomenys. Naudojami trys virtualaus kietojo disko tipai: fiksuotas, dinaminis ir diferencijuojamas. Fiksuoto disko dydis nurodomas VM kūrimo metu ir pagrindinėje operacinėje sistemoje iš karto užima nurodytą dydį. Dinaminis diskas plečiasi kartu su virtualios mašinos duomenimis. Jis užima tiek vietos, kiek užima VM duomenys. Diferencijuojamas diskas gali būti siejamas tiek su fiksuotu, tiek su dinaminiu disku.
.xml	.vmx	konfigūracinis failas	Kiekviena virtuali mašina privalo turėti vieną ir tik vieną konfigūracinį failą. Šiame faile saugomi visi virtualios mašinos konfigūraciniai nustatymai ir techninės įrangos parametrai.
.vsv	.vmem	atminties failas	Šis failas yra naudojamas tuomet, kai virtuali mašina yra išjungiama. Jis suteikia galimybe darbą tęsti toje pačioje situacijoje. Atminties failo dydis būna tokio paties dydžio kaip RAM kiekis, kurį tuo metu naudoja VM.

1.2.1. Virtualių mašinų būsenos

Virtuali mašina kiekvienu diskretaus laiko momentu yra vienoje būsenoje, o su kiekvienu nauju laiko momentu pereina į kitą būseną bei turi baigtinį būsenų skaičių, todėl galime teigti, kad virtuali mašina yra baigtinis automatas. Paveiksle 8 pateikta virtualios mašinos būsenų diagrama.

`



Pav. 8 virtualios mašinos būsenų diagrama

Formaliai virtualią mašiną VM galime išreikšti priklausomybe

VM=(X, Z, S, δ, λ), kur
X={x1,x2,…,xn} – baigtinis įėjimų alfabetas.
Z={z1,z2,…,zm – baigtinis išėjimų alfabetas.
S={s1,s2,…,sk} – baigtinė būsenų aibė.

δ ir λ – charakteringosios funkcijos. $s_{t+1} = \delta$ (xt, st), $z_t = \lambda$ (x_t, s_t), kur x_t, z_t , s_t – (atitinkamai) įėjimo simbolis, išėjimo simbolis ir automato būsena laiko momentu t.

Virtuali mašina yra determinuotas baigtinis automatas, kadangi VM charakteringosios funkcijos δ ir λ yra apibrėžtos visoms būsenų ir įėjimo signalų kombinacijoms, bei nėra jokių ribojimų įėjime esantiems signalams, t.y. bet kuriuo momentu gali būti paduotas bet kuris įėjimo signalas.

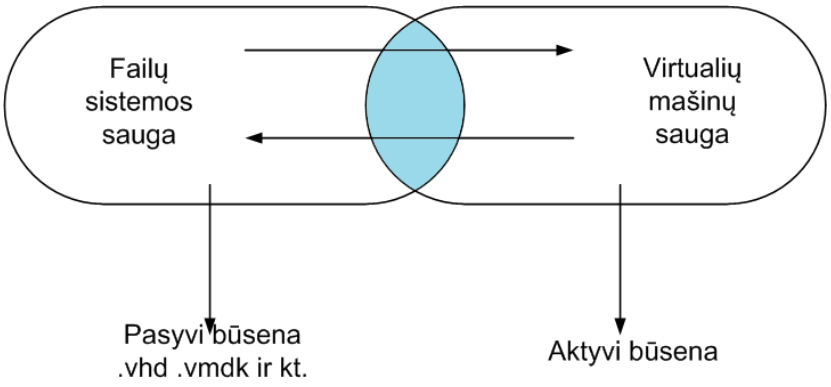
1.2.2. Virtualių mašinų failų saugos problemos

Virtualios mašinos gyvavimo ciklas nėra apibrėžtas, tačiau VM turi baigtinį būsenų skaičių, todėl vienu ar kitu laiko momentu virtuali mašina įgauna pasyvią būseną. Išjungtoje būsenoje esanti virtuali mašina yra tik failų rinkinys. Kiekvienas ją sudarantis failas saugo

`

tam tikrą informaciją bei yra būtinas virtualios mašinos funkcionavimui, o modifikavus bent vieną failą gali sutrikti visos virtualios mašinos darbas. Svarbiausias visos sistemos failas yra virtualus kietasis diskas, kuriame saugomi virtualios mašinos duomenys.

Naujausios virtualizacijos platformos ar trečių šalių programos suteikia galimybę prijungti (angl. mount) nenaudojamą t.y. pasyvioje būsenoje esantį virtualų diską kaip atskirą particiją. Po prijungimo galima skaityti ir rašyti informaciją į šį diską taip, tarsi jis būtų dar vienas kompiuterio kietasis diskas. Administratoriai naudoją minėtą funkciją norėdami ištraukti reikiamus duomenis ar įdiegti naują programinę įrangą nejungdami virtualios mašinos. Vertindami funkcionalumo patogumą turime atkreipti dėmesį į galimą saugumo spragą. Šioje situacijoje virtualių mašinų sauga persipina su failų sauga (9 pav.), nes niekas nėra užtikrintas, kad prijungto disko duomenys nebus modifikuoti ar perskaityti neautorizuoto asmens.



Pav. 9 Failų sistemų sauga virtualiose mašinose

Saugi failų sistema turi užtikrinti pagrindinius informacijos saugumo tikslus – konfidencialumą, vientisumą ir prieinamumą. Šiems tikslams pasiekti failų sistemose taikomi įvairūs metodai: prieigos prie duomenų ribojimas, atsilaisvinusios vietos laikmenoje išvalymas, duomenų šifravimas, failų sistemos darbo registravimas.

Microsoft Windows BitLocker paslauga

Siekiant išspręsti duomenų vientisumo ir konfidencialumo saugos problemas vienas iš galimų pasirinkimų yra viso disko šifravimo FDE (angl. full disk encryption) [18]. Duomenų šifravimas dažnai naudojamas siekiant apsaugoti informaciją nuo modifikacijos ar nutekėjimo. FDE gali užšifruoti visus duomenis, t.y. įskaitant operacinės sistemos failus, laikinuosius failus ir vartotojo failus. Šifruotų duomenų saugumas pagrįstas viešo ir privataus rakto principu, kuris naudojamas duomenų užšifravimui ir iššifravimui.

`

Microsoft Windows BitLocker yra vienas iš daugelio programinės įrangos pagrindu veikiančių sprendimų. Šis įrankis standartiškai diegiamas kartu su serverių virtualizacijai skirta Hyper-V platforma bei Windows Server operacine sistema.

BitLocker disko šifravimas yra saugos funkcija, kuri apsaugo duomenis kompiuteryje užšifruodama visus operacinės sistemos tome saugomus duomenis [27]. Šis įrankis veikia naudodamas paprastuosius tomus, kur viename tome yra vienas skaidinys. Tomui paprastai priskiriama loginio disko raidė. Patikimos platformos modulis TPM (angl. trusted platform module) yra į kompiuterį įmontuotas mikrolustas. Jis naudojamas saugoti kriptografinę informaciją, t.y. šifravimo raktus. TPM saugoma informacija gali būti apsaugota nuo išorinių programinės įrangos atakų ir fizinės vagystės. BitLocker naudoja TPM, taip apsaugodama operacinę sistemą ir vartotojo duomenis bei padėdama užtikrinti, kad kompiuterio duomenys nebuvo pakeisti, net jei kompiuteris buvo paliktas be priežiūros, pamestas arba pavogtas. Šis disko šifravimo įrankis gali būti naudojamas ir be TPM, o šifravimo raktai gali būti saugomi USB atmintinėje ar kitame aparatinės įrangos atmintuke.

Microsoft Windows EFS paslauga

Dar vienas būdas, galintis užtikrinti duomenų konfidencialumą, yra failų sistemų šifravimas [22][23]. Aplankų ir failų užšifravimas - tai būdas juos apsaugoti nuo nepageidaujamos prieigos. Šifruojama failų sistema EFS (angl. encrypting file system) yra operacinės sistemos funkcija, kuria galima įrašyti informaciją į standųjį diską šifruotu formatu. EFS veikimas pagrįstas mišria simetrinio ir asimetrinio rakto technologija. Abi šios šifravimo technologijos turi savo stipriąsias ir silpnąsias puses, o EFS šifravime stengiamasi pasinaudoti šių technologijų privalumais.

Simetrinis šifravimo raktas (dar vadinamas slaptuoju raktu) naudoja vieną raktą užšifruoti ir iššifruoti duomenis [24]. Vienas simetrinio šifravimo algoritmo privalumų - greitas veikimas, todėl jis idealiai tinka šifruoti didelius duomenų kiekius. Šio algoritmo problema - raktų paskirstymas. Jeigu reikia suteikti prieigą prie užšifruotų duomenų daugiau nei vienam asmeniui, privaloma suteikti prieigą ir prie slaptojo rakto. Asimetrinis raktų šifravimo algoritmas buvo sukurtas siekiant išspręsti šią paskirstymo problemą. Asimetrinis šifravimo algoritmas pagrįstas dviejų raktų principu – viešuoju ir privačiuoju. Viešasis raktas gali būti laisvai platinamas, o privatusis privalo būti saugomas. Asmuo gali užšifruoti informaciją savo viešuoju raktu, o šią informaciją galės iššifruoti tik asmuo, turintis slaptąjį raktą.

Pirmą kartą šifruojant failą EFS vartotojui priskiria viešojo ir privataus raktų porą. Kai failas yra užšifruotas, EFS generuoja atsitiktinį skaičių, kuris vadinamas FEK (angl. file

`

encryption key). FEK naudojamas užšifruoti failo turinį, naudojant duomenų šifravimo standartą DESX. DESX apdoroja duomenis tris kartus su trimis skirtingais raktais. FEK yra saugoma operacinėje sistemoje ir yra užkoduotas RSA algoritmu, naudojant vartotojo viešąjį raktą. Tokiu prinicipu kiekvienam šifruotam failui saugomos dvi šifruotės – viena failui, o kita failo FEK.

Microsoft Windows BitLocker ir Microsoft Windows EFS paslaugų palyginimas

Yra keletas skirtumų tarp viso disko šifravimo (Microsoft Windows BitLocker) ir šifruojamos failų sistemos (Microsoft Windows EFS) [28]. BitLocker sukurtas apsaugoti visiems asmeniniams ir sistemos failams diske (kuriame įdiegta operacinė sistema), jeigu kompiuteris neteisėtai pasisavinamas arba neteisėtai bandoma juo pasinaudoti. EFS naudojama padėti apsaugoti atskirus failus bet kuriame vartotojo diske. 2 lentelėje nurodomi pagrindiniai skirtumai tarp BitLocker ir EFS.

2 lentelė BitLocker ir EFS palyginimas

Viso disko šifravimas (Microsoft Windows BitLocker paslauga)	Failų sistemos šifravimas (Microsoft Windows EFS paslauga)
BitLocker užšifruoja visus operacinės sistemos disko, fiksuotų duomenų diskų ir keičiamųjų duomenų diskų asmeninius ir sistemos failus.	EFS iš eilės užšifruoja asmeninius failus ir aplankus, tačiau neužšifruoja viso disko turinio.
BitLocker nepriklauso nuo atskirų vartotojo profilių, susijusių su failais. Viso disko šifravimas gali būti įjungtas arba išjungtas visiems arba grupei vartotojų.	EFS šifruoja failus pagal su jais susijusius vartotojų profilius. Jei kompiuteryje yra keletas vartotojų arba grupių, kiekvienas jų gali šifruoti savo failus atskirai.
BitLocker naudoja patikimos platformos modulį (TPM), specialųjį mikrolustą daugelyje kompiuterių, kurie palaiko papildomas saugos funkcijas, skirtas operacinės sistemos diskui šifruoti.	EFS nereikalauja specialiosios aparatūros.
Įjungti arba išjungti viso disko šifravimą gali tik administratorius.	Visi naudotojai gali naudoti EFS.
Užtikrinamas duomenų konfidencialumas ir vientisumas, bet neužtikrinamas prieinamumas.	Užtikrinamas duomenų konfidencialumas ir prieinamumas, bet neužtikrinamas vientisumas.

`

1.2.3. Antivirusinių programų licencijavimas

Organizacijose, kuriose reguliariai naudojama virtualizacijos technologija, iškyla didelė problema perkant bei diegiant antivirusinę programinę įrangą. Tai ypač aktuali švietimo įstaigoms, kuriose kiekvienais metais keičiasi besimokantys asmenys bei jų skaičius, todėl sunku prognozuoti, kiek ir kokiam laikotarpiui reikės licencijų. Ruošiant darbą, buvo atliktas tyrimas, kurio metu anonimiškai apklausti didžiausi antivirusinių programų pardavėjai. Tyrimo metu buvo teiraujamasi apie kiekvieno pardavėjo platinamos antivirusinės programos licencijavimą virtualiose mašinose. Pateikta konkreti situacija: „Pavyzdžiui, jeigu įmonės mokymų klasėje yra 20 kompiuterių, kurie naudoti tik 2 mėnesius. Po 2 mėnesių naikinamos virtualios mašinos ir kitiems mokymams kuriamos naujos virtualios mašinos. Kaip skaičiuojamos antivirusinių programų licencijos?“

Antivirusinių programų pardavėjai vienodai traktavo virtualias mašinas bei antivirusinių programų licencijavimą jose. Buvo gauti tokie atsakymai:

- „Virtuali mašina yra traktuojama lygiai taip pat, kaip ir realus fizinis kompiuteris. Todėl pvz, jeigu jūs turite 20 kompiuterių ir juose norėsite leisti 20 virtualių mašinų, turėsite įsigyti 20+20 licencijų.“
- „Antivirusinė programa licencijuojama pagal tai, kiek kompiuterių prie jos jungiama. O dėl laikotarpio - galima užsakyti reikiamam laikotarpiui, pvz. 2 mėn., arba užsakote vieneriems ar dvejiems metams. Licencijos „nepririštos“ prie kompiuterių, o „pririštos“ prie įmonės.”
- „Siūlome įsigyti licencijas su išskaičiavimu kiekvienai aktyviai VM. Antivirusinės licenciją galite „permesti“ nuo kiekvienos (senos) išdiegiamos VM į kitą - naujai diegiamą.”

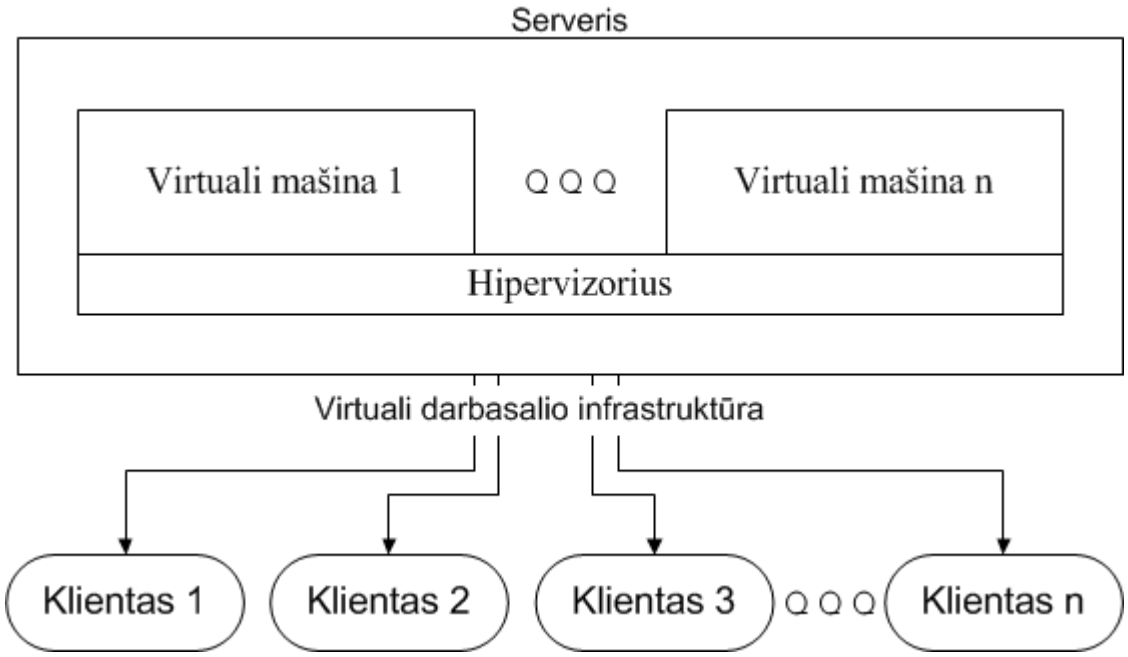
Iš pateiktų atsakymų akivaizdu, kad kiekvienai virtualiai mašinai reikia pirkti atskirą licenciją. Vieni siūlo licencijas skaičiuoti pagal tai, kiek yra virtualių mašinų, o pasibaigus VM gyvavimo ciklui naikinti virtualią mašiną kartu su antivirusinės licencija. Kiti siūlo licencijas skaičiuoti išvis neatsižvelgiant į aktyvių virtualių mašinų kiekį.

1.2.4. Darbo vietų virtualizacijos apsaugos problemos

Naudojantis VDI tipo darbo vietos virtualizacija, virtuali mašina emuliuojama per serverį, o vartotojas gali naudoti storo kliento (angl. fat client) arba plono kliento (angl. thin client) programas. Visos operacijos, kurias atlikdavo įprastas personalinis kompiuteris yra iškeliamos į duomenų centrą, vartotojui paliekant tik informacijos išsiuntimui, gavimui bei atvaizdavimui reikalingus įrankius.

`

VDI tipo virtualizacija vartotojams leidžia turėti dedikuotus sisteminius resursus, individualias aplikacijas, galimybę perkrauti sistemą bei jungtis naudojant įvairius įrenginius. Administratoriams, VDI sprendimas, suteikia galimybę valdyti visus vartotojus centralizuotai, naudojant virtualizacijos platformas, tokias kaip vSphere ar Hyper-V. Virtualios darbo vietos infrastruktūra pateikta 10 paveiksle.



Pav. 10 Darbo vietos virtualizacija naudojantis VDI

Kuriant virtualias mašinas chaotiškai, kai nėra prognozuojamas jų veiklos ir gyvavimo laikas, susiduriama su antivirusinės programinės įrangos licencijavimo problemomis, nes sunku nustatyti reikiamą licencijų skaičių. Pavyzdžiui, studentui universitete sukuriama VDI vienai savaitei – vienam laboratoriniam darbui. Kita VDI sukuriama vienam mėnesiui – keliems laboratoriniams darbams. Dar kita VDI sukuriama visam semestrui – visiems, vieno modulio, laboratoriniams darbams. Kiekvienam moduliui, o kartais ir laboratoriniam darbui, reikalinga nauja VDI, nes keičiasi operacinės sistemos aplinka, programinė įranga bei reikalavimai darbo vietai.

O jeigu yra 10 ar daugiau įvairių modulių su visiškai skirtingais VDI gyvavimo laikais bei darbo vietos reikalavimais? Kaip tokiu atveju apsaugoti visas darbo vietas bei laikytis antivirusinės programinės įrangos licencijavimo tvarkos? Antivirusinių programų licencijavimas virtualioms mašinoms traktuojamas taip pat, kaip ir realiam fiziniam kompiuteriui, todėl kiekvienai virtualiai mašinai reikalinga nauja licencija. Šis licencijų skaičiavimo būdas apsunkina jų įsigijimą, todėl sunku prognozuoti reikiamą licencijų skaičių, ko pasėkoje ir išlaidas.

`

Ši problema aktuali visoms įstaigoms, kuriose taikomas vieno naudotojo VDI principas. Šio principo esmė, kad kiekvienas naudotojas gauna atskirą VDI, kur atitinkama virtuali mašina leidžiama iš virtualių mašinų telkinio, o tame telkinyje lieka kai yra išjungiama bei laukia sekančio įjungimo.

1.3. Išvados

- ✓ Virtualizacijos technologija leidžia efektyviau išnaudoti procesorių branduolius, atmintį, diskus ir valdiklius, kiekviename fiziniame įrenginyje.
- ✓ Platus virtualizacijos taikymas, virtualių darbo vietų sukūrimui, padidina erdvę kenkėjiškoms atakoms.
- ✓ Antivirusinių programų licencijavimas virtualioms mašinoms traktuojamas taip pat, kaip ir realiam fiziniam kompiuteriui, todėl kiekvienai virtualiai mašinai reikalinga nauja licencija.
- ✓ Nereguliarus virtualių mašinų naudojimas apsunkina antivirusinės programinės įrangos licencijavimo tvarkos užtikrinimą ir palaikymą.
- ✓ Virtualių mašinų turinio saugojimas atviro formato failuose, piktavaliams suteikia galimybes įnešti nepageidaujamą programinę įrangą į pasyvioje būsenoje esančias virtualias mašinas.

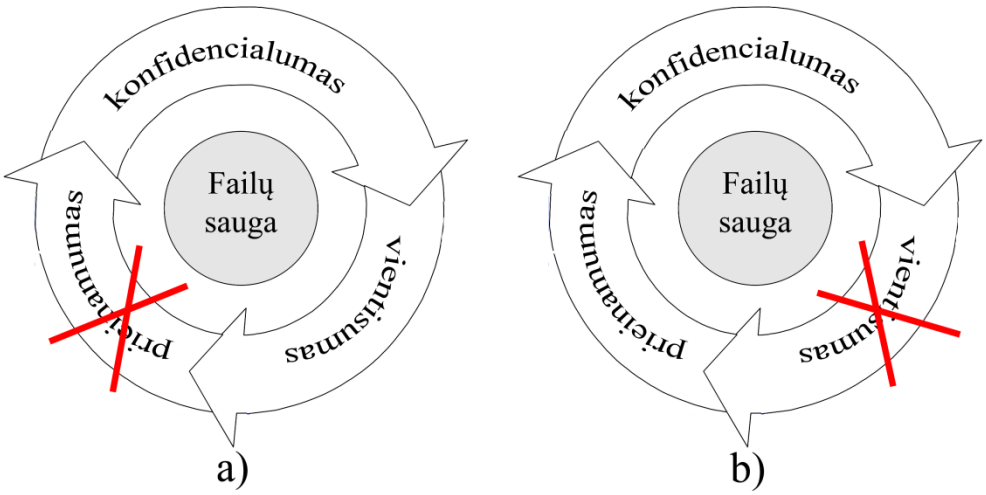
`

2. VIRTUALIOS APLINKOS SAUGOS SISTEMOS PROTOTIPAS

Virtualios aplinkos saugos tyrimo motyvacija

Pasyvioje būsenoje esanti virtuali mašina yra tik fizinis failas, kurį galima modifikuoti įsilaužus į pagrindinę operacinę sistemą, ir taip sukelti dideles saugumo pasekmes. Literatūroje aprašyta daugybė galimų atakų, viena tokių - „Blue pill“. Ši ataka pristatyta saugumo tyrinėtojos Joanos Rutkowskos [5]. Atakos veikimo principas - pagrindinė operacinė sistema perkeliama į virtualizuotos operacinės sistemos lygmenį, o tarp aparatinės įrangos ir operacinės sistemos įterpiamas virtualizuojantis „rootkit“ kodas. „Blue pill“ metu pagrindinė operacinė sistema nebetenka galimybės priešintis, nes visos instrukcijos, ateinančios iš pagrindinės operacinės sistemos, yra perimamos virtualizacijos lygmenyje ir gali būti modifikuojamos.

Pagrindinės operacinės sistemos atžvilgiu, kai virtuali mašina yra pasyvioje būsenoje, VM saugą galima prilyginti failų sistemos saugai. Saugi failų sistema turi užtikrinti pagrindinius informacijos saugumo tikslus – konfidencialumą, vientisumą ir prieinamumą. Šiems tikslams pasiekti taikomi įvairūs metodai. Analizės dalyje buvo aprašyti naudojami saugos metodai t.y. visko disko šifravimas naudojantis BitLocker ir failų sistemos šifravimas EFS, tačiau jie tik iš dalies užtikrina pagrindinius informacijos saugumo tikslus (11 pav).



Pav. 11 Šifravimas a) viso disko pagrindinių informacijų saugumo tikslų užtikrinimas b) failų sistemos pagrindinių informacijų saugumo tikslų užtikrinimas

Paveiksle 11a pavaizduota, kokius pagrindinius informacijos saugumo tikslus užtikrina viso disko šifravimas naudojantis BitLocker. Užšifravus visą diską užtikrinamas duomenų konfidencialumas bei vientisumas, tačiau užkertamas kelias duomenų prieinamumui. Taip pat atimama galimybė pagrindinės operacinės sistemos pagalba prijungti virtualią mašiną bei joje

`

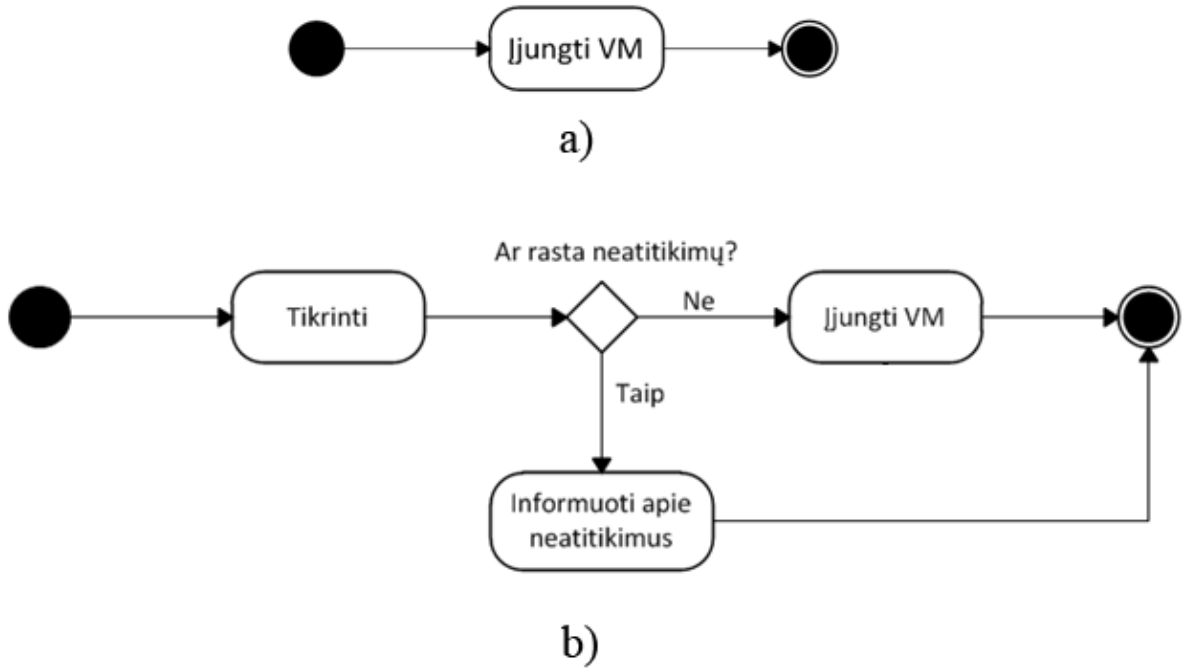
naršyti. Administratoriai naudoją šią funkciją norėdami ištraukti reikiamus duomenis ar įdiegti naują programinę įrangą nejungdami virtualios mašinos, todėl viso disko šifravimas nėra patogus bei mėgstamas administratorių. Be to, viso disko šifravimas yra daug laiko reikalaujantis darbas, kadangi kuo didesnė virtuali mašina, tuo ilgiau diskas šifruojamas.

Paveiklse 12b pavaizduota, kokius pagrindinius informacijos saugumo tikslus užtikrina failų sistemos šifravimas. Užšifravus failų sistemą, užtikrinamas duomenų konfidencialumas bei prieinamumas, tačiau neužtikrinamas vientisumas. Aktuali išlieka ta pati problema, kad ir į užšifruotą failų sistemą galima įnešti kenkėjišką programą, o aktyvavus virtualią mašiną aktyvuojasi ir kenkėjiška programa.

Tinkamo sprendimo, kuris padėtų užtikrinti visus pagrindinius informacijos saugumo tikslus – konfidencialumą, vientisumą ir prieinamumą, virtualiose mašinose, kol VM yra pasyvioje būsenoje, nėra.

2.1. Virtualios aplinkos saugos problemos formulavimas

Įprastai virtualios mašinos įjungiamos be jokių papildomų tikrinimų, t.y. niekada netikrinama, ar virtualios mašinos failų struktūra nepakitusi nuo to laiko, kai ji buvo išjungta. Šį „AS-IS“ modelį vaizduoja 12a paveikslas.



Pav. 12 a) AS-IS modelis b)TO-BE modelis

Administratoriui tai yra įprasta, patogu ir greita, tačiau šiame procese galima įžvelgti saugumo spragų. Analizės dalyje aprašyme, kad pasyvioje būsenoje esanti virtuali mašina yra

`

tik fizinis failas. Pagrindinės operacinės sistemos ar įvairių trečių šalių įrankių pagalba šį failą galima prijungti, o jo failų sistemoje naršyti tarsi paprastame kietajame diske. Piktavaliui įsilaužus į pagrindę operacinę sistemą iškyla saugumo grėsmė visoms, atitinkamame serveryje esančioms, virtualioms mašinoms. Virtuli mašina yra baigtinis automatas, todėl vienu ar kitu laiko momentu įgyja pasyvią būseną, o tuo metu atsiranda galimybė įnešti kenkėjišką programą. Administratoriui aktyvavus virtualią mašiną kartu aktyvuojasi ir kenkėjiška programa.

Saugioje sistemoje privaloma tikrinti kiekvieną virtualią mašiną, kurios būsena (visos galimos virtualios mašinos būsenos pateiktos 3 lentelėje) iš išjungta arba išsaugota keičiasi į įjungiama arba atkuriama. Būtina tikrinti ar VM failų struktūra nepakitusi nuo to laiko kai ji buvo išjungta. Esant kokiem nors neatitikimams sistema privalo informuoti administratorių apie galimą saugumo pavojų. Šį „TO-BE“ modelį, vaizduoja 12b paveikslas.

3 lentelė visos įmanomos virtualios mašinos būsenos

Virtualios mašinos būsena	Būsenos aprašymas	Būsenos trukmė	Ar galima prijungti?
Išjungta	VM išjungta be galimybės atkurti sesiją.	Ilgalaikė	**Taip**
Išsaugota	VM išjungta su galimybe atkurti sesiją.	Ilgalaikė	**Taip**
Įjungiama	VM įjungiama be sesijos atkūrimo.	Trumpalaikė	Ne
Atkuriama	Atkuriama VM sesija.	Trumpalaikė	Ne
Veikia	VM įjungta, jei reikėjo atkurta sesija.	Ilgalaikė	Ne
Sustabdyta	VM darbas sustabdytas.	Ilgalaikė	Ne
Saugoma	Sauga išjungiamos VM sesija.	Trumpalaikė	Ne
Išjungiama	VM išjungiama, sesija nesaugoma.	Trumpalaikė	Ne

2.2. Virtualios aplinkos saugos sistemos prototipo modelis

Norėdami apsaugoti virtualias mašinas, turime užtikrinti, kad būdamos pasyvioje būsenoje jos nebuvo modifikuotos ar pažeistos. Tai galime padaryti suskaičiuodami išjungiamos virtualios mašinos kontrolinę sumą bei palyginti su įjungiamos VM kontroline suma. Kontrolinės sumos pagalba galima patikrinti failų ar siunčiamų duomenų vientisumą. Paprastai kontrolinė suma naudojama įsitikinti ar du duomenų blokai yra vienodi. Ši suma gali būti skaičiuojama įvairiais būdais, naudojant skirtingus algoritmus. Paprasčiausia kontrolinė suma gali būti tiesiog failo baitų skaičius. Tačiau šis kontrolinės sumos tipas nėra pats patikimiausias, todėl dažniausiai naudojami sudėtingesni skaičiavimo algoritmai.

`

2.2.1. Duomenų vientisumo užtikrinimas

Nepatikimo disko sindromas (angl. untrusted disk assumption) paprastai yra paplitęs tinklu sujungtose duomenų sistemose, kur kliento failų sistema bendrauja su disku per nesaugų tinklą, todėl failų sistema potencialiai gali būti pažeista, jeigu kenkėjas aktyviai modifikuoja ar pasyviai klausosi failų sistemos duomenų srauto. Mūsų atveju svečio operacinė sistema nepasitiki pagrindine operacine sistema, todėl būtina užtikrinti svečio operacinės sistemos saugumą net ir virtualiai mašinai esant pasyvioje būsenoje.

Kontrolinės sumos skaičiavimas yra gerai žinomas duomenų vientisumą užtikrinantis metodas [16][17]. Vientisumą galima patikrinti palyginus dvi kontrolines sumas t.y. sumą suskaičiuotą prieš tam tikrą laiko momentą palyginus su suma suskaičiuota einamuoju laiko momentu. Jeigu šios sumos sutampa duomenys nepažeisti, jei nesutampa duomenys pažeisti.

Kriptografinio algoritmo pasirinkimas

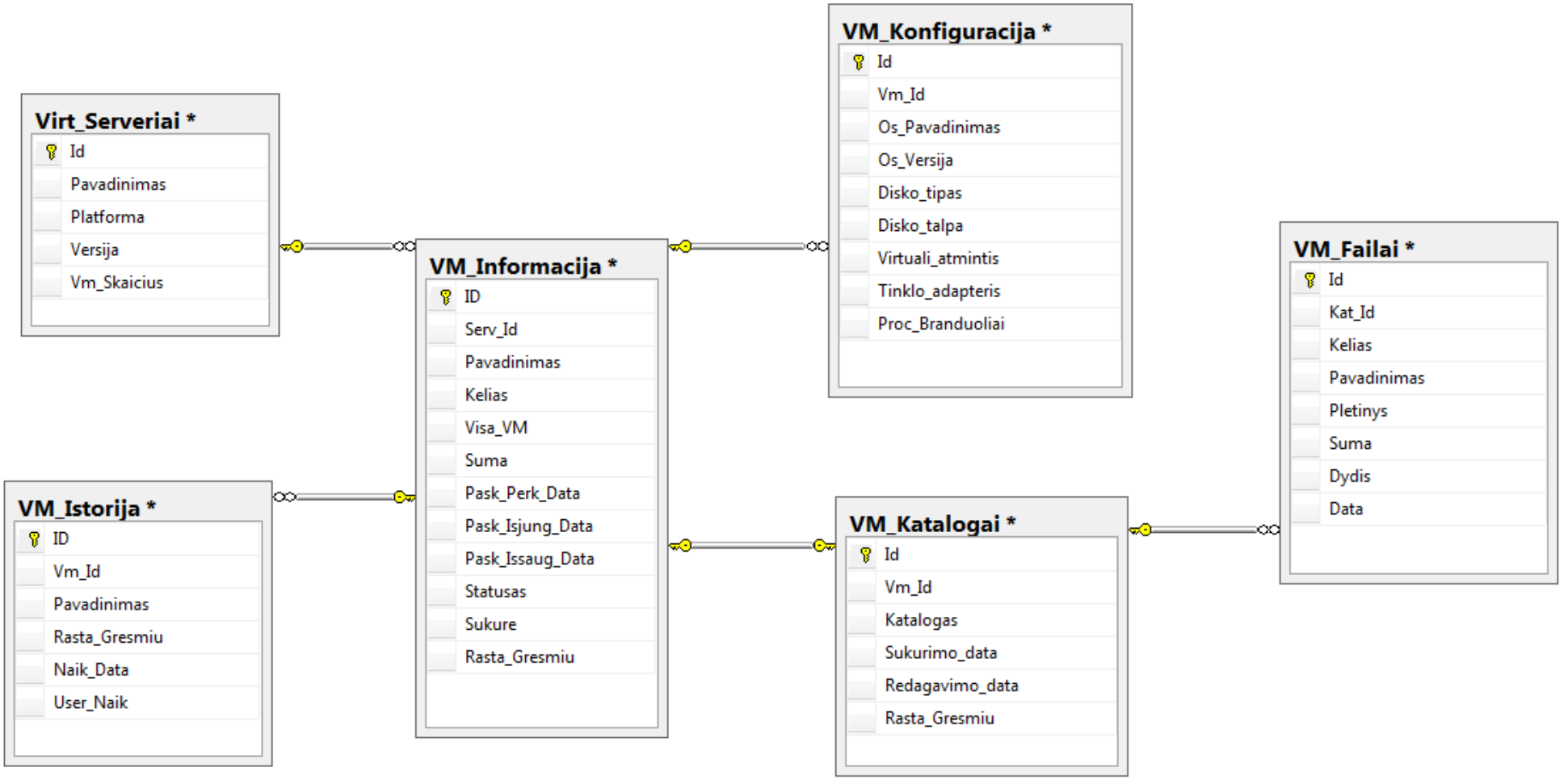
Šiuo metu labiausiai paplitę yra MD5 ir SHA-1 kriptografiniai maišos algoritmai, tačiau tiek vienas, tiek kitas yra pažeidžiamas [15]. Nuolat plečiantis kompiuterių technikai, jau tampa įmanoma rasti MD5 ir SHA-1 funkcijų kolizijas [19][20]. Iš pradžių buvo teigiama, kad šios kolizijos nepavojingos failo turinio kontrolinių sumų skaičiavimui, tačiau vėliau buvo pristatytos naujos publikacijos, kuriose įrodoma esą kolizijos egzistuoja ir kontrolinių sumų skaičiavime.

NESSIE (New European Schemes for Signatures Integrity and Encryption) rekomenduoja naudoti Whirlpool, SHA-256, SHA-384 arba SHA-512 kriptografinės maišos funkcijas kurios yra atsparios kolizijoms [21]. Remdamiesi rekomendacijomis failo turinio kontrolinės sumos skaičiavimui naudosime SHA-256 algoritmą.

Kontrolinės sumos saugojimas

Apskaičiuotą, išjungiamos virtualios mašinos, kontrolinę sumą reikia išsaugoti ir laikyti iki atitinkama virtuali mašina bus įjungiama arba panaikinta. Patogus ir patikimas būdas duomenis saugoti duomenų bazėje. Čia informacija saugoma struktūrizuotai bei ją lengvą apdoroti. Duomenų bazė privalo būti tinkamai sukonfigūruota bei apsaugota nuo neautorizuotų naudotojų informacijos nuskaitymo ar modifikavimo.

13 paveiksle pateikiama duomenų bazės releacinė schema, o 4-9 lentelėse pateikiamos duomenų bazės releacinės lentelės.



Pav. 13 Duomenų bazės releacinė schema

4 lentelė. Virt_Serveriai releacinė lentelė

Virt_Serveriai				
Lauko pavadinimas	Raktas	Tipas	Privalomas	Paskirtis
Id	Taip	Skaitinis	Taip	Vidinis identifikatorius.
Pavadinimas	Ne	Simbolinis	Taip	Virtualaus serverio pavadinimas.
Platforma	Ne	Simbolinis	Taip	Naudojama virtualizacijos platforma.
Versija	Ne	Simbolinis	Taip	Virtualizacijos platformos versija.
Vm_skaicius	Ne	Skaitinis	Taip	Virtualių mašinų skaičius esantis virtualiame serveryje.

Lentelėje Virt_Serveriai saugoma informacija apie visus virtualius serverius esančius debesyje. Čia saugomas informacija apie atitinkamame serveryje naudojamą virtualizacijos platformą bei virtualių mašinų skaičių.

5 lentelė. VM_Informacija releacinė lentelė

VM_Informacija				
Lauko pavadinimas	Raktas	Tipas	Privalomas	Paskirtis
Id	Taip	Skaitinis	Taip	Vidinis identifikatorius.
Serv_Id	Taip	Skaitinis	Taip	Virtualaus serverio Id, kuriam priklauso virtuali mašina.
Pavadinimas	Ne	Simbolinis	Taip	Virtualios mašinos pavadinimas.
Kelias	Ne	Simbolinis	Taip	Kelias iki virtualios mašinos.
Visa_VM	Ne	Simbolinis	Taip	Nurodomas 1 jei saugoma visos VM kontrolinė suma. Nurodomas 0 jei saugoma jautraus katalogo kontrolinė suma.
Pask_Perk_Data	Ne	Data ir laikas	Ne	Paskutinė virtualios mašinos perkrovimo data.
Pask_Isjung_Data	Ne	Data ir laikas	Ne	Paskutinė virtualios mašinos išjungimo data.
Pask_Issaug_Data	Ne	Data ir laikas	Ne	Paskutinė virtualios mašinos išjungimo su galimybe atkurti data.
Statusas	Ne	Simbolinis	Taip	Virtualios mašinos statusas.
Sukurimo_Data	Ne	Data ir laikas	Taip	Virtualios mašinos sukūrimo data.
Rasta_Gresmiu	Ne	Skaitinis	Ne	Aptiktų grėsmių skaičius.

Lentelėje VM_Informacija saugomi detalūs kiekvienos virtualios mašinos duomenys. Laukas Visa_VM nurodo ar kontrolinė suma suskaičiuota visai virtualiai mašinai ar tik jautriam jos katalogui. Šioje lentelėje fiksuojami kiekvienos virtualios mašinos sukūrimo, persikrovimo, išjungimo ir išjungimo su galimybe atkurti laikai. Taip pat skaičiuojamas ir fiksuojamas aptiktų grėsmių skaičius.

`

6 lentelė. VM_Istorija releacinė lentelė

VM_Istorija				
Lauko pavadinimas	Raktas	Tipas	Privalomas	Paskirtis
Id	Taip	Skaitinis	Taip	Vidinis identifikatorius.
Pavadinimas	Ne	Simbolinis	Taip	Virtualios mašinos pavadinimas.
Rasta_Gresmiu	Ne	Simbolinis	Taip	Aptiktų grėsmių skaičius.
Naik_Data		Date ir laikas	Taip	Virtualios mašinos ištrynimo data ir laikas.
Vart_Naik	Ne	Skaitinis	Taip	Vartotojo ištrynusio VM inicialai.

Lentelėje VM_Istorija saugoma istorinė informacija apie ištrintas virtualias mašinas bei ištynimo data ir vartotojo ištrynusio virtualią mašina inicialai. Įrašai fiziškai netrinami, nes atsiradus poreikiui visad galima peržiūrėti istoriją apie aptiktas grėsmes, nustatyti pavojingiausią virtualų serverį.

7 lentelė. VM_Konfiguracija releacinė lentelė

VM_Konfiguracija				
Lauko pavadinimas	Raktas	Tipas	Privalomas	Paskirtis
Id	Taip	Skaitinis	Taip	Vidinis įrašo identifikatorius.
Vm_Id	Taip	Simbolinis	Taip	Jungiamasis laukas su lentele VM_Informacija.
Os_pavadinimas	Ne	Simbolinis	Taip	Operacinės sistemos įdiegtos VM pavadinimas.
Os_versija	Ne	Simbolinis	Taip	Operacinės sistemos įdiegtos VM versija.
Disko_tipas	Ne	Skaitinis	Taip	Virtualaus kietojo disko tipas.
Disko_talpa	Ne	Skaitinis	Taip	Virtualaus kietojo disko talpa.
Virtuali_atmintis	Ne	Skaitinis	Taip	Virtualaus atmintis skaičius.
Tinklo_adapteris	Ne	Skaitinis	Taip	Virtualus tinklo adapteris.
Proc_Branduoliai	Ne	Skaitinis	Taip	Virtualaus procesoriaus branduolių skaičius.

Lentelėje VM_Konfiguracija saugomi virtualios mašinos konfigūraciniai duomenys. Čia talpinama informacija apie operacines sistemas veikiančias virtualiose mašinose, naudojamus virtualius disko tipus (gali būti fiksuotas, dinaminis, diferencijuojamas) bei diskų užimamą vietą serveryje, virtualią atmintį, tinklo adapterį ir virtualios mašinos procesoriaus branduolių skaičių.

`

8 lentelė. VM_Katalogai releacinė lentelė

VM_Katalogai				
Lauko pavadinimas	Raktas	Tipas	Privalomas	Paskirtis
Id	Taip	Skaitinis	Taip	Vidinis įrašo identifikatorius.
Vm_Id	Taip	Simbolinis	Taip	Jungiamasis laukas su lentele VM_Informacija.
Pavadinimas	Ne	Simbolinis	Taip	Jautraus katalogo pavadinimas
Sukurimo_data	Ne	Data ir laikas	Ne	Katalogo sukūrimo data ir laikas
Redagavimo_data	Ne	Data ir laikas	Ne	Katalogo redagavimo data ir laikas
Rasta_Gresmiu	Ne	Skaitinis	Ne	Aptiktų grėsmių skaičius.

Lentelėje VM_Katalogai saugoma informacija apie jautrius katalogus, kuriuos siekiama apsaugoti. Šioje lentelėje išsaugoma katalogo sukūrimo bei redagavimo datos. Taip pat skaičiuojamas ir fiksuojamas aptiktų grėsmių skaičius.

9 lentelė. VM_Failai releacinė lentelė

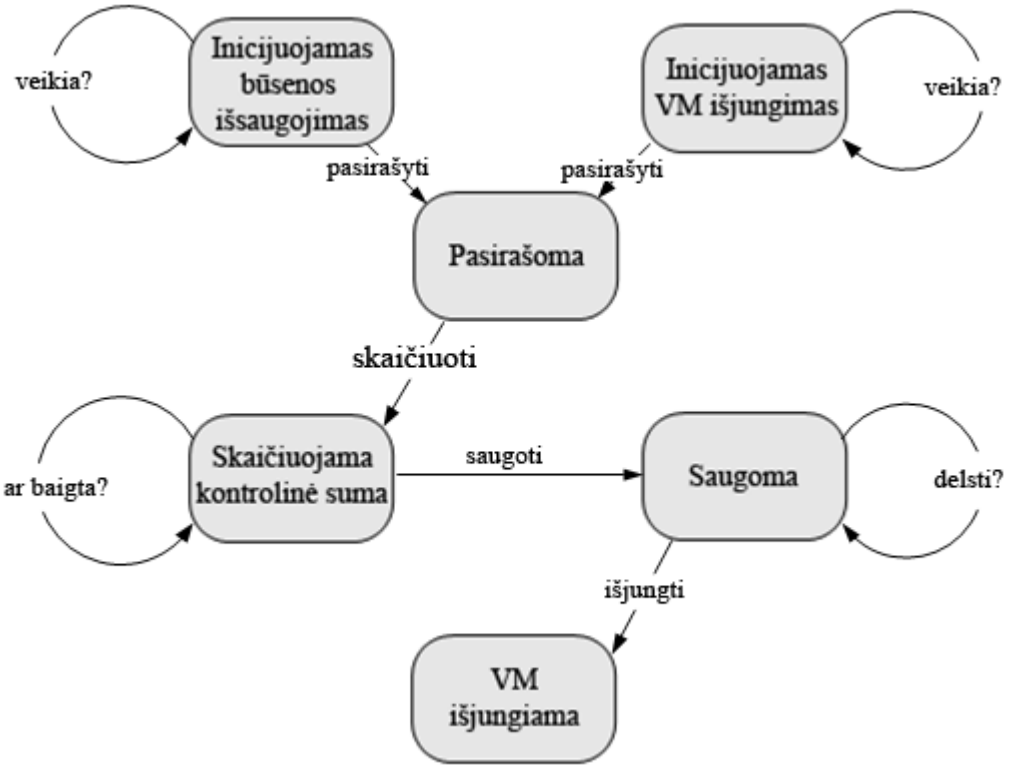
VM_ Failai				
Lauko pavadinimas	Raktas	Tipas	Privalomas	Paskirtis
Id	Taip	Skaitinis	Taip	Vidinis įrašo identifikatorius.
Kat_Id	Taip	Simbolinis	Taip	Jungiamasis laukas su lentele VM_Katalogai.
Pavadinimas	Ne	Simbolinis	Taip	Failo pavadinimas.
Kelias	Ne	Simbolinis	Taip	Kelias iki failo.
Pletinys	Ne	Simbolinis	Taip	Failo plėtinys.
Dydis	Ne	Skaitinis	Taip	Failo dydis.
Suma	Ne	Simbolinis	Taip	Failo kontrolinė suma.
Data	Ne	Data ir laikas	Taip	Įrašo sukūrimo data ir laikas.

Lentelėje VM_Failai saugomi kiekvieno failo esančio jautriame kataloge duomenys. Čia, kartu su visais failo parametrais, saugoma failo turinio kontrolinė suma.

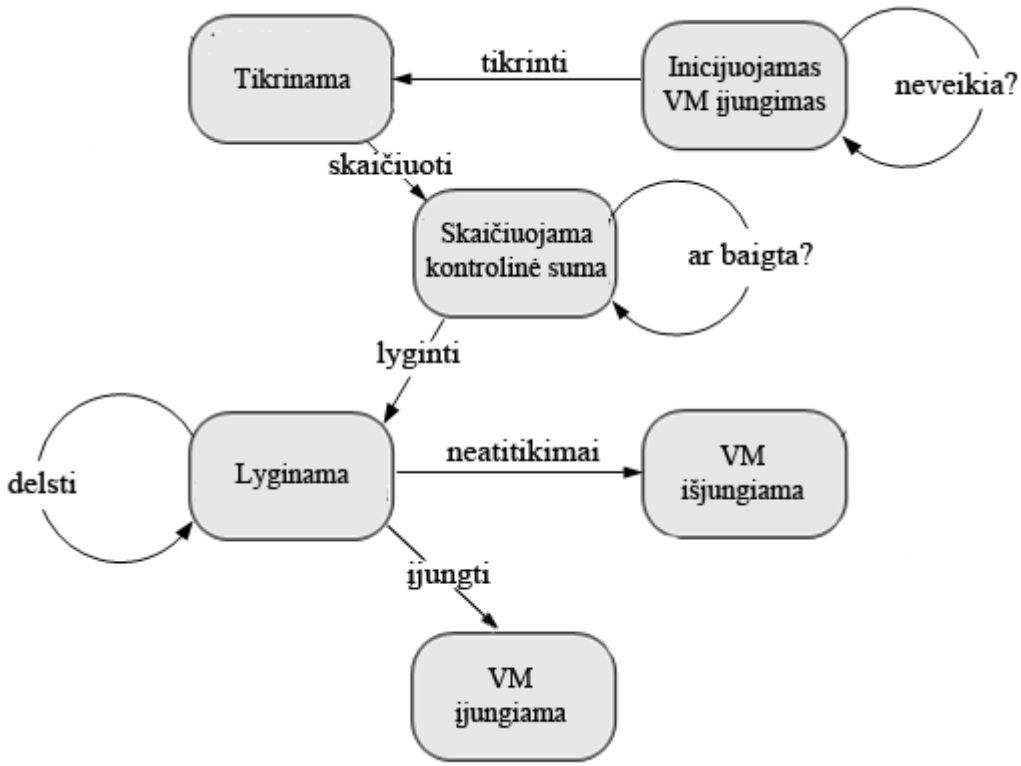
Virtualios mašinos pasirašymo ir tikrinimo modelis

Kontrolinių sumų skaičiavimas ir saugojimas turėtų būti vykdomas kritiniame VM taške. Išjungiamos virtualios mašinos kontrolinė suma turi būti skaičiuojama paskutinėje būsenoje po VM išjungimo inicijavimo. Įjungiamos virtualios mašinos kontrolinė suma turi būti skaičiuojama pirmoje būsenoje po įjungimo inicijavimo. Paveiksluose 14-15 pateiktas saugos modelio pasirašymo ir tikrinimo diagramos.

`



Pav. 14 Virtualios mašinos failo pasirašymo modelis



Pav. 15 Virtualios mašinos failo parašo tikrinimo modelis

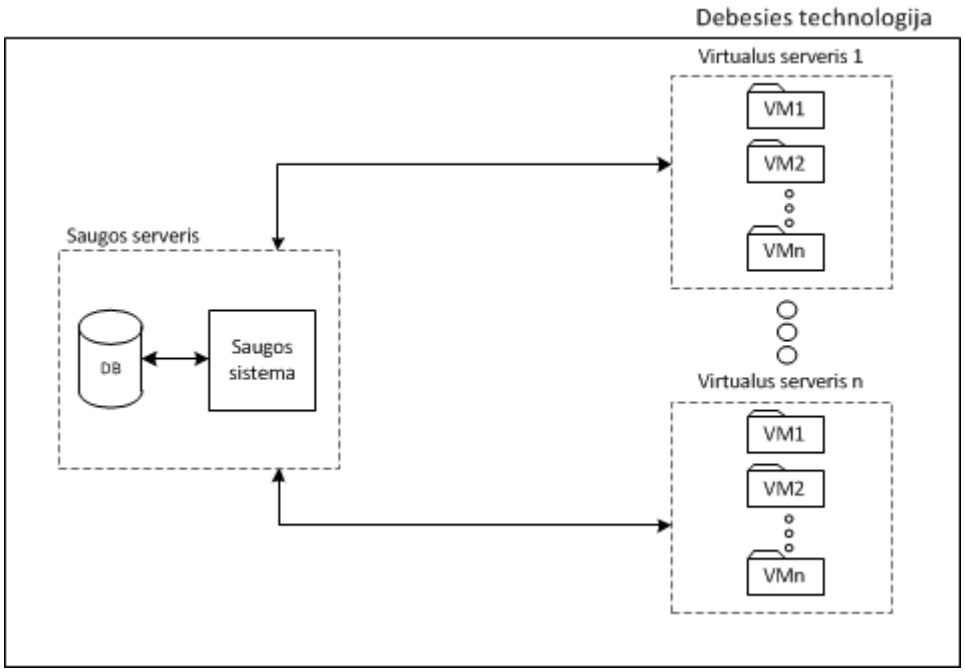
`

14 paveiklse matome, jog inicijavus virtualios mašinos išjungimą visų pirma atliekamas VM pasirašymas, kurio metu apskaičiuojama ir išsaugoma virtualios mašinos kontrolinė suma. Atlikus visus šiuos žingsnius virtuali mašina išjungiama.

Inicijavus virtualios mašinos įjungimą visų pirma patikrinam ar, būdama pasyvioje būsenoje, virtuali mašina nebuvo pažeista. Naujai suskaičiuojama virtualios mašinos kontrolinė suma bei palyginama su seniau apskaičiuota bei duomenų bazėje išsaugota suma. Jei kontrolinės sumos sutampa VM įjungiama, jei sumos nesutampa virtuali mašina neįjungiama.

Saugos sistemos schema

Šiuolaikinėje debesų kompiuterijoje (angl. cloud computing) viename debesyje yra daugybė virtualių serverių, todėl norėdami aprėpti visas debesyje esančias virtualias mašinas, saugos sistemą diegiama atskirame, nuo išorės izoliuotame, serveryje. Šis serveris atsakingas tik už virtualių mašinų saugą, o su kitais virtualiai serveriai bendrauja saugiu kanalu. Šiame serveryje taip pat diegiama duomenų bazę, kurioje saugoma kiekvienos VM kontrolinė suma. Bendra saugos sistemos schema pateikta 16 paveiksle. Kiekviename virtualiame serveryje diegiamas sistemos klientas, kurio pagalba bendrauja pagrindinė operacinė sistema ir saugos serveris.



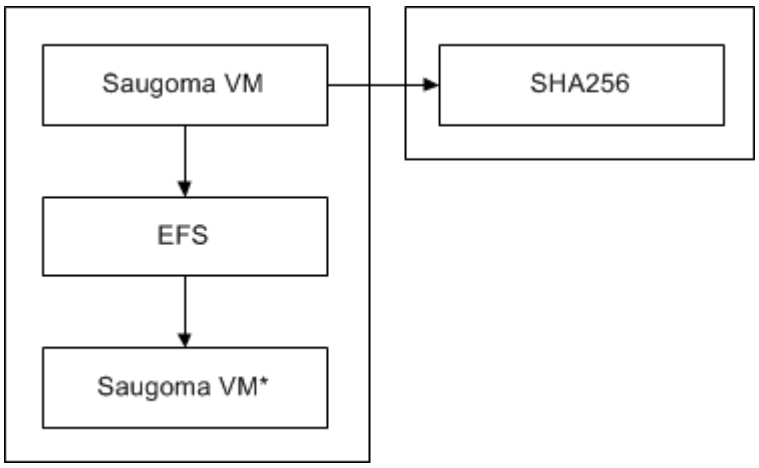
Pav. 16 bendra saugos sistemos schema

`

2.2.2. Virtualios mašinos apsaugojimas

Kontrolinės sumos skaičiavimo algoritmas

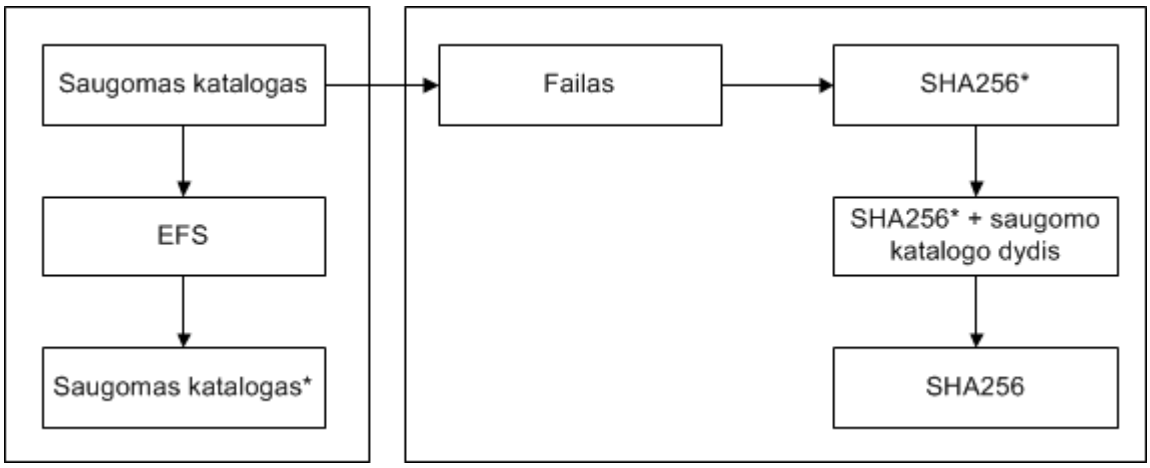
SHA-256 algoritmo pagalba galima sukurti unikalią 256 bitų failo turinio kontrolinę sumą, tačiau kontrolinės sumos suskaičiavimas ir lyginimas garantuoja tik duomenų vientisumą ir prieinamumą. Norėdami užtikrinti ir duomenų konfidencialumą turime pasinaudoti šifruojamos failų sistemos EFS metodu, kuris puikiai užtikriną konfidencialumą bei užkerta kelią neautorizuotų vartotojų prieigą. VM apsaugojimo algoritmas pateiktas 17 paveiksle



Pav. 17 VM apsaugojimo procesas

Kontrolinės sumos skaičiavimas yra laiko reikalaujantis algoritmas – kuo didesnė virtuali mašina, tuo ilgiau suma skaičiuojama. Todėl kartais patogiau ir greičiau skaičiuoti tik jautrių katalogų (pvz.: windows ar system32) sumas. Suskaičiavę kiekvieno failo kontrolinę sumą, naudodamiesi SHA-256 algoritmu, garantuojame, kad failas nebus modifikuotas. Taip pat turime užtikrinti, kad į saugomą katalogą nebus įneštas joks papildomas failas (turime išvengti pažeidžiamumo, kuris būdingas naudojant tik EFS šifravimą.), todėl į kiekvieno failo kontrolinę sumą papildomai įmaišome saugomo katalogo dydį. Tokiu būdu garantuojame, kad nei vienas failas esantis katalogo viduje nebus modifikuotas, o į saugomą katalogą nebus įneštas joks papildomas failas. Katalogo, esančio virtualioje mašinose, apsaugojimo algoritmas pateiktas 18 paveiksle.

`



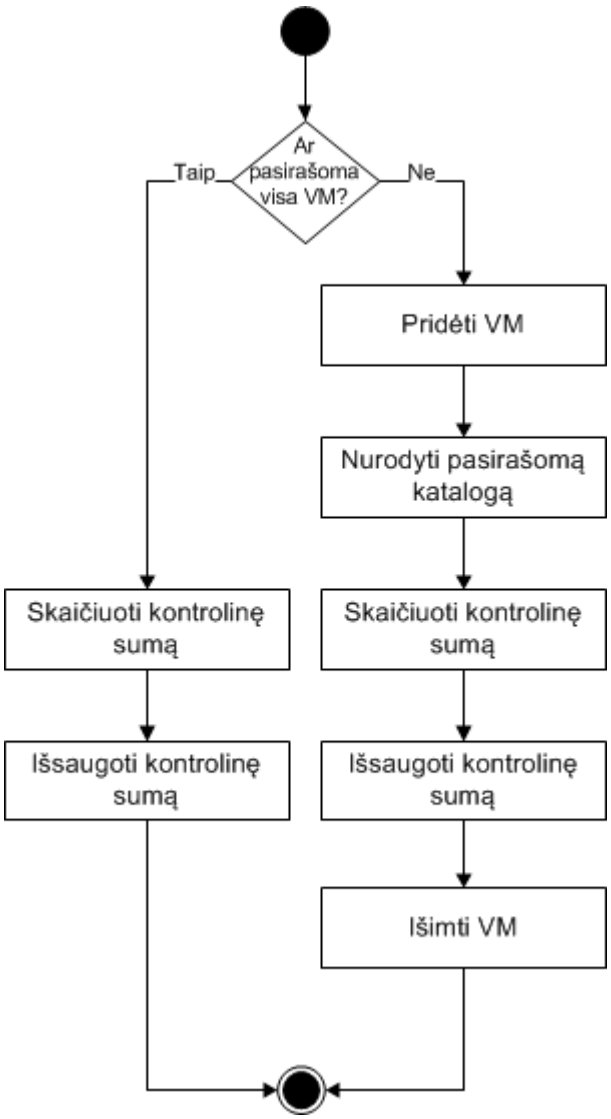
Pav. 18 Katalogo, esančio virtualioje mašinose, apsaugojimo procesas

Virtualios mašinos pasirašymo algoritmas

Įprastai visos virtualios mašinos yra vieno šablono kopijos, todėl, visų pirma, būtina suskaičiuoti ir išsaugoti šio šablono kontrolinę sumą. Sukūrus naują virtualią mašiną t.y. klonavus šabloną, turime VM, kurios kontrolinė suma jau žinoma. Todėl galime teigti, kad naujai sukurtos virtualios mašinos saugomos nuo pirmos egzistavimo akimirkos.

Kiekvieną kartą išjungdamas virtualią mašiną administratorius privalo naujai pasirašyti išjungiamą VM. 19 paveiksle pavaizduotas virtualios mašinos pasirašymo algoritmas. Jei pasirašoma visa virtuali mašina – tai suskaičiuojama bei išsaugoma jos kontrolinė suma. Jei pasirašomi tik svečio operacinei sistemai jautrus katalogas – tai atitinkama virtuali mašina prijungiama, suskaičiuojama bei išsaugoma nurodyto katalogo kontrolinė suma, virtuali mašina išimama.

`

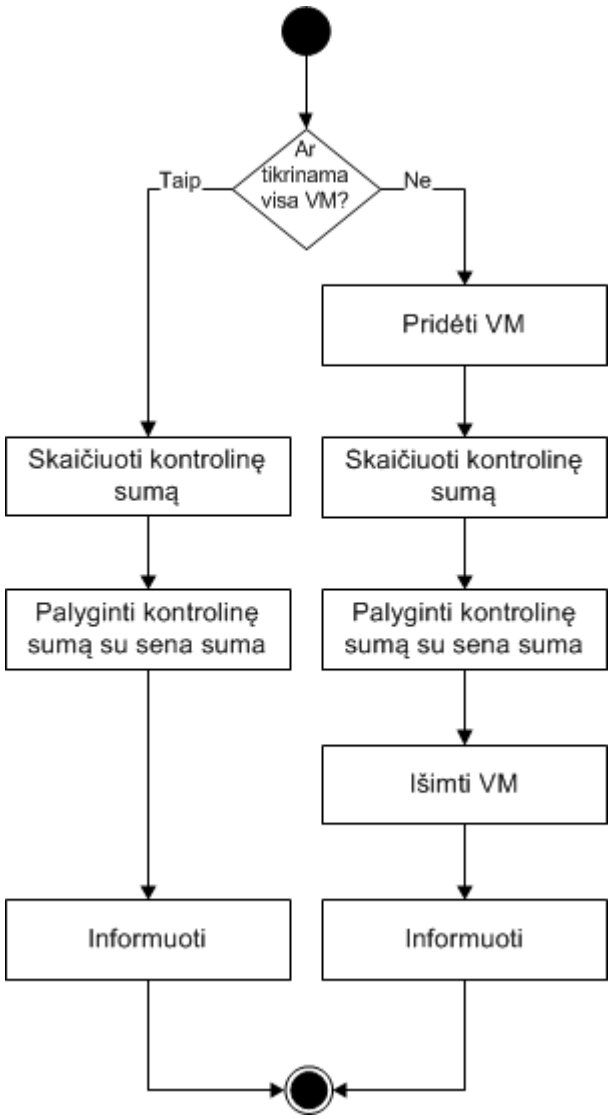


Pav. 19. Virtualios mašinos pasirašymo algoritmas

Virtualios mašinos vientisumo tikrinimo algoritmas

Prieš aktyvuojant virtualią mašiną privaloma patikrinti ar virtualioje mašinoje, kol ji buvo išjungta, neįvyko pakeitimų. 20 paveiksle pateiktas virtualios mašinos vientisumo tikrinimo algoritmas. Nustatoma ar pasirašymo metu buvo pasirašyta visa virtuali mašina ar tik svečio operacinei sistemai jautrus katalogas. Jeigu buvo pasirašyta visa VM – tai iš naujo suskaičiuojama visos virtualios mašinos kontrolinė suma bei palyginama su duomenų bazėje išsaugota suma. Jeigu buvo pasirašytas tik katalogas esantis virtualioje mašinoje – tai iš naujo suskaičiuojamos visų, atitinkamame kataloge esančių, failų kontrolinės sumos bei palyginamos su prieš tai išsaugotomis sumomis. Skaičiavimo ir lyginimo rezultatai pateikiami sistemos administratoriui.

`



Pav. 20. Virtualios mašinos vientisumo tikrinimo algoritmas

2.3. Išvados

- ✓ Pasiūlytas išorinis virtualių mašinų disko atvaizdo failo apsaugos metodas, naudojant kontrolinių sumų skaičiavimą ir pasirinkto katalogo failų sistemos šifravimą EFS.
- ✓ Suprojektuotas ir realizuotas virtualios aplinkos saugos sistemos prototipas pagrįstas kontrolinių sumų skaičiavimu ir failų sistemos šifravimu EFS, leidžiantis, prieš virtualios mašinos paleidimą, atlikti jos disko atvaizdo tikrinimą dėl piktavališkos programinės įrangos.
- ✓ Suprojektuota ir realizuota reliacinė duomenų bazė virtualių mašinų telkinio pasirašymo informacijai saugoti ir valdyti.

`

3. EKSPERIMENTINIS VIRTUALIOS APLINKOS SAUGOS SISTEMOS PROTOTIPO TYRIMAS

Šioje dalyje aprašomas eksperimentui suprogramuotas įrankis. Įrankio pagalba pasiekti rezultatai atvaizduojami diagramomis, bei palyginami su kitais, pagrindinius informacijos saugumo tikslus užtikrinančiais, metodais.

3.1. Tyrimo metodika

Tyrimui paruoštos keturios virtualios mašinos, kurios eksperimento metu buvo šifruojamos 5 skirtingais metodais. Visi variantai pateikiami 10 lentelėje.

10 lentelė. Tyrimo metu naudoti variantai

VM pavadinimas	VM dydis, GB	Saugomos failų sistemos dydis, GB	Šifravimo metodas
VM10-4	10	0,97	VASS pasirašant system32 katalogą.
		4	VASS pasirašant visą VM.
		0,97	VASS pasirašant system32 katalogą + EFS.
		0,97	EFS šifruojant system32 katalogą.
		4	BitLocker.
VM10-8	10	0,97	VASS pasirašant system32 katalogą.
		8	VASS pasirašant visą VM.
		0,97	VASS pasirašant system32 katalogą + EFS.
		0,97	EFS šifruojant system32 katalogą.
		8	BitLocker.
VM20-8	20	0,97	VASS pasirašant system32 katalogą.
		8	VASS pasirašant visą VM.
		0,97	VASS pasirašant system32 katalogą + EFS.
		0,97	EFS šifruojant system32 katalogą.
		8	BitLocker.
VM20-16	20	0,97	VASS pasirašant system32 katalogą.
		16	VASS pasirašant visą VM.
		0,97	VASS pasirašant system32 katalogą + EFS.
		0,97	EFS šifruojant system32 katalogą.
		16	BitLocker.

`

Šifruojamos dvi virtualios mašinos, kurių savasis dydis (disko dydis, kuris išskiriamos sukūrimo metu) 10GB, tačiau skirtingas failų sistemos dydis. Atitinkamai virtualios mašinos, kurios pavadinimas VM10-4 failų sistemos dydis 4GB, o VM10-8 failų sistemos dydis 8GB. Taip pat šifruojamos dvi virtualios mašinos, kurių savasis dydis 20GB, čia virtualios mašinos, kurios pavadinimas VM20-8 failų sistemos dydis 8GB, o VM20-16 failų sistemos dydis 16GB. Visų VM apsaugai naudojami 5 skirtingi metodai:

1. VASS (virtualios aplinkos saugos sistema) pasirašant system32 katalogą;
2. VASS (virtualios aplinkos saugos sistema) pasirašant system32 katalogą ir šifruojant Microsoft Windows EFS paslauga;
3. VASS (virtualios aplinkos saugos sistema) pasirašant visą VM;
4. Microsoft Windows EFS paslauga šifruojant system32 katalogą;
5. Microsoft Windows BitLocker paslauga;

3.1.1. Virtualių mašinų saugumo profiliai

Priklausomai nuo naudojamo apsaugos metodo, užtikrinami skirtingi informacijos saugumo tikslai. 11 lentelėje pateikti eksperimente naudoti VM apsaugotos metodai, čia kiekvienam metodui priskirtas saugumo profilis.

11 lentelė. Apsaugos metodų saugumo profiliai

Apsaugos metodo pavadinimas	Užtikrinamas vientisumas	Užtikrinamas konfidencialumas	Užtikrinamas prieinamumas	Apsaugoma visa failų sistema	Saugumo profilis
VASS	Taip	Ne	Taip	Ne	Vidutinis
VASS visos VM	Taip	Ne	Taip	Taip	Didelis
VASS + EFS	Taip	Taip	Taip	Ne	Didelis
EFS	Ne	Taip	Taip	Ne	Žemas
BitLocker	Taip	Taip	Ne	Taip	Didelis

Kiekvienas saugumo profilis sudarytas pagal tai kokius veiksnius užtikrina ir kiek atitinkamas veiksnys yra aktualus projektuojamai sistemai.

Analizuodami 11 lentelę matome jog VASS pasirašant visą virtualią mašiną, VASS pasirašant jautrų operacinės sistemos katalogą bei jį šifruojant EFS ir BitLocker turi didelį saugumo profilį. Visi šie metodai tenkina tris kriterijus iš kriterijų. VASS pasirašant jautrų operacinės sistemos katalogą bei failų sistemos šifravimas EFS tenkina du kriterijus iš trijų, tačiau VASS užtikrina vientisumą, kuris yra aktualesnis projektuojamai sistemai, todėl VASS priskiriamas vidutiniškai saugiam profiliui, o EFS žemam.

`

3.2. Tyrimo rezultatai

Skaitiniai rezultatai

Tyrimo metu nustatėme greičiausią apsaugos metodą bei įvertinome kiekvieno metodo veikimą esant skirtingam virtualios mašinos savajam dydžiui bei skirtingam failų sistemos dydžiui. Lentelėje 12 paryškinti apsaugos metodai, kurie priskiriami dideliam saugumo profiliui pagal 3.1.1 skyriuje nustatytus saugumo reikalavimus.

12 lentelė. Tyrimo metu naudoti variantai

VM pavadinimas	Apsaugojimo laikas, min	Saugomos failų sistemos dydis, GB	Šifravimo metodas
VM10-4	4:04	0,97	VASS pasirašant system32 katalogą.
	2:33	**4**	**VASS pasirašant visą VM.**
	9:10	**0,97**	**VASS pasirašant system32 katalogą + EFS.**
	5:06	0,97	EFS šifruojant system32 katalogą.
	7:10	**4**	**BitLocker.**
VM10-8	4:24	0,97	VASS pasirašant system32 katalogą.
	4:28	**8**	**VASS pasirašant visą VM.**
	9:25	**0,97**	**VASS pasirašant system32 katalogą + EFS.**
	5:01	0,97	EFS šifruojant system32 katalogą.
	13:00	**8**	**BitLocker.**
VM20-8	4:20	0,97	VASS pasirašant system32 katalogą.
	6:23	**8**	**VASS pasirašant visą VM.**
	9:22	**0,97**	**VASS pasirašant system32 katalogą + EFS.**
	4:96	0,97	EFS šifruojant system32 katalogą.
	17:10	**8**	**BitLocker.**
VM20-16	4:15	0,97	VASS pasirašant system32 katalogą.
	10:35	**16**	**VASS pasirašant visą VM.**
	9:17	**0,97**	**VASS pasirašant system32 katalogą + EFS.**
	5:02	0,97	EFS šifruojant system32 katalogą.
	22:56	**16**	**BitLocker.**

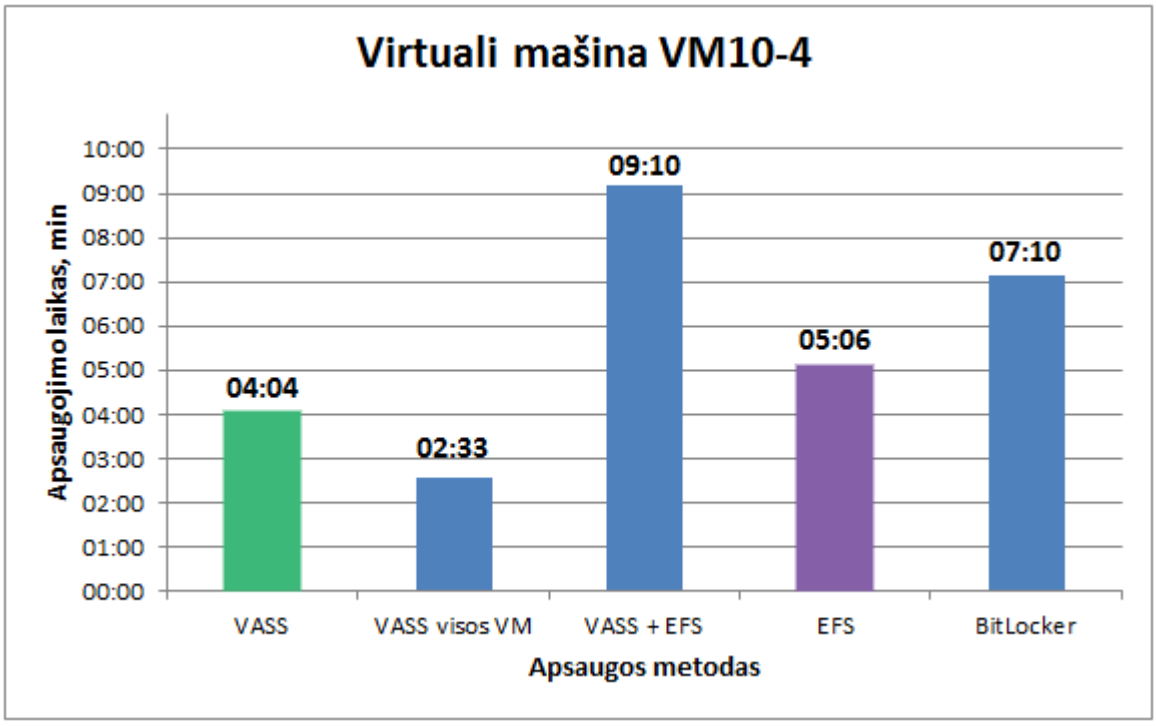
`

Analizuodami gautus rezultatus galime daryti išvadą, jog šifruojant virtualias mašinas Microsoft Windows BitLocker paslauga ar naudojantis VASS bei pasirašant visą VM, apsaugojimo laikas tiesiogiai priklauso nuo virtualios mašinos dydžio. Tuo tarpu pasirašant jautrų operacinės sistemos katalogą VASS, pasirašant jautrų operacinės sistemos katalogą VASS, o pasirašytą katalogą dar užšifruojant Microsoft Windows EFS paslauga ar naudojantis tik EFS, virtualių mašinų apsaugojimo laikas nepriklauso nuo VM dydžio kadangi didėjant virtualiai mašinai jos jautraus katalogo dydis nekinta.

Grafiniai rezultatai

Visose diagramose skirtingomis spalvomis išskiriami saugumo profiliai. Didelį saugumo profilį užtikrinantys metodai vaizduojami mėlyna spalva, vidutinį – žalia, žemą – violetine.

Saugant virtualią mašiną VM10-4, kurios savasis dydis 10GB, o failų sistemos dydis 4GB, akivaizdžiai greičiausiai virtualią mašiną apsaugojo VASS pasirašant visą VM (pav. 21). Antras bei trečias pagal greitumą metodai buvo VASS pasirašant jautrų operacinės sistemos katalogą ir EFS šifruojant jautrų operacinės sistemos katalogą. Akivaizdu, kad didelis failų kiekis esantis kataloge užima daugiau laiko kontrolinės sumos skaičiavimo algoritme ar failų sistemos šifravime EFS nei viso virtualaus kieto disko kontrolinės sumos suskaičiavime. Pastebime, jog optimaliausiai virtualią mašiną VM10-4 apsaugo VASS pasirašant visą virtualią mašiną. Šis metodas ne tik veikė greičiausiai, bei ir yra priskiriamas prie didelį saugumo profilį užtikrinančių metodų.

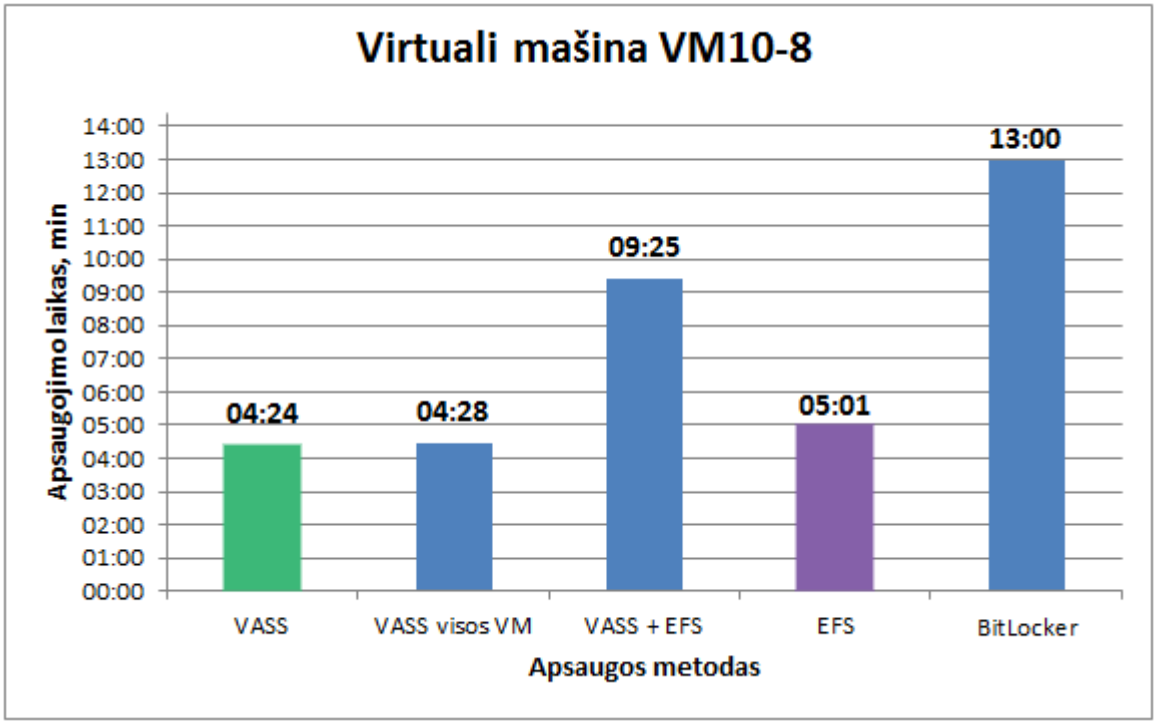


Pav. 21. Virtualios mašina VM10-4

`

Saugant virtualią mašiną VM10-8, kurios savasis dydis 10GB, o failų sistemos dydis 8GB, beveik identišku greičiu virtualią mašiną apsaugojo VASS pasirašant jautrų operacinės sistemos katalogą bei VASS pasirašant visą VM (pav. 22). Pastebima, jog padidėjęs virtualios mašinos savasis dydis sulėtino VASS pasirašant visą VM metodo veikimą, o tuo tarpu VASS pasirašant jautrų katalogą veikimo laikas beveik nepakito. Taip pat nepakito ir VASS + EFS šifravimo laikas, tačiau jis vis dar veikė lėčiau nei VASS pasirašant visą VM.

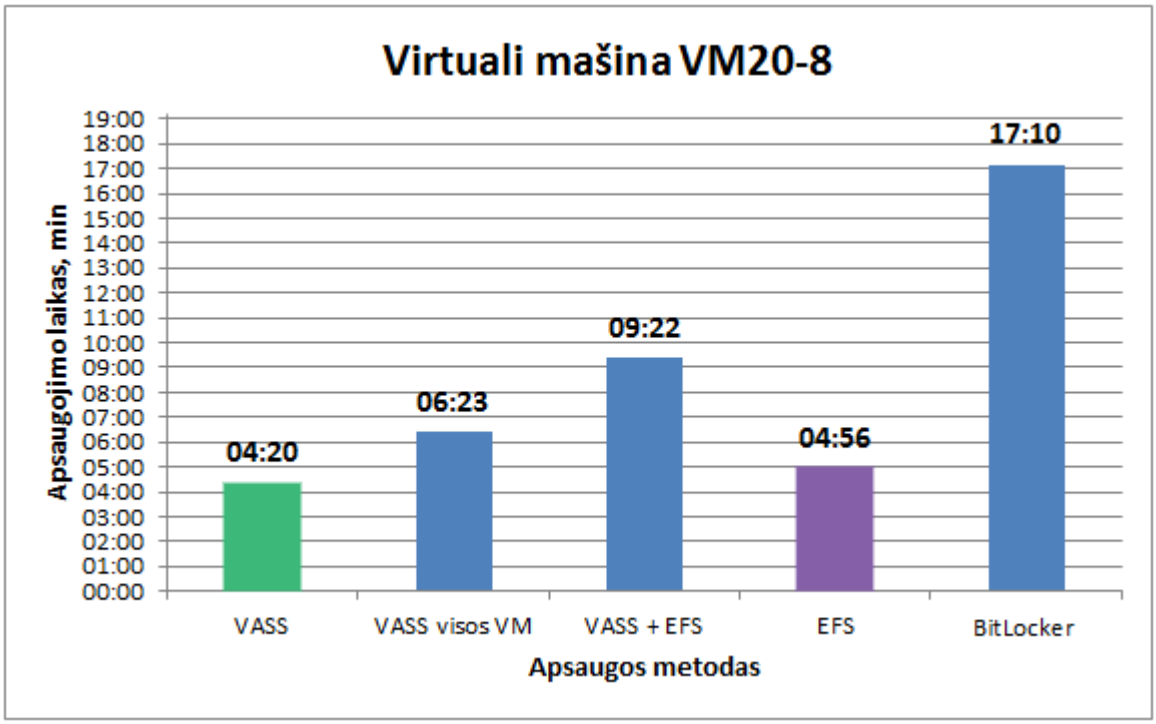
Priklausomai nuo siekiamo saugumo profilio bei planuojamų laiko sąnaudų galima išskirti optimaliausiai tinkantį metodą. Greičiausiai virtualią mašiną VM10-8 apsaugojo VASS pasirašant jautrų operacinės sistemos katalogą, tačiau šis metodas priskiriamas prie vidutinį saugumo profilį užtikrinančių metodų. Didelį saugumo profilį užtikrinančių metodų tarpe vėl greičiausiai virtualią mašiną apsaugojo VASS pasirašant visą VM.



Pav. 22 Virtuali mašina VM10-8

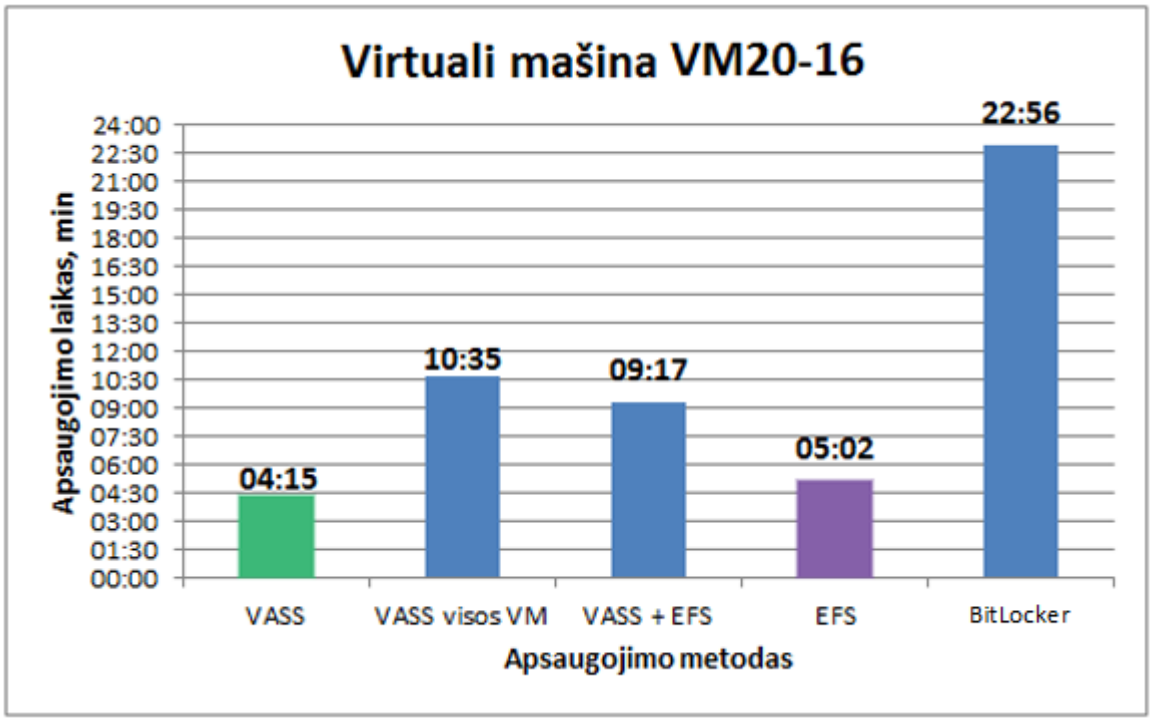
Analizuodami virtualios mašinos VM20-8 (pav. 23), kurios savasis dydis 20GB, o failų sistemos dydis 8GB, apsaugojimo rezultatus pastebime, jog vėl priklausomai nuo siekiamo saugumo profilio bei planuojamų laiko sąnaudų, optimaliausiais skiriami skirtingi metodai - VASS pasirašant jautrų operacinės sistemos katalogą ir VASS pasirašant visą VM. Vienas virtualią mašiną apsaugojo greičiausiai, bet yra priskiriamas vidutiniškai saugaus profilio kategorijai, o kitas veikė lėčiau, bet yra priskiriamas saugių profilio kategorijai.

`



Pav. 23 Virtuali mašina VM20-8

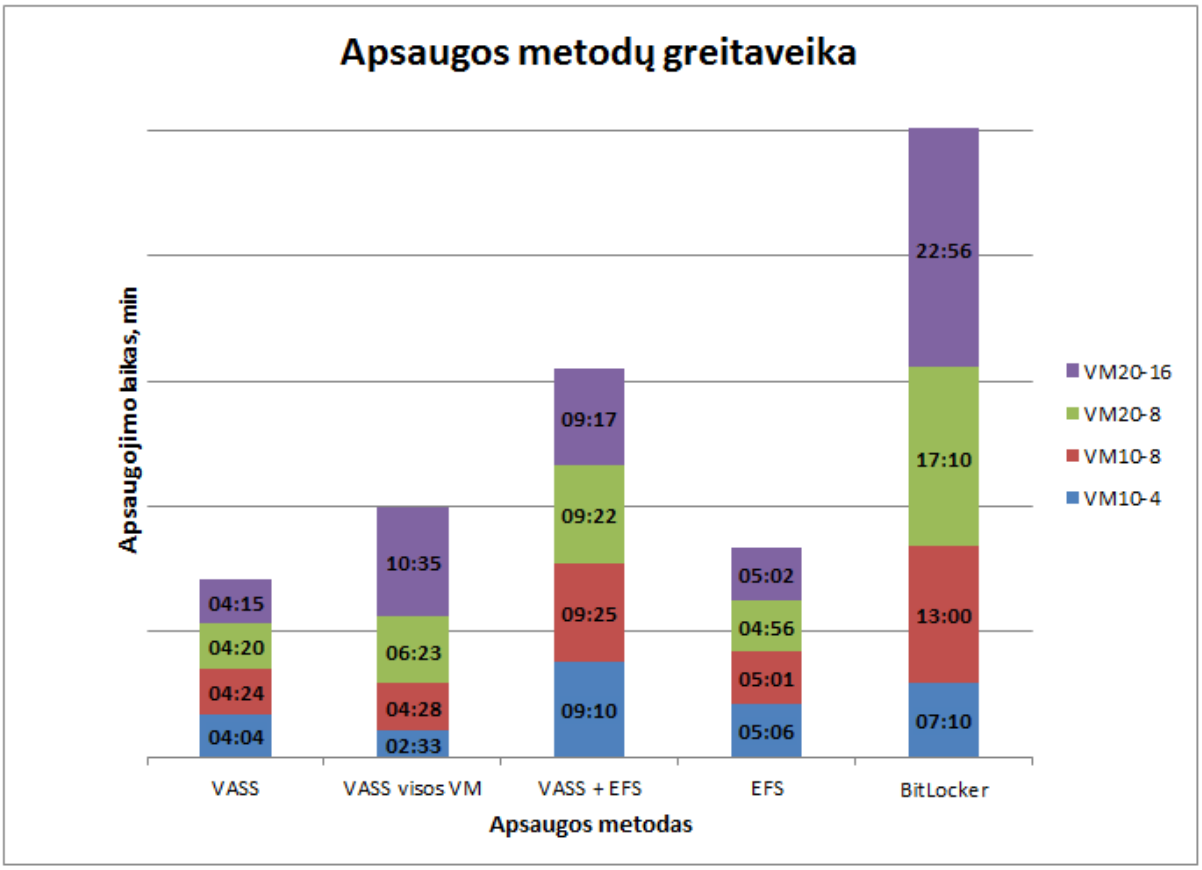
Pastebime, jog apsaugant virtualią mašiną VM20-16, kurios savasis dydis 20GB, o failų sistemos dydis 16GB, padidėjęs virtualios mašinos savasis bei failų sistemos dydis įtakojo saugiam profiliui priskiriamų metodų greitį. Čia greičiausiai, saugiam profiliui priskiriamų metodų tarpe, virtualią mašiną apsaugojo VASS + EFS metodas. Vėl absoliučiai greičiausiai, bet priskiriamas tik vidutiniškai saugių profilių kategorijai, virtualią mašiną apsaugojo VASS pasirašant jautrų operacinės sistemos katalogą.



Pav. 24 Virtuali mašina VM20-16

`

Visų matavimų suminį rezultatą pateikiame 25 paveiksle. Čia kiekvienos virtualios mašinos apsaugojimo laikas vaizduojamas skirtinga spalva, ko pasėkoje galime matyti VM dydžio bei apsaugojimo laiko santykį.



Pav. 25 Apsaugos metodų greitaveika

Analizuodami 25 paveikslą galime teigti, jog virtualios mašinos apsaugos laikas tiesiogiai priklauso nuo VM dydžio, apsaugos metodo bei siekiamo saugumo lygio. Mažiausią t.y. 10GB savojo bei 4GB failų sistemos dydžio virtualią mašiną greičiausiai apsaugojo VASS pasirašant visą VM. Be to šis metodas priskiriamas saugiam profilio tipui, todėl laiko ir saugos atžvilgiu yra optimaliausias. Šis metodas virtualią mašiną apsaugojo per 2:33 minutės, kas yra 1,7 karto greičiau nei antroje vietoje likęs VASS pasirašant jautrų operacinės sistemos katalogą.

Didžiausią t.y. 20GB savojo bei 16GB failų sistemos dydžio virtualią mašiną greičiausiai, per 4:15 minutės, apsaugojo VASS pasirašant jautrų operacinės sistemos katalogą, tačiau šis metodas priskiriamas tik vidutiniam saugumo profiliui, todėl siekiant užtikrinti didelį saugumo lygį, nėra tinkamas. Saugiam profiliui priskiriamų metodų tarpe greičiausiai, per 9:17 minutės, virtualią mašiną apsaugojo VASS pasirašant jautrų operacinės sistemos katalogą, o pasirašytą katalogą šifruojant Microsoft Windows EFS paslauga. Šis metodas 1:18 minutės virtualią mašiną apsaugojo greičiau nei antroje vietoje likęs VASS pasirašant visą virtualią mašiną.

`

3.3. Išvados

Atlikto virtualios aplinkos saugos sistemos prototipo eksperimentinio tyrimo metu nustatyta:

- ✓ Didėjant virtualių mašinų dydžiui proporcingai didėjo VASS pasirašant visą VM bei viso disko šifravimo laikas. Kitų apsaugos metodų veikimo, virtualios mašinos dydis, neįtakojo, nes didėjant virtualiai mašinai, pasirašomo jautraus katalogo dydis nekito.
- ✓ VASS pasirašant visą VM ir Microsoft Windows BitLocker paslauga bereikalingai apsaugo tuščia disko talpą, nes VM10-8 ir VM20-8, kurių failų sistemų dydžiai vienodi, apsaugomos skirtingu greičiu. Pasirašant virtualias mašinas VASS metodu jų saugojimo laikas pailgėjo 1:55 minutės, kas yra 1,3 karto, o šifruojant virtualias mašinas Microsoft Windows BitLocker paslauga jų saugojimo laikas pailgėjo 4:10 minutės, kas taip pat yra 1,3 karto.
- ✓ Remiantis eksperimento rezultatais galime daryti išvadą jog didesnes nei 20 GB virtualias mašinas optimaliausiai apsaugo VASS pasirašant jautrų operacinės sistemos katalogą, o pasirašytą katalogą užšifruojant Microsoft Windows EFS paslauga.

`

4. IŠVADOS

- ✓ Nereguliarus virtualių mašinų naudojimas apsunkina antivirusinės programinės įrangos licencijavimo tvarkos užtikrinimą ir palaikymą.
- ✓ Virtualių mašinų turinio saugojimas atviro formato failuose, piktavaliams suteikia galimybes įnešti nepageidaujamą programinę įrangą į pasyvioje būsenoje esančias virtualias mašinas.
- ✓ Suprojektuotas ir realizuotas virtualios aplinkos saugos sistemos prototipas pagrįstas kontrolinių sumų skaičiavimu ir failų sistemos šifravimu EFS, leidžiantis, prieš virtualios mašinos paleidimą, atlikti jos disko atvaizdo tikrinimą dėl piktavališkos programinės įrangos.
- ✓ Suprojektuota ir realizuota reliacinė duomenų bazė virtualių mašinų telkinio pasirašymo informacijai saugoti ir valdyti.
- ✓ VASS pasirašant visą VM ir Microsoft Windows BitLocker paslauga bereikalingai apsaugo tuščia disko talpą, nes VM10-8 ir VM20-8, kurių failų sistemų dydžiai vienodi, apsaugomos skirtingu greičiu. Pasirašant virtualias mašinas VASS metodu jų saugojimo laikas pailgėjo 1:55 minutės, kas yra 1,3 karto, o šifruojant virtualias mašinas Microsoft Windows BitLocker paslauga jų saugojimo laikas pailgėjo 4:10 minutės, kas taip pat yra 1,3 karto.
- ✓ Remiantis eksperimento rezultatais galime daryti išvadą jog didesnes nei 20 GB virtualias mašinas optimaliausiai apsaugo VASS pasirašant jautrų operacinės sistemos katalogą, o pasirašytą katalogą užšifruojant Microsoft Windows EFS paslauga.

`

5. LITERATŪROS SĄRAŠAS

1. **Lunsford D. L. //** Virtualization Technologies in Information Systems Education // Journal of Information Systems Education, Vol. 20(3) School of Accountancy and Information Systems The University of Southern Mississippi, 730 East Beach Blvd. Long Beach, MS 39560, 2009
2. **Daniel J.** // Server Virtualization Architecture and Implementation // Magazine Crossroads Volume 16 Issue 1, New York, USA, 2009
3. **Brilingaitė A., Kybartas R. //** Programavimas debesų kompiuterijos (cloud computing) aplinkoje // Mokomoji knyga, e-ISBN 978-609-433-082-7, Kauno technologijos universitetas, Kaunas, 2011
4. **Scarfone K., Souppaya M., Hoffman P.** // Guide to Security for Full Virtualization Technologies // Computer Security Division Information Technology Laboratory National Institute of Standards and Technology Gaithersburg, MD 20899-8930, 2011
5. **Ray E., Schultz E.** // Virtualization Security // Proceedings of the 5th Annual Workshop on Cyber Security and Information Intelligence Research, Design & Consulting, Inc. 826 North Red Robin Street Orange, CA 92869, 2009
6. **Tolnai A., Solms S.** // A Virtualized Environment Security (VES) Model for a Secure Virtualized Environment // Internet Technology and Secured Transactions (ICITST), International Conference for, University of Johannesburg, South Africa, 2010
7. **Gebhardt C., Tomlinson A.** // Security consideration for virtualization // Technical Report RHUL-MA-2008-16, Department of Mathematics Royal Holloway, University of London Egham, 2008
8. **Roberts, J., and Yacono, J. //** Server Virtualization Offers Many Opportunities // CRN, No. 1076, 2003
9. **Vaughan-Nichols S.J.** // Virtualization Sparks Security Concerns // Journal Computer Volume 41 Issue 8, Freelance Technol., Mills River, NC, 2008
10. **Garfinkel T., Rosenblum M.** // When Virtual is Harder than Real: Security Challenges in Virtual Machine Based Computing Environments // Stanford University Department of Computer Science, Berkeley, CA, USA, 2005
11. **King S.T., Chen P.M.** // SubVirt: implementing malware with virtual machines // Proceeding SP '06 Proceedings of the 2006 IEEE Symposium on Security and Privacy // Michigan University, 2006
12. **Ahmad D., Arce I. //** Ghost in the Virtual Machine // Security & Privacy, IEEE, Volume: 5, Issue: 4, 2007

`

13. **Sala G., Sgandurra D., Baiardi F. //** Security and Integrity of a Distributed File Storage in a Virtual Environment // Fourth International IEEE Security In Storage Workshop, University of Pisa, 2007
14. **Karger P. A. //** Is Your Virtual Machine Monitor Secure? **//** Third Asia-Pacific Trusted Infrastructure Technologies Conference, IBM Corporation, Thomas J. Watson Research Center, Yorktown Heights, NY 10598, USA, 2008
15. **Sakalaukas E., Listopadskis N., Dosinas G. S., Katvickis A.** // Kriptografinės sistemos // Mokomoji knyga, ISBN 978-9955-686-76-7, Kauno technologijos universitetas, Kaunas, 2008
16. **Sivathanu G., Wright C. P., Zadok E. //** Enhancing File System Integrity Through Checksums // Technical Report FSL-04-04, Stony Brook University, 2009
17. **Gauravaram P., Kelsey J., Knudsen L., Thomsen S.** // On hash functions using checksums // International Journal of Information Security, Volume 9 Issue 2, National Institute of Standards and Technology (NIST), 2010
18. **Liang M., Chang C. //** Full Disk Encryption based on Virtual Machine and Key Recovery Scheme // Journal of Information and Computing Science Vol. 6, No. 3, Institute of Electronic Technology, Information Engineering University, Zhengzhou China, 2010
19. **Chen S., Jin C.** // An Improved Collision Attack on MD5 Algorithm // Information Security and Cryptology, Institute of Electronic Technology, the University of Information Engineering, Zhengzhou, China, 2008
20. **Mendel F., Rechberger C. //** Collisions for 70-step SHA-1: on the full cost of collision search // SAC'07 Proceedings of the 14th international conference on Selected areas in cryptography, Berlin, 2007
21. **NESSIE consortium //** Portfolio of recommended cryptographic primitives // NESSIE Consortium, 2003
22. **Ferguson N. //** AES-CBC + Elephant diffuser A Disk Encryption Algorithm for Windows Vista // Microsoft corporation, 2006
23. **Li S., Jis X.** // Research and Application of Transparent Encrypting File System Based on Windows Kernel // Computational Intelligence and Software Engineering (CiSE), 2010 International Conference on, Technology, Guizhou University, China, 2010
24. **Wei B.** // Provable security of the modified encrypting file system for microsoft Windows // ICICS'09 Proceedings of the 7th international conference on Information, communications and signal processing, New Jersey, 2009

`

25. Understanding the files that make up a VMware virtual machine. [Žiūrėta: 2012-03-05]. Prieiga per internetą: <http://searchvmware.techtarget.com/tip/Understanding-the-files-that-make-up-a-VMware-virtual-machine>
26. Virtualization Decisions 2011 Purchasing Intentions Survey. [Žiūrėta: 2012-03-05]. Prieiga per internetą: <http://searchservervirtualization.techtarget.com/Virtualization-Decisions-2011>
27. Failų apsaugojimas naudojant „BitLocker“ disko šifravimą. [Žiūrėta: 2012-03-16]. Prieiga per internetą: <http://windows.microsoft.com/lt-LT/windows7/Help-protect-your-files-using-BitLocker-Drive-Encryption>
28. Koks skirtumas tarp „BitLocker“ disko šifravimo ir Koduojamųjų failų sistemos?. [Žiūrėta: 2012-03-16]. Prieiga per internetą: < http://windows.microsoft.com/lt-LT/windows7/Whats-the-difference-between-BitLocker-Drive-Encryption-and-Encrypting-File-System>

`

6. SUMMARY

Virtualization is technology allowing computer hardware apply as it was computer software. This technology enables to run several operational systems on the same machine simultaneously. Usually single operational system controls all hardware recourses of the technical platform. Virtualized system expends existing operational system with the hypervisor layer which provides multiple instances for all operational system and creates virtual machine.

Virtual machine is software application where operational system and programs can be installed in the same manner as it can be done on the computer hardware. The virtual machine in turned off mode is only a file. This file can be located to the separated partition using virtual platform or third parties programs and can be browsed as in ordinary computer file system. This functionality opens a weak spot because there are no means to ensure that when virtual machine is off no system files will be modified.

VASS safety system for virtual environment is design and specified in this paper work. VASS provides possibility to ensure main safety goals such as confidentiality, integrity and accessibility. Integrity of the virtual machine is achieved using file content control sum calculation algorithm based on SHA-256. This algorithm is recommended by NESSIE as sustainable and resistant to collisions. Files system decoding EFS is used to ensure confidentiality of virtual machine.

During the practical study four virtual machines were monitored. Five different methods were applied to ensure security in all virtual machines. Graphical and computational study results are presented in this work. Security profile is applied to every method. Security profile is defined by information security criteria and the actual impact of the particular criterion to the designed system. The result of the study revealed that security time for virtual machine depends on the applicable security profile, planned time limits and size of the virtual machine.

`

7. SANTRUMPŲ IR TERMINŲ ŽODYNAS

Terminas	Paaiškinimas
Virtuali mašina	Programinė techninės įrangos realizacija, kuri leidžia vykdyti programas taip pat kaip ir aparatinė mašina.
Hipervizorius	Programinės įrangos abstraktusis sluoksnis hipervizorius, kuris įterpiamas tarp aparatinės įrangos ir pagrindinės operacinės
Pagrindinė operacinė sistema	Pagrindinių kompiuterio ar serverio operacinė sistema.
Svečio operacinė sistema	Operacinė sistema kurią vykdo virtuali mašina.
VDI (angl. Virtual Desktop Infrastructure)	Sprendimas, kuris leidžia darbo vietos procesams vykti centralizuotoje bei virtualizuotoje duomenų centro aplinkoje, prie kurios vartotojas jungiasi naudodamas savo darbo vietoje esančią įrangą.
FDE (ang. full disk encryption)	Viso disko šifravimo metodas.
TPM (angl. trusted platform module)	TPM yra mikrolustas, sukurtas pateikti pagrindines su sauga susijusias funkcijas. TPM paprastai diegiamas kompiuterio pagrindinėje plokštėje ir susisiekia su likusia sistemos dalimi naudodamas aparatūros magistralę.
EFS (ang. encrypting file system)	Failų sistemos šifravimo metodas.
FEK (angl. file encryption key)	Generuoja atsitiktinis skaičius.
VASS	Virtualios aplinkos saugos sistema.