KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
KOMPIUTERIŲ KATEDRA

Giedrius Tamašauskas

Saugi „Apache“ žiniatinklio serverių failų sistema

Magistro darbas

Darbo vadovas:
Algimantas Venčkauskas

Kaunas, 2010

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
KOMPIUTERIŲ KATEDRA

Giedrius Tamašauskas

Saugi „Apache“ žiniatinklio serverių failų sistema

Magistro darbas

Recenzentas

doc. dr. G. Ziberkas
2010-05-26

Vadovas

doc.dr. A. Venčkauskas
2010-05-26

Atliko

IFN-8/3 gr. stud.
Giedrius Tamašauskas
2010-05-26

Kaunas, 2010

Secure „Apache“ WEB server file system

Summary

“Apache” is the one of the most popular web server. The main objective of this work is to create a secure file system for “Apache” web server. The paper analyzes the “Apache” Web files and other safety systems. This paper points out and explores the security limitations of existing server file system. For the “Apache” Web files system to be more secured it was added four additional properties: journaling, secure file deletion, data encryption during storage and flexible user access management. The paper implemented the new user graphical interface, which provides safe features for the file system to increase the protection of the owner. New graphic interface allows the authorized owner to be more flexible and faster in securing data managing in this file system, eliminating the need of command line and editing configurations files manually.

Saugi „Apache“ žiniatinklio serverio failų sistema

Santrauka

„Apache“ yra vienas iš populiariausių žiniatinklio serverių. Pagrindinis šio darbo tikslas sukurti saugią „Apache“ žiniatinklio serverio failų sistemą. Darbe analizuojant „Apache“ žiniatinklio ir kitų failų sistemų saugos galimybes nustatyta, kad šio serverio failų sistemos saugos galimybės yra ribotos. Kad „Apache“ žiniatinklio failų sistema būtų saugi pridedamos keturios papildomos savybės, tai žurnaliavimas, saugus failo ištrynimas, duomenų šifravimas jų saugojimo metu, bei lankstesnis vartotojų prieigos valdymas. Darbe realizuota vartotojo grafinė sąsaja, kurios dėka suteikiamos saugios failų sistemos savybės savininko duomenims apsaugoti. Realizuojamoje grafinėje sąsajoje duomenų savininkas lanksčiau ir aiškiau suteikia saugios failų sistemos savybes savo duomenims, neįvedinėdamas komandinių eilučių ir neredaguodamas konfigūracinių failų.

Įvadas

Visuotinės kompiuterizacijos laikais duomenų sauga tampa vis svarbesne užduotimi kompiuterių saugos specialistams. Viena iš saugos sričių yra žiniatinklio serverių apsauga. Šiuose serveriuose saugomi svarbūs duomenys, kuriems reikia garantuoti apsaugą ir neleisti, kad piktavalis asmuo vienokiu ar kitokiu būdu pakenktų šiai informacijai.

„Apache“ yra žiniatinklio serveris, naudojamas kompiuterių prisijungimui prie Interneto. Žiniatinklio serverio funkcija yra aptarnauti Interneto naršyklės pasiųstas užklausas – rasti tam tikro turinio dokumentus. [4]

Šiame darbe analizuojamos „Apache“ žiniatinklio serverių failų sistemos saugos galimybės, aptariamos problemos, su kuriomis susiduria informacijos prižiūrėtojai, norėdami nustatyti saugos parametrus. Taip pat analizuojamos analoginių failų sistemų saugos galimybės ir padarius išvadas projektuojama saugi „Apache“ žiniatinklio serverio failų sistema. Realizacijoje informacijos savininkui sukuriama grafinė sąsaja, kurios pagalba „Apache“ žiniatinklyje savo talpinamai informacijai savininkas gali paprasčiau suteikti saugios failų sistemos savybes.

Tyrimo objektas ir problema

Tiriamas „Apache“ žiniatinklio serveris ir jo failų sistemos saugos galimybės. Norima nustatyti ar „Apache“ žiniatinklio serverio failų sistemos saugos galimybės yra ribotos lyginant su kitomis failų sistemomis ir kokių savybių trūksta, kad būtų galima užtikrinti vartotojų duomenų saugumą.

Pagrindinę problemą galima aprašyti dviem punktais:

1. Saugi „Apache” žiniatinklio serverių failų sistema. Svarbu išsiaiškinti ir suprasti kokios turi būti saugios „Apache“ žiniatinklio serverio failų sistemos savybės, kad būtų galima užtikrinti vartotojo duomenų saugumą.

2. Sudėtingas „Apache” žiniatinklio serverio saugos parametrų nustatymas. Vartotojui su minimaliom žiniom, turi būt lanksčiai ir aiškiai suteikta galimybė naudotis saugios failų sistemos teikiamomis funkcijomis. Daugybė komandinių eilučių ir konfigūravimas tekstinių failų reikalauja didelių žinių ir dažnai iššaukia nepilnavertį saugos parametrų išnaudojimą, tokiu atveju iškyla grėsmė serveryje saugomiems duomenims.

1. „APACHE“ ŽINIATINKLIO IR KITŲ FAILŲ SISTEMŲ SAUGOS GALIMYBIŲ ANALIZĖ

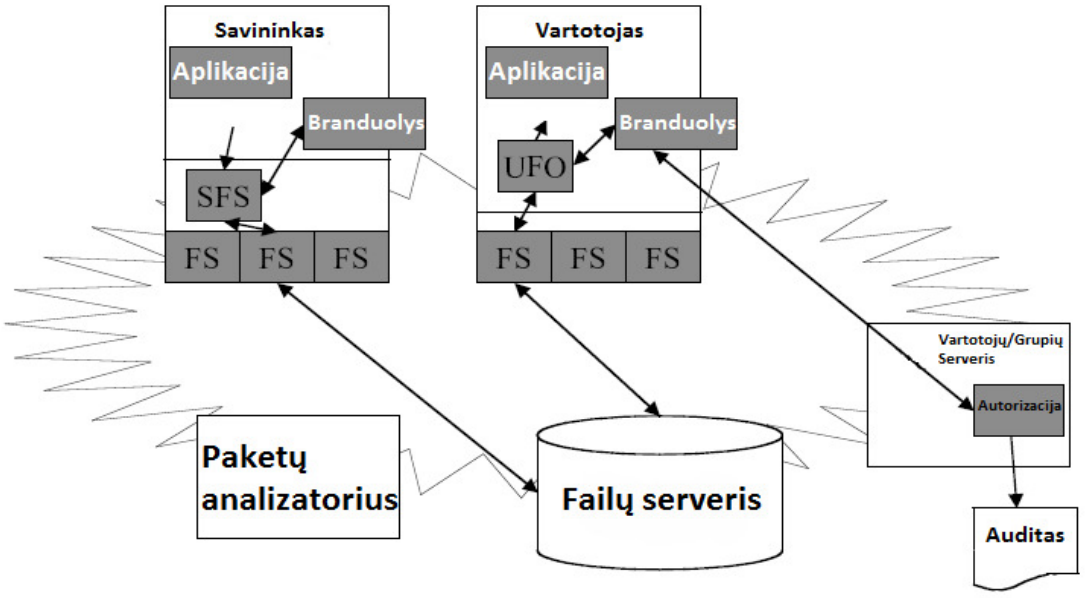
Analizuojamos „Apache“ žiniatinklio ir kitų failų sistemų saugos galimybės, nustatoma kokių saugos savybių neturi „Apache“ žiniatinklio serverio failų sistema, kad failų sistema buti saugi.

1.1. FAILŲ SISTEMŲ SAUGOS GALIMYBIŲ ANALIZĖ

Analizuojamos esamos failų sistemos, saugios failų sistemos modeliai. Peržvelgiama kokios savybės yra realizuojamos norint, kad failų sistema būtų saugi.

1.1.1. Saugios failų sistemos architektūra

Du duomenų laikmenų korporacijos specialistai James P. Hughes ir Christopher J. Feist moksliniame straipsnyje aprašė saugios failų sistemos (SFS) modelį. Kuriame naudojama “nuo pradžios iki galo” (end-to-end) užšifravimo palaikymas vartotojams, gaunantiems prieigą prie rinkmenų per bet kokį tinklą. Ši architektūra susideda iš prieigos kontrolės sąrašo, protingos kortelės (smart card) autentiškumo nustatymo, vartotojų/grupių serverio ir saugios failų sistemos (SFS) kliento.[10]



1 pav. Saugios failų sistemos architektūra pagal James P. Hughes ir Christopher J. Feist

Ši modeliuojama saugi failų sistema taip pat leidžia decentralizuotas prieigos teises, kur mažos žmonių grupės gali apibrėžti, kam leidžiama pažiūrėti į tam tikrą informaciją (be sistemos administratoriaus, kuriuo nepasitikima pagalbos) ir kokios nustatytos taisyklės privalo tai apibrėžti. Apibrėžiamos tokie vartotojai:

- **Informacijos savininkas**— turi duomenis ir visas teises kaip elgtis su duomenimis ir gali nustatyti, kas gali "pamatyti" jų duomenis.
- **Informacijos vartotojas** — turi poreikį tam tikriems duomenims ir turi (nepaneigiamą) gebėjimą perduoti informaciją.
- **Grupės atstovas** — apibrėžtas grupės narys, kuris taip pat gali pasirūpinti auditu.

Informacija turi būti pranešta tarp informacijos savininko ir vartotojo, išskyrus grupės atstovui. Jokiam kitam asmeniui neleidžiama "pamatyti" atviros neužšifruotos informacijos. Informacijos nuosavybė yra pagrįsta organizacijomis ir jų įgaliojimu: informacijos savininkas ir vartotojai gali dinamiškai apibrėžti ir sukurti savo grupes informacijos pasidalinimui.

- ✓ “nuo pradžios iki galo” (end-to-end) šifravimas — kitos sistemos užšifruoja sąsaja su tinklo serveriu, bet saugioje failų sistemoje užšifruojama nuo informacijos savininko iki informacijos vartotojo.
- ✓ Paskirstytas narių skaičius — apibrėžtas vartotojų, ne sistemos administratorių.
- ✓ Talpyklų šifravimas — atsarginės kopijos, archyvai, tinklai, ir visa kita, viskas apsaugota.
- ✓ Audito tyrimas — failų prieigos užklausos yra prižiūrimos grupės atstovo.
- ✓ Priverstina kriptografija — stiprus užšifravimas naudojamas informacijai apsaugoti, taip pat protingoms kortelėms (smart cards) bei naudojama PKI (Public Key Infrustucture - viešųjų raktų infrastruktūra) apsaugoti ir valdyti raktams.

1.1.1.1. Prieigos kontrolės sąrašai

Prieigos kontrolės sąrašas leidžia savininkui tiksliai apibrėžti, kas turi prieigą prie jo duomenų. Tai apibrėžiama, naudojant XML (eXtensible Markup Language) ir leidžia keliems skirtingiems metodams apibrėžti prieigą. Toliau aprašomas sąrašas reikšminių žodžių ir jų apibrėžimų, kurie dalyvauja saugioje failų sistemoje prieigos kontrolės sąrašuose.

- ✓ Savininko grupė — grupė, kuri valdo failą.
- ✓ Rašytojas (writer) – failo sukūrėjas.

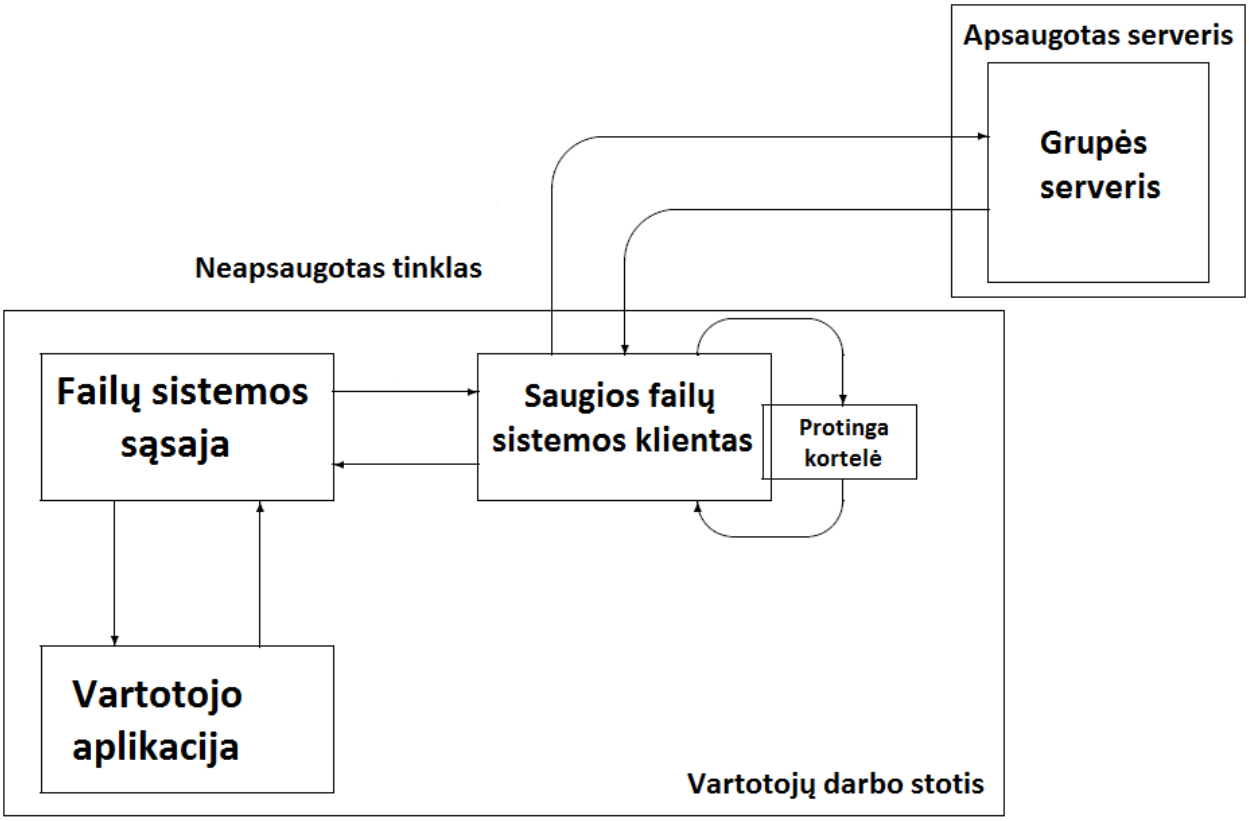
- ✓ ACL — Apibrėžia prieigos kontrolės sąrašo pradžią.
- ✓ Asmuo — apibrėžiamas asmuo, kuris turi prieigą prie failo.
- ✓ Grupė — Apibrėžia grupės serverį, kuris bus panaudotas, kad patikrintų grupės narių ryšius projekte.
- ✓ Projektas — apibrėžia projektą, kuris turi prieigą prie failo.
- ✓ Pagrindiniai duomenys — failų rakto užšifruotas tekstas , užšifruotas asmeniui ar grupės serveriui, priklausančiam nuo XML bloko.

1.1.1.2. Protingos kortelės (Smart cards)

Protingos Kortelės yra panaudotos saugioje failų sistemoje atskirų vartotojų autentiškumo nustatymui. Kiekviena protinga kortelė turi savyje mikroprocesorių ir mažą kiekį nekintamos tiesioginės kreipties atminties (RAM). Tai leidžia vartotojo privačiam raktui būti laikomam tiktai protingoje kortelėje ir niekur kitur. Kiekvieną kartą kai dokumentas ar raktas turi būti pasirašyti ar iššifruotas visos kriptografinės funkcijos reikalauja, kad protingoje kortelės privatus raktas būtų įvykdytas mikroprocesoriaus. Naudojant šį metodą privatus raktas niekada nepalieka protingos kortelės. Iš tikrųjų, net protingos kortelės savininkas niekada nepažins jo ar jos privataus rakto, žinos tiktai PIN, kuris atrakina šia kortelę.

1.1.1.3. Grupės serveris

Grupės Serveris yra vienintelis patikimas objektas šios modeliuojamos saugios failų sistemos architektūroje. Visi failų raktai yra užšifruoti grupės serverio viešuoju raktu ir laikomi kiekvieno užšifruoto failo antraštėje. Šioje antraštėje saugomas prieigos kontrolės sąrašas. Kai vartotojas prašo prieigos prie failo, antraštė yra persiunčiama grupės serveriui, kuris tada nustato, ar vartotojas turi prieigą prie failo. Jei vartotojas tikrai turi prieigą prie failo, failo raktas yra grąžinamas vartotojui, tokiu būdu jis sugeba iššifruoti duomenis. Grupės serveris taip pat tarnauja kaip vienas administracinis saugios failų sistemos architektūros punktas. Visi vartotojų pridėjimai ir pašalinimai iš grupių ir projektų yra atliekami grupės serverio. Audito tyrimas yra atliekamas grupės serverio, kuris gali detalizuoti, koks failas buvo atidarytas, kada buvo atidarytas, iš kur buvo atidarytas ir kas atidarė.

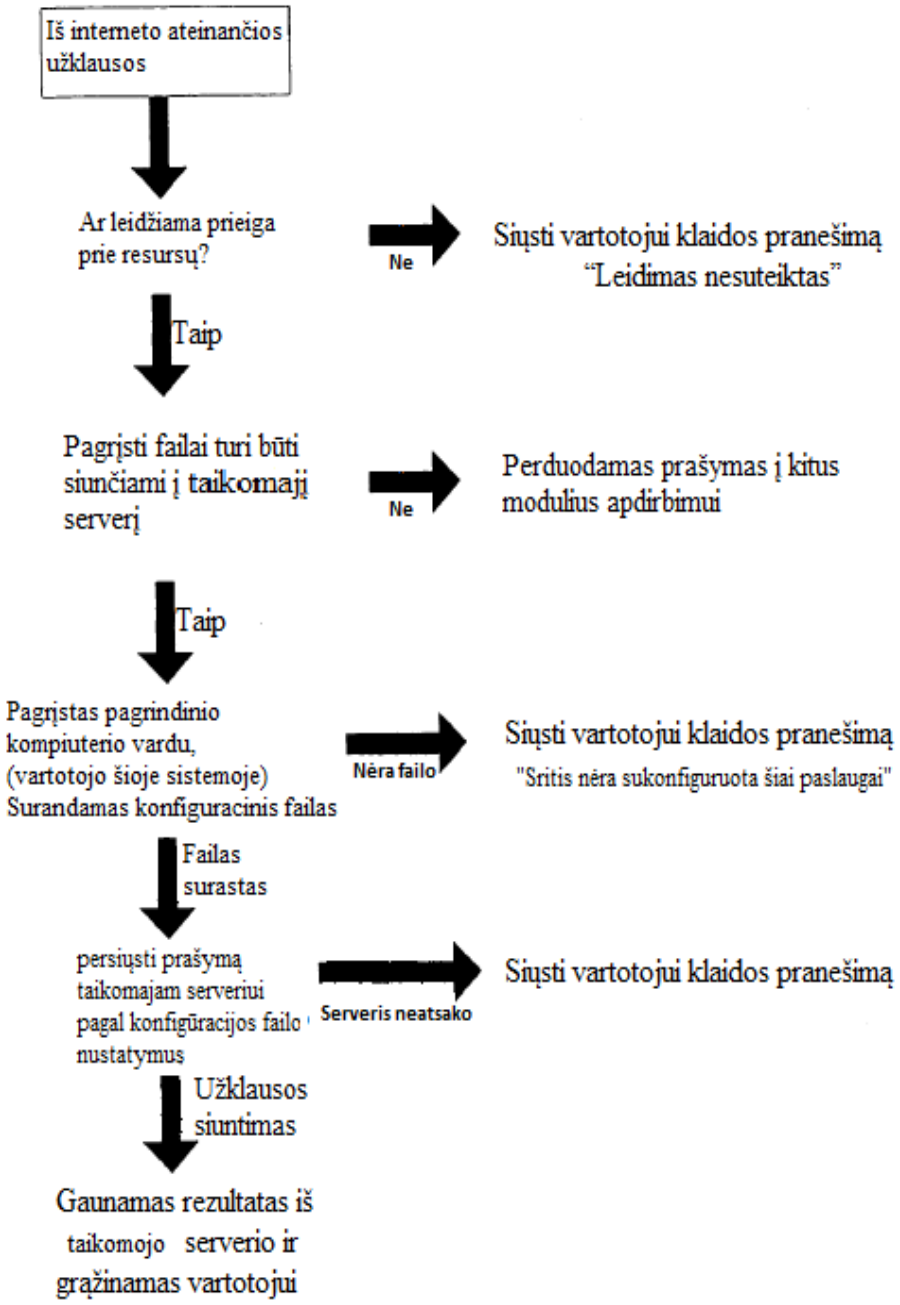


2 pav. failų prieiga

- ✓ Saugios failų sistemos klientas siunčia XML failą grupės serveriui su naujai sukurtu parašu, kad patikrinti ar suteikiamas leidimas. Gavus patvirtinimą, saugios failų sistemos klientas atšifruoja failą panaudojant failo raktą iš protingos kortelės ir tada gauna prieigą į failų sistemos sąsają.
- ✓ Failų sistemos sąsaja gauna nesaugomą failą, kuris atėjo iš saugios failų sistemos kliento ir perduoda tai į vartotojo aplikaciją.
- ✓ Vartotojo aplikacija gauna nesaugomą rinkmeną ir tęsia funkcionavimą, tarytum niekas neįvyko.

1.1.2. Prieiga prie žiniatinklio serverio failų sistemos resursų

2009 metų sausio mėnesio 13 dieną Jungtinėse valstijose užpatentuotas Bill F.Campbell ir Todd F.Burroughs metodas kaip sistema valdo į žiniatinklio serverį ateinančias užklausas, kuriomis bandoma prieiti prie tam tikrų sistemos failų. Žemiau pateikiamas metodo bendras vaizdas ir aprašymas.[6]



3 pav. schemiškas metodo vaizdas

Vartotojas siunčia užklausą gauti tam tikro turinio failą. Sistema nustato ar siunčiantis užklausą vartotojas turi teisę prieiti prie šių resursų. Jeigu sistemos žiniatinklio serveris nustato, kad vartotojas neturi leidimų prie sistemos resursų, žiniatinklio serveris siunčia klaidos pranešimą, informuodamas vartotoją, kad jis neturi teisių prieiti prie šių resursų. Jei žiniatinklio serveris nustato, kad užklausa turi būti praleidžiama, tuomet sistema praleidžia šią užklausą tolimesniam vykdymui.

Priklausomai nuo failo tipo nusprendžiama ar bus užklausa siunčiama į taikomąjį serverį ar bus nusiusta i kitus sistemos modulius tolimesniam apdirbimui.

Jei užklausa turi būti siunčiama toliau i taikomąjį serverį tuomet pirmiausia surandamas sistemoje konfigūracinis failas. Jei failas nerandamas, vartotojui siunčiamas klaidos pranešimas, kuris informuoja, kad sistema nėra sukonfigūruota teikti šią paslaugą.

Suradus konfigūracinį failą, pagal jame esančius nustatymus, persiunčiama užklausa i taikomąjį serverį. Jei surandamas konfigūracinis failas, bet dėl tam tikrų priežasčių neina pasiekti pačio taikomojo serverio, tuomet vartotojui siunčiamas klaidos pranešimas, kad šis serveris yra nepasiekiamas.

Persiuntus užklausą taikomajam serveriui gaunamas iš jo atsakymas ir nusiunčiamas vartotojui atsakymas.

1.1.3. Ext3 ir NTFS failų sistemos savybė – žurnaliavimas

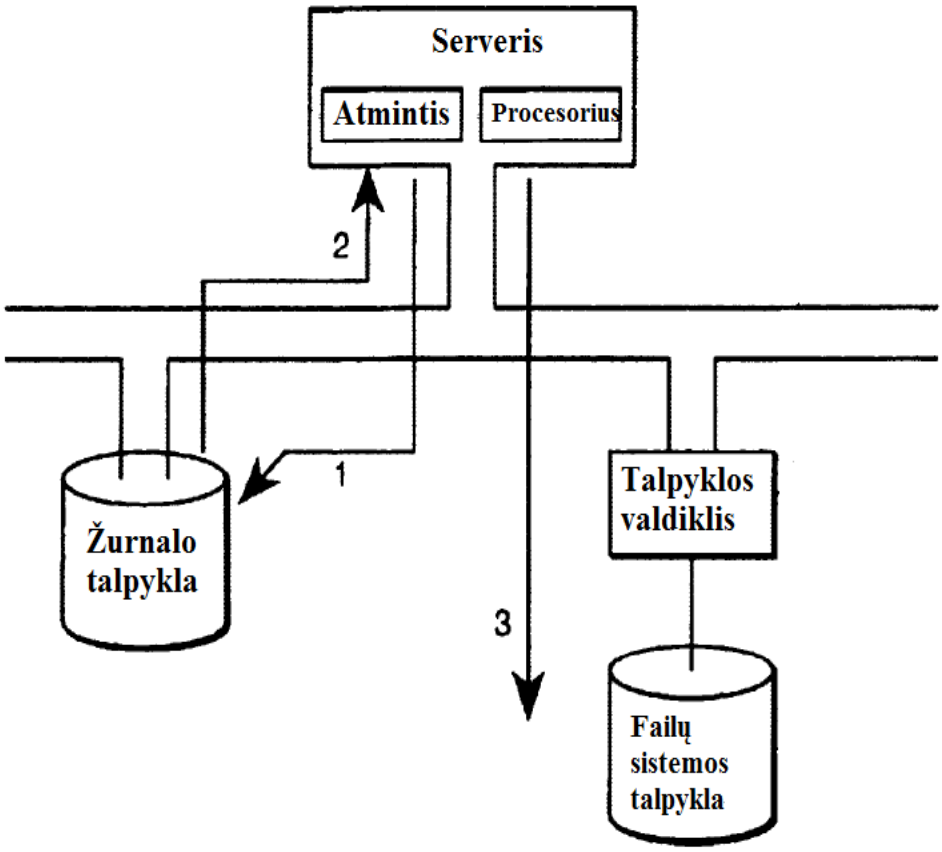
Žurnalizacija – visi vartotojų veiksmai sistemoje yra registruojami ir vėliau gali būti peržiūrėti. Žurnalizacijos failų sistema yra failų sistema, kuri saugo įrašus apie pakeitimus, kuriuos saugo žurnale (paprastai failų sistemoje, tam tikroje įrašams paskirtoje vietoje) prieš įrašant pakeitimus į pagrindinę failų sistemą. Jei sistema „lūžta“ arba dingsta maitinimas, tokios failų sistemos yra greitesnės, kad greičiau vėl pasijungtų į tinklą ir mažiau tikėtina, kad duomenys bus sugadinti.

1.1.3.1. Žurnalizacijos modelis

William P.Delaney ir Rodney A. DeKoning 2005 metų balandžio 5 dieną užpatentavo failų sistemos patobulinimo metodą, kuriame aprašomas failų sistemos žurnaliavimo savybė. Bendras modelio vaizdas matomas žemiau pateiktuose paveikslėliuose.

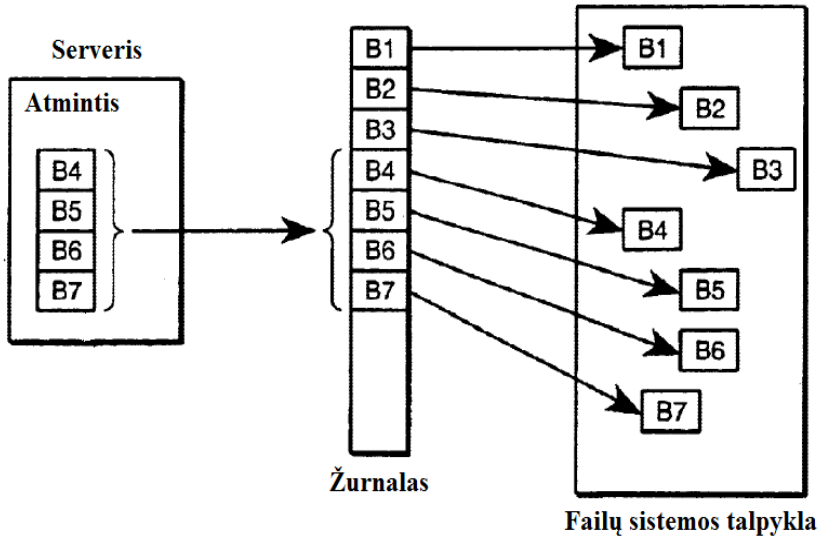
Modelio esmė yra ta, kad duomenys iš atminties prieš įrašant į failų sistemą įrašomi į žurnalą, o po to tik į pačią failų sistemą. Pagal žurnalo įrašus galima atkurti duomenis. Į žurnalą gali būt įrašomi tik metaduomenys arba metaduomenys ir rinkmenų struktūra, priklausomai nuo žurnaliavimo paskirties.[7]

Po sistemos „lūžimo“ tikrinami žurnalo paskutiniai įrašai prieš pat sistemai sugendant ir kokios transakcijos buvo atliktos ir patikrinama ar failų sistemoje yra įrašyti tie duomenys, jei ne tuomet labai greitai pagal žurnale esančius įrašus sistema atstatoma.



4 pav. žurnaliavimo modelis pagal William P.Delaney ir Rodney A. DeKoning

Žemiau pateiktame paveikslėlyje pavaizduojama kaip sistemoje keliauja duomenų fragmentai. Tai gali būti, kad ir metaduomenys. Matome, kad pirmiausiai duomenys laikomi serverio atmintyje ir duomenų fragmentai pirmiausiai įrašomi į žurnaliavimui paskirta vietą diske. Duomenys įrašomi iš eilės ir tikrinama ar žurnalas nepersipildė. Pagal žurnaliavimui skirtą talpyklos dydį priklausys, kada jis užsipildys. Po to kai įrašoma i žurnalą, duomenys saugomi failų sistemoje.



5 pav. žurnaliavimo veiksmų eiga

1.1.4. NTFS failų sistema

1 lentelė. NTFS failų sistema

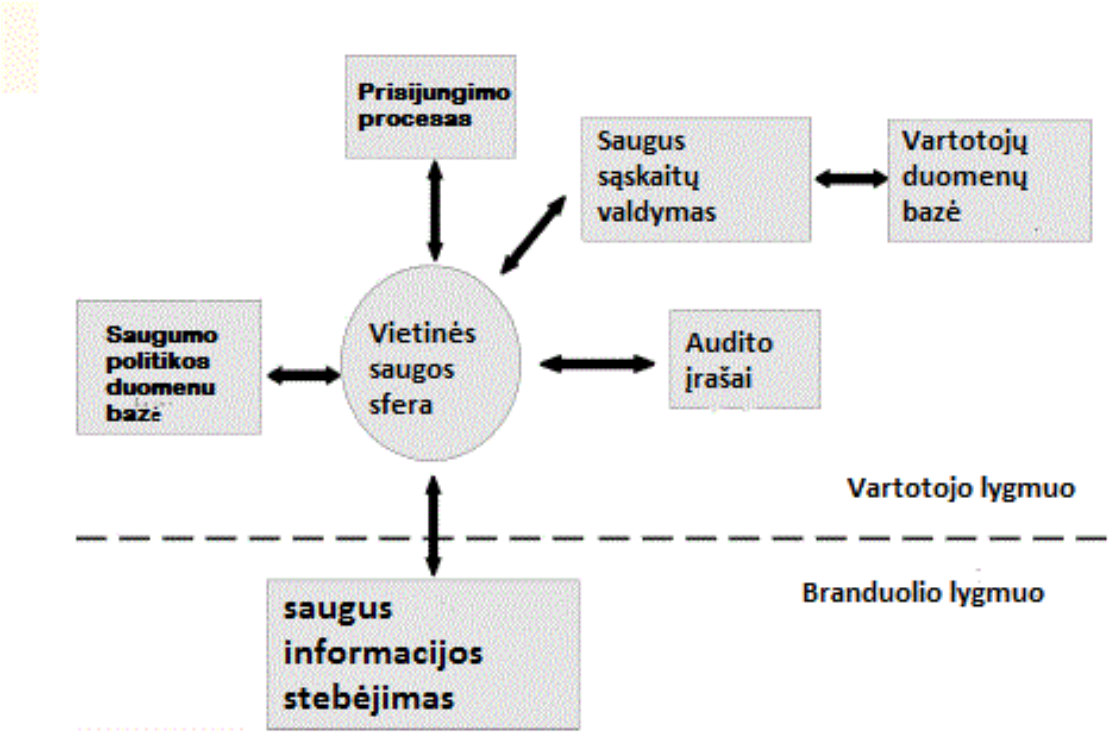
Požymis	Aprašymas
Užšifruota failų sistema	*Administratorius gali pasirinktinai užšifruoti failus ir katalogus NTFS 2000. Užšifravimas vartotojui yra skaidrus. Pagrindinė Windows programavimo sąsaja žino užšifravimo požymį duotai rinkmenai. Kad išsaugotumėte tai, nenaudojamas savo kaip savininko funkcijos, perkelti/kopijuoti rinkmenas, bet pasitikima sistemos CopyFile () ar MoveFile ().*
Disko kvotavimas	*Disko kvotavimo pagalba galima nustatyti maksimalią disko vietą kiekvienam vartotojui ir visuomet būs išmestas perspėjimas kai ji ar jis*

	viršys nustatyta ribą. Kvotavimas yra skaidrus vartotojams, kurie paprastai mato visą neužimtą disko vietą.
Reparse points (išskaidymo taškai)	*Išskaidymo taškai yra vartotojo apibrėžtų duomenų kolekcija skiriama rinkmenai ar aplankui. Šių duomenų formatas yra suprantamas aplikacijos, kuri saugo duomenis, ir failų sistemos filtrus, kuriuos jūs įdiegiate, kad interpretuotumėte ir apdirbtumėte tai rinkmenai ar aplankui.*
Reti failai	*Reti failai yra labai dideli failai, kuriuose nėra labai daug informacijos. Kai reti failai naudojami sklandžiai, sistema neišskiria vietos kietajame diske išskyrus vietas, kur yra kitokia informacija, kuri yra kažkokia kitokia nei nuliai.*
Prijungimo taškai	*Talpos prijungimo taškas yra egzistuojantis kelias, kur prijungiama kita talpa. Atsižvelgiant į tai, vartotojai ir paraiškos gali kreiptis į pajungtą talpą šiuo parinktu keliu. Tai leidžia jums suvienyti į vieną logišką rinkmenų išdėstymo sistemą skirtingas rinkmenos išdėstymo sistemas.*
Pakeitimų žurnalas (journaling)	*Pakeitimų žurnalas yra duomenų bazė, kurioje saugomi įrašai apie kiekvieną pakeitimą faile arba kataloge NTFS 2000 sistemoje. Ši sistema turi savo savą žurnalą su įrašais, atspindinčiais visų rinkmenų ir aplankų pakeitimus, kurie įvyko.*
Desktop.ini	*Viskas kas nepritaikoma aplankams yra saugoma desktop.ini faile. Tai yra mažas tekstinis failas, kur laikoma informacija apie aplankų piktogramas ir informacinius paaiškinimus.*

1.1.4.1. Vietinė saugos sfera

Vietinė saugos sfera yra saugos sistemos širdis. Ji atsakinga už patvirtintus vietinius ir nutolusius prisijungimus visokio tipo sąskaitoms.

- Tikrina sistemoje vartotojų prieigos leidimus
- Registravimosi metu kuria prieigos simbolius.
- Valdo vietinę saugumo politiką
- Aprūpina vartotojų patvirtinimą ir autentiškumo nustatymą
- Kontroliuoja tikrinimo politiką
- Įrašų audito žinučių generavimas



6 pav. Windows NT NTFS saugos modelis

1.1.5. Windows aplinkoje katalogų ir failų apsauga

Windows aplinkoje naudojamos failo prieigos teisės nustato kiekvienam vartotojui operacijų, kurias jis galės atlikti su tam tikru failu rinkinį.

Bendruoju atveju prieigos teisės gali būti aprašytos prieigos teisių matrica, kurioje stulpeliai atitinka sistemos failus, o eilutės atitinka pačius vartotojus. Stulpelio ir eilutės susikirtime nurodomos galimos operacijos.

2 lentelė. prieigos teisių matrica

Vartotojas/ failai	Failas1	Failas2
Vartotojas1	Skaityti	Rašyti
Vartotojas2	Vykdyti	Skaityti, rašyti

Windows operacinėje sistemoje yra du pagrindiniai teisių apibrėžimo būdai:

1. kiekvieno failo savininkas gali pats nustatyti kiekvienam vartotojui leidžiamas operacijas
2. pati sistema suteikia vartotojui tam tikras naudojimosi ištekliumi t.y. teisėmis pagal tai, kokiai grupei tas vartotojas priklauso.[19]

1.1.5.1. Failų sistemos užšifravimas

Failai ar katalogai šifruojami naudojant vieną raktą failo sistemos lygiu. Šifruojant failus duomenys šifruojami taikomųjų programų pagalba į kurias įtraukiami duomenų šifravimo moduliai.

Failų sistemos užšifravimas (Encrypted File System - EFS) [15] leidžia vartotojams užšifruoti ir iššifruoti failus, ir tokiu būdu apsaugo juos nuo įsilaužėlių, kurie gali neteisėtai prieiti prie slaptų duomenų (pavyzdžiui, pavogę nešiojamąjį kompiuterį ar išorinį diskų įrenginį ar įsilaužus i serverį).

Šifravimas yra skaidrus: Vartotojai dirba su užšifruotais failais ir aplankais, kaip su bet kokiais kitais failais ir aplankais. Jei vartotojas yra asmuo užšifravęs failą ar aplanką, sistema automatiškai iššifruoja failą ar aplanką, kai vartotojas prie jų prieina.

1.1.5.2. Pakartotinio objektų panaudojimo saugumas

Tai svarbus kreipinių valdymo priemonių papildymas, apsaugantis nuo atsitiktinio ar sąmoningo slaptos informacijos ištraukimo iš "šiukšlių". Kadangi informacija apie subjektą taip pat objektas, todėl svarbu užtikrinti pakartotinio subjekto panaudojimo saugumą.

1.1.6 Leidimai (permisions)

1.1.6.1. Prieiga

Bet koks vartotojas gali pasiekti failą ar katalogą trim budais:

- **read** (skaityti) – failo peržiūra tik skaitymui.
- **write** – redaguoti arba pakeisti failą arba ištrinti jį.
- **execute** – tam tikriems failams yra nustatyta ši prieiga, kad galėtų juos vykdyti programos. Pagal nutylėjimą, failai ir katalogai „Apache" žiniatinklio serveryje paprastai yra be įrašymo teisių todėl pirma turime pakeisti leidimus (permissions) jei šiuos failus turės redaguoti programos. Taip pat kai kurie failai esantys puslapio kataloge gali būti programos, tokiems failams taip pat reikėtų suteikti vykdymo (execute) teisę.

Skaitymo leidimas nustatomas failams, kurie tik pateikia informaciją ir negalima informacijos keisti. Rašymo leidimas reikalingas dinamiškai sukurti failus, modifikuoti failų turinį, arba ištrinti juos. Vykdymo leidimas reikalingas paleisti vykdomuosius failus, kaip pavyzdys kaip kad PHP skriptai. Dažniausi leidimai, kurie naudojami yra **(r-x),** t.y. skaitymo ir vykdymo.[14]

Katalogai ir failai, kurie bus viduje kito katalogo, vadinkime vaiko, o pagrindinį katalogą vadinkime tėvo. Taigi leidimus, kuriuos nustatome tėvo katalogui galios ir vaiko katalogui ir failams jo viduje, bet galima nustatyti taip, kad nepageidaujami leidimai negaliotų vaiko aplankams ir failams.

1.1.6.2. Vartotojai

Yra trys vartotojų grupės , kurie gali pasiekti failus, Naudojant šias grupes, galima lanksčiau suteikti leidimus. Jei vienas vartotojas turi leidimą pakeisti ar pašalinti failą tai tuo metu, kiti vartotojai tiktai sugebės skaityti/žiūrėti ir galbūt vykdyti failą.

- **owner** – failo savininkas.
- **group** – narys priklausantis grupei, kuriai priklauso failo savininkas.
- **other** - bet kas kitas turintis prieigą prie serverio.

Yra trys būdai vartotojams pasiekti failus ir katalogus ir yra trys galimi vartotoju tipai, todėl galima failus ir katalogus pasiekti devyniais būdais.

3 lentelė. Failų ir katalogų prieiga

Failo savininkas	Skaityti iš failo rašyti į failą vykdyti failą
Failo savininko grupės narys	Skaityti iš failo rašyti į failą vykdyti failą
Visi kiti likę	Skaityti iš failo rašyti į failą vykdyti failą

1.1.7. Išvados

Išanalizavus failų sistemas ir saugios failų sistemos modelius, galima spręsti, kad saugi failų sistema turi turėti tokias savybes:

1. Atsarginės kopijos, archyvai, visi saugomi duomenys, ir visa kita, viskas apsaugota užšifruojant.
2. Kalbant apie žiniatinklio failų sistemą ir prieigą per naršyklę prie sistemoje saugomų duomenų, saugioje failų sistemoje svarbu, kad informacija būtų saugiai pernešta tarp informacijos savininko ir informacijos naudotojo, turi būti naudojamas "nuo pradžios iki galo" (end-to-end) šifravimas. Tokiu būdu užtikrinamas duomenų konfindencialumas, bei vientisumas.
3. Svarbu decentralizuoti prieigos valdymą, mažos žmonių grupės arba pats informacijos savininkas turi nustatyti prieigos teises vartotojams, kurie nori prieiti prie informacijos.
4. Saugioje failų sistemoje naudojama tokia savybė kaip žurnaliavimas. Įjungus žurnaliavimą duomenys iš atminties prieš įrašant į failų sistemą įrašomi į žurnalą, o po to tik į pačią failų sistemą. Po sistemos „lūžimo" tikrinami žurnalo paskutiniai įrašai, patikrinama ar failų sistemoje yra įrašyti tie duomenys, jei ne tuomet labai greitai pagal žurnale esančius įrašus sistema atstatoma. Tokiu būdu užtikrinamas duomenų pasiekiamumas.
5. Saugioje failų sistemoje labai svarbu, kad nebūtų galima ištraukti svarbios informacijos „iš šiukšlių". Tam reikalingas saugus failų ištrynimas užrašant atsitiktine bitu seka. Svarbu, kad bandant atkurti duomenis iš serverio talpyklos su specialiom programom, nebūtų atkurti ištrinti duomenys.
6. Saugioje failų sistemoje svarbu fiksuoti visus sistemoje vykstančius veiksmus ir laiką, pavyzdžiui, koks vartotojas ir kokiu laiku atliko tam tikrus veiksmus. Visi įrašai saugomi seanso žurnaluose.
7. Saugioje failų sistemoje vartotojas turi būti autentifikuotas ir autorizuotas.

1.2. „APACHE" ŽINIATINKLIO SERVERIO FAILŲ SISTEMOS SAUGOS GALIMYBIŲ ANALIZĖ

Pagal failų sistemų analizės padarytas išvadas, kur išanalizavom kokias savybes turi turėti saugi failų sistema, bus analizuojamos „Apache" žiniatinklio galimybės, kokias savybes „Apache" turi ir kokių ne. Peržvelgiamos *.htaccess .htpasswd .htgroup* savybės, standartiniai „Apache" moduliai, bei kitos saugos savybės.

1.2.1. .htaccess apžvalga

Tai yra specifiniai žiniatinklio serverių failai. Nurodyti .htaccess failo parametrai taikomi katalogui, kuriame yra failas, taip pat ir visiems to katalogo pakatalogiams. [1]

Norint pakeisti „Apache" serverio katalogų saugumo nustatymus savo svetainėje, reikia sukurti svetainės *public_html* ar vidiniame kataloge tekstinį failą, kurio pavadinimas *.htaccess* (pavadinimas turi būtinai prasidėti tašku). Šiuo failu galima uždrausti ar leisti prieigą prie kataloge esančių puslapių, nukreipti lankytojus į kitą žiniatinklio adresą, pakeisti standartinius serverio klaidos pranešimus ir t.t.

Būtina peržvelgti svarbiausias komandas, kuriomis nustatomi saugumo parametrai. Šios komandos skirstomos į grupes, tokias kaip priėjimo valdymo, katalogų naršymo, suspaudimo įjungimo, failų paslėpimo ir kitos grupės. Dažnai norima visiškai išjungti priėjimą prie katalogų (pavyzdžiui, turime direktoriją su programiniais failais, kurie įterpiami į pagrindini failą. Tokiu atveju tik pagrindinis failas turi turėti priėjimą prie programinių failų, bet nė vienas kitas negali jų atidaryti).

1.2.1.1. Prieigos kontrolė (Access control) naudojant .htaccess

Allow, Deny

Dažnai norime prileisti žmones prie žiniatinklio serverio resursų, pagrįstus kažkuo kitu negu, kas jie yra. Pagal tai iš kur jie ateina. Prieigos suvaržymas , pagrįstas kažkuo kita negu vartotojo tapatumas, apskritai vadina *Prieigos Kontrolė (Access control).*

Tokiu atveju sukuriamas *.htaccess* failas kataloge, kurį norite apsaugoti, ir įkeliamos atitinkamos eilutės.[2]

Šiame faile turi būti tam tikros blokavimo taisyklės, *Kiekviena taisyklė aprašoma taip:*

Deny from IP_adresas

arba

Deny from puslapio_IP

arba

Deny from domenas

Uždraudžiame visus priėjimus *deny from all.* Jeigu norime leisti priėjimą tam tikram IP, įrašome tai: deny from all, allow from 127.0.0.1.Taip pat galime uždrausti failus, tiesiogiai nurodydami juos ir panašiai. Naudojant taisyklių aprašymo taisykles galima lanksčiai apriboti prieigą pagal minėtas savybes.

Satisfy

Papildomai naudojant autorizacija naudojama *Satisfy* direktyva.

- Jei reikia, kad atitiktų IR vartotojo IP IR slaptažodis, tam naudojamas *Satisfy all*.
- Jei reikia, kad atitiktų ARBA vartotojo IP ARBA slaptažodis, tam naudojamas *Satisfy any*.

1.2.1.2. Prieigos uždraudimas pagal failų plėtinius

Norėdami uždrausti priėjimą prie tam tikrų failų, galime naudoti reguliarius išsireiškimus (Regular expression) taip uždrausdami priėjimą prie failų, kurių galūnes nustatome *.htaccess* faile. Galime modifikuoti uždrausdami nustatytus failus (tarkim konfigūracijos failus, *robots.txt*, logų failus ir visa kita ką tik norime[1]

1.2.1.3. Svetainės katalogų bylų rodymo uždraudimas

Pasitaiko, kad svetainės direktorijose nebūna bylų, kurios pagal nutylėjimą būna pagrindinės svetainės bylos (pvz*.: index.html, index.php* ir pan.), tokiu atveju vartotojui atsidarius svetainę bus rodomos visos bylos esančios tame kataloge (pvz.: jeigu *www.Jusu_svetaine.lt* neturi pagrindinės bylos, bus rodomas visas katalogo turinys), o tai gali tapti rimta duomenų apsaugos problema. Kad

apsaugoti svetainės katalogo bylų rodymą, *.htaccess* byloje reikia įrašyti tokią eilutę „*Options –Indexes“, tuomet nebus rodomos visos kataloge esančios bylos*.[16]

1.2.1.4. Prieigos uždraudimas pagal valandas

Viena iš *.htaccess* failo savybių yra galimybė uždrausti prieigą pagal valandas. Galima uždrausti prieiga tam tikromis valandomis įrašius į .htaccess failą atitinkamas eilutes, tuomet lanksčiau suteikiama prieiga prie serveryje saugomų duomenų. Galima nustatyti kokiom valandom bus uždrausta prieiga.[2]

1.2.2. Naudotojo autentikacijos nustatymas su .htpasswd

Naudotojų sąskaitos specialia komanda (pagalbine programa) *.htpasswd*, sukuriamos administratoriaus pasirinktame faile, kuris privalo būti nepasiekiamas per žiniatinklį.

Sukuriant vartotoją paprašoma įvesti slaptažodį ir jį pakartoti. Vartotojo vardas ir slaptažodis saugomas nurodytame faile. Atidarę failą, kiekvienoje eilutėje matytume po naudotojo vardą po vardo dvitaškį ir toliau naudotojo slaptažodis užkoduotas MD5 algoritmu.[8]

4 lentelė. Vartotojo slaptažodžiai

naudotojas:slaptažodis
giedrius:Hj7skXX9chm3.sjU9yzJmM
petras:Ik5lJYxjr7sS33kIuVz9iG

1.2.3. Prieigos suteikimas grupei vartotojų

Norint, kad prieigą gautų ne vienas, o keli vartotojai, galima šiuos vartotojus priskirti vienai grupei. Norint tai padaryti naudojama *AuthGroupFile* direktyva. Sukuriamas grupės failas, kuriame surašomi grupės nariai. Surašymo būdas visai nesudėtingas, šį failą galima lengvai sukurti su teksto redagavimo programa.

Tai yra tiesiog sąrašas grupės narių, kurie surašyti iš eilės, atskiriant narius tarpais be kablelių. Įtraukus vartotojus į grupes ir sukūrus tiems vartotojams slaptažodžius. Visi nariai, kurie yra įtraukti į grupę ir turės slaptažodžio įrašą slaptažodžių faile, jiems bus suteikta prieiga, teisingai įvedus slaptažodį.

Taip pat yra ir kitas būdas leisti prieigą daugiau kaip vienam vartotojui. Vietoj to, kad kurti grupės failą, tiesiog galima naudoti direktyvą *„Require valid-user “*. Naudojant šį variantą,

bus suteikta prieiga visiems, kurie yra slaptažodžių faile, žinoma, įvedus teisingą slaptažodį kai to pareikalaujama. Tokiu būdu galima net pamėgdžioti grupės elgesį, priskirdami atskirą slaptažodžių failą kiekvienai grupei. Šio metodo pranašumas yra toks, kad „Apache“ turi tikrinti vieną failą, o ne du. Trūkumas yra toks, kad reikia saugoti didelį kiekį slaptažodžių failų ir kiekvieną kartą tai reikia teisingai nurodyti *AuthUserFile* direktyvoje.

1.2.4. Naudotojo autentifikacija naudojant .mod_auth ir .mod_ssl

Dažniausiai naudojamas žiniatinklio serveris „Apache“ turi modulį apsaugoti katalogo turinį nuo pašalinių akių. Tai naudotojo autentifikacijos modulis *mod_auth*. Kai vartotojas suveda puslapio adresą, kuris reikalauja autentifikacijos, naršyklė atidaro dialogo langą, prašantį įvesti vardą ir slaptažodį. Kitas būdas patekti į autentifikacijos reikalaujantį puslapį yra vartotojo vardą ir slaptažodį, atskyrus dvitaškiu bei su internetine eta gale, įrašyti į adreso eilutę tarp protokolo ir domeno vardo:[5][11]
protokolas://naudotojas:slaptazodis@www.domenas.com/katalogas/failas.pletinys

Lankytojui įvedus teisingus prisijungimo duomenis, jam parodomas adresuotas informacinis puslapis, kuris gali būti tiek paprastas dokumentas (html, txt, pdf), tiek scenarijaus (script) sugeneruotas turinys, tiek parsiųstinas vykdomasis failas. Reikia paminėti, kad vartotojo vardą ir slaptažodį, atskyrus dvitaškiu, naršyklė prieš siųsdama į serverį, užkoduoja base64 algoritmu, kuris serveryje atkoduojamas į grynąjį tekstą ir sulyginamas su užregistruotom prisijungimo sąskaitom. gan lengvai naudotojo vardas ir slaptažodis gali būti nuskaitytas ir atsikoduotas kažkur pakelėj link serverio. Dėl šios priežasties saugumo vardan kartu su naudotojo autentikacija naudinga naudot SSL.[13]

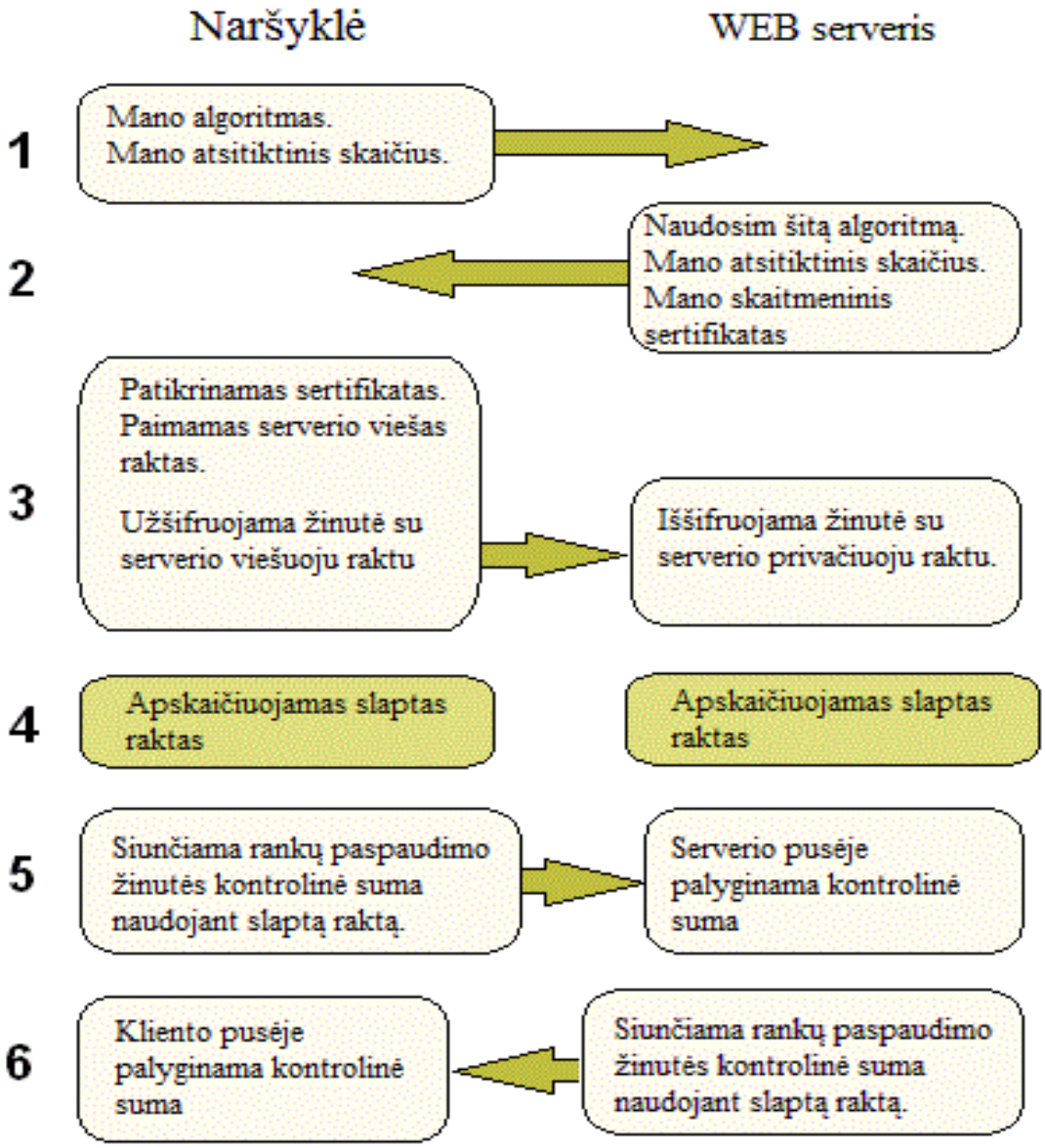
1.2.5. Automatinis peradresavimas į HTTPS protokolą

Jei svetainei užsakytas SSL sertifikatas ir sertifikato talpinimo paslaugą, galima tam tikrai svetainės direktorijai suderinti taip, kad ji lankytojams būtų rodoma tik per šifruotą HTTPS protokolą. Tai naudinga apsaugant svetainės dalis, kuriose lankytojas į serverį siunčia konfidencialią informaciją. Tam į svetainės *public_html* ar vidinę direktoriją įkeliamas *.htaccess* failas su nustatymais, kurie atliktų šia funkciją.[16]

1.2.6. Konfidencialumo ir vientisumo užtikrinimas perduodant duomenis tarp serverio ir kliento

SSL (Secure Sockets Layer) - Kriptografinis protokolas, skirtas informacijos, sklindančios internete apsaugojimui šifruojant [17]. Užtikrina, kad persiunčiami duomenys bus užkoduoti ir tarpinės grandys, kaip DNS (domenų vardų serverių) administratoriai, negalės jų tiesiogiai perskaityti.

Kai jungiamės prie internetinių banko sąskaitų ar kitų ypatingo saugumo reikalaujančių svetainių, dažniausiai suvedame “https://”, kur “s” nurodo saugų SSL jungimąsi. Komunikacijos saugumo sluoksnį įgyvendina „Apache“ modulis *mod_ssl*.



7 pav. „Kompiuterio darbalaukio enciklopedijoje“ 2005 metais pavaizduotas ssl veikimo modelis

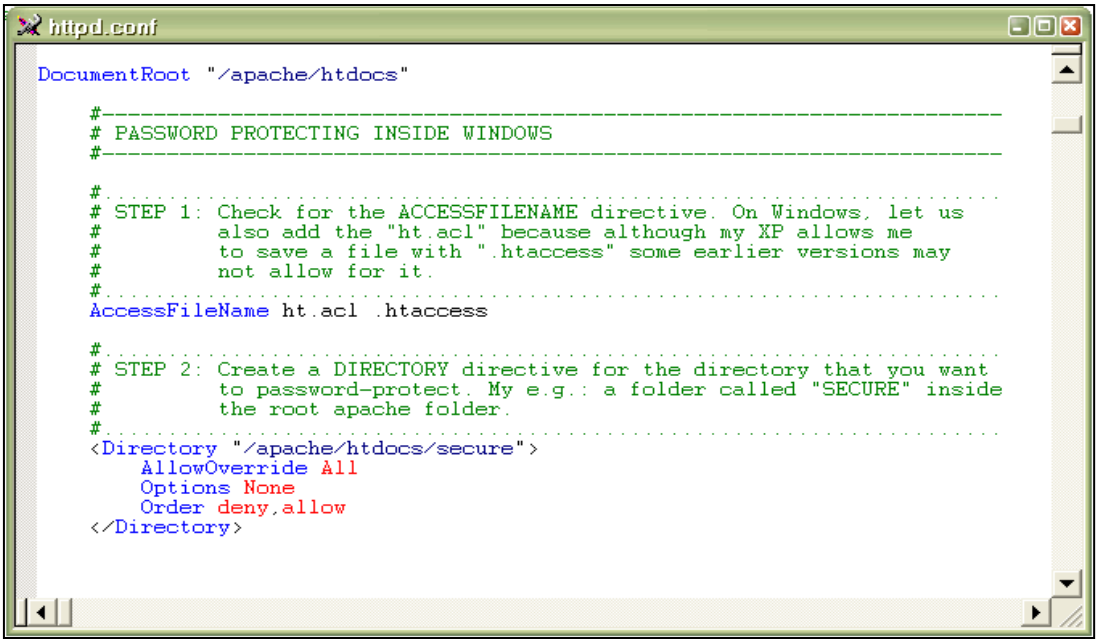
1.2.7. Grafinių sąsajų analizė

Analizuojama „Apache“ žiniatinklio serverio saugos parametrų nustatymų aplinka, taip pat paanalizuojami ir panašių žiniatinklio serverių aplinka.

1.2.7.1. Katalogų ir failų saugos nustatymai „Apache” žiniatinklio serveryje

„Apache“ turi daugybę privalumų. Bene vienintelis nusiskundimas dėl „Apache“ yra toks, kad kaip ir daugelis Unix operacinėse sistemose veikiančių programų, neturi grafinės sąsajos, skirtos valdymui. „Apache“ nustatymai, tarp jų ir katalogų saugos nustatymai, yra konfigūruojami per operacinės sistemos komandinę eilutę arba redaguojant konfigūracinius failus.

(8 pav.) vaizduojama aplinka, kurioje katalogams suteikiamos apsaugos savybės. Tai vykdoma komandinėmis eilutėmis, kas yra labai nepatogu ir reikalauja daug žinių iš tų, kurie nori pakeisti saugos lygį vienam ar kitam katalogui. „Apache“ vartotojams iškyla rimtų sunkumų, pasirinktam katalogui suteikiant tam tikrus saugumo nustatymus.



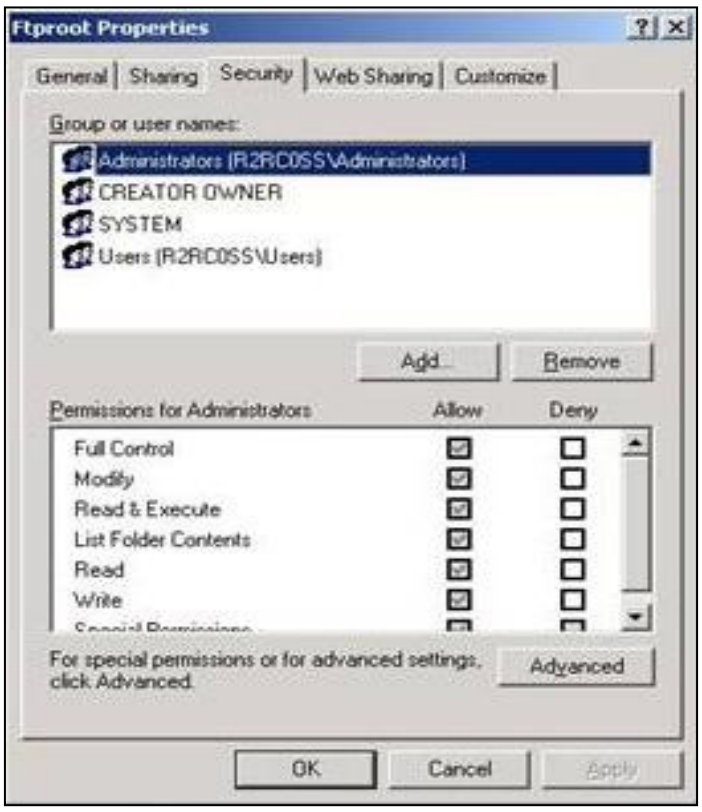
8 pav. Katalogų saugos redagavimas komandinėje eilutėje

Sudėtingas valdymas visad apsunkina patį procesą – šiuo atveju saugumo parametrų suteikimą „Apache“ žiniatinklio serverių katalogams. Kuo sudėtingiau atlikti veiksmus, tuo didesnė tikimybė įvelti daugiau klaidų, žinoma, ir pats procesas užtrunka

daug ilgiau. Net patyręs specialistas, kiekvieną kartą įvesdamas tas pačias komandas, neišvengs klaidų, nekalbant apie papildomai sugaištą laiką aiškinantis niuansus. Todėl labai svarbu turėti patogų įrankį, su kuriuo greitai, patogiai ir su minimalia klaidų rizika būtų galima atlikti veiksmus.

1.2.7.2. Windows aplinkoje realizuota saugos nustatymų grafinė sąsaja

„Windows Server 2003 Žiniatinklio Edition“ yra skirtas tinklo tarnyboms ir išteklių nuomai. Katalogų apsauga realizuojama priskiriant leidimus individualiems vartotojams ar vartotojų grupėms. Grafinė sąsaja išdėstyta aiškiai ir, lyginant su „Apache“ komandų įvedimu, čia bereikia uždėti varneles prie tokio saugumo parametro, kokį mes norime suteikti. Akivaizdu, kad grafinė sąsaja yra lanksti ir patogi.

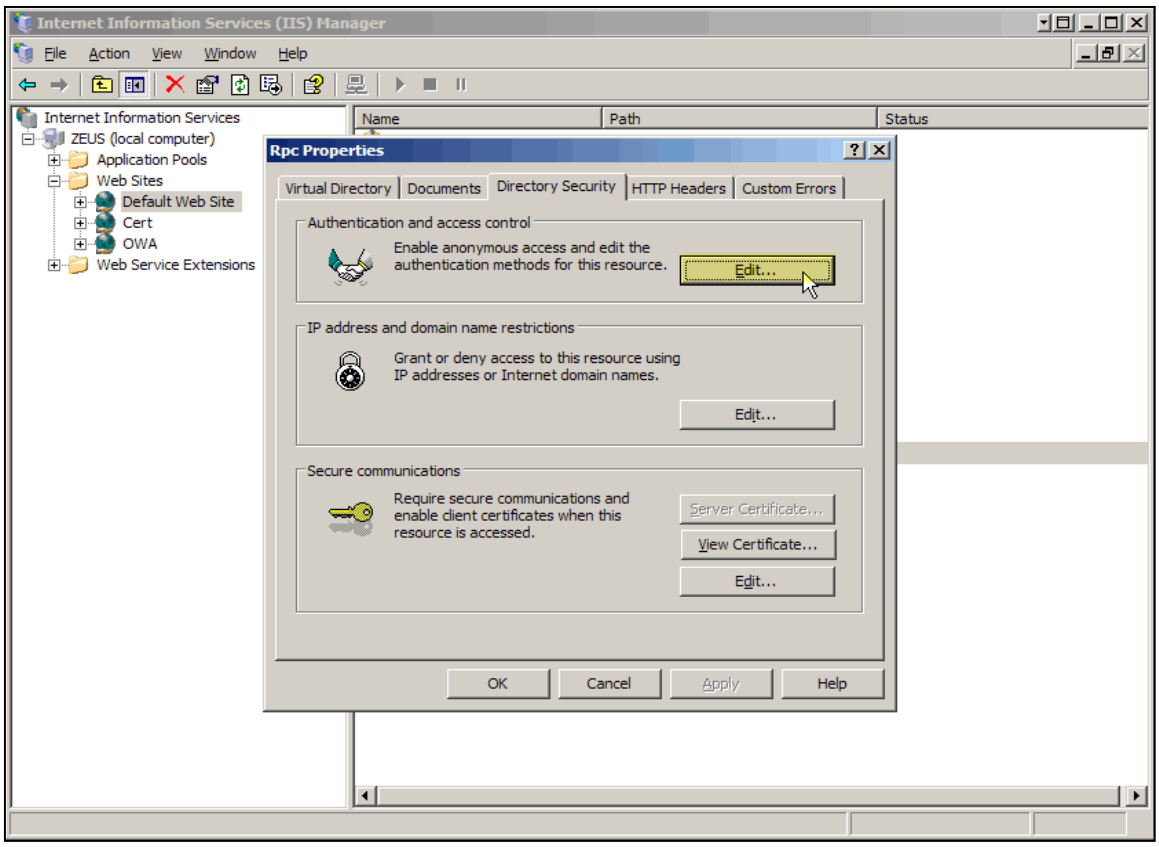


9 pav. „Windows Server 2003 Žiniatinklio Edition“**grafinė sąsaja**

1.2.7.3. Zeus žiniatinklio serveris

Zeus - aukštos kokybės interneto serveris. Dėl unikalios programinės įrangos architektūros Zeus serveris yra labai galingas, vienu metu gali palaikyti šimtus tūkstančius prisijungimų, gali saugoti šimtus tūkstančių žiniatinklių. Zeus žiniatinklio serveris kuriamas didelėms interneto svetainėms, kuriose vienu metu gali lankytis daug vartotojų.[20]

Pirmiausia Zeus programinė įranga turi puikiai išvystytą grafinę sąsają. Todėl Zeus, skirtingai nei daugumoje kitų serverių, nereikia atsiminti dešimčių komandų, norint jas rašyti į konfigūracinius failus, nes jame yra grafinė sąsaja kiekvienai produkto funkcijai. Grafiniai realaus laiko įrankiai iš karto parodo atsiradusias problemas. Ir nesvarbu, ar administruojama 1 arba 1000 internetinių svetainių, dėl puikiai išvystyto šio produkto funkcionalumo, nepajuntama didelių sunkumų.



10 pav. Zeus grafinė sąsaja ir katalogų apsaugos nustatymas

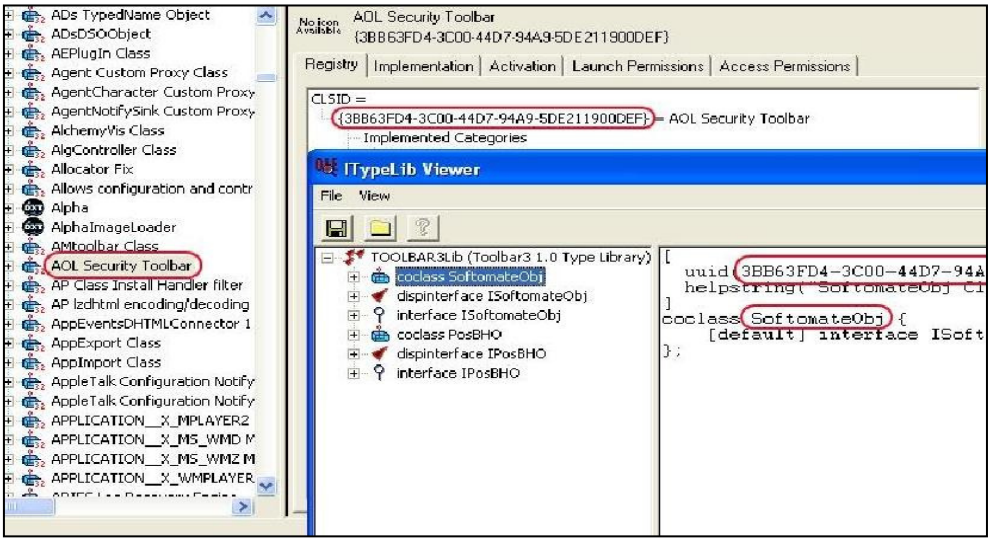
Zeus grafinė sąsaja pavaizduota (10 pav.) Lyginant su „Apache“ žiniatinklio serverio aplinka (8 pav.), iš karto matomi dideli skirtumai. Zeus aplinka yra aiški, lengvai

suprantama. Norint suteikti saugos elementus, užtenka paspausti mygtuką ir nurodyti, kokius saugumo parametrus norime suteikti pasirinktam katalogui. Čia nėra komandinių eilučių rašymo ir seka komentarai, paaiškinantys, ką reiškia kiekvienas elementas. Paprastumas ir aiškumas leidžia atlikti norimą veiksmą greitai ir be klaidų, o visas sudėtingas komandas atlieka pati programinė įranga, palengvindama darbą vartotojui.

1.2.7.4. AOL žiniatinklio serveris

AOL - *Amerika Online's* Atviro šaltinio žiniatinklio serveris. AOL yra atviro kodo programinė įranga, lengvai konfigūruojama dėl savo lankstumo. Šioje programinėje įrangoje yra integruotas API (*application program interface*), kuris skirtas rašyti programų modulius. AOL architektūra visiškai kitokia nei „Apache“, dėl to kai kurie darbai atliekami efektyviau. AOL yra daugiagijis serveris, naudojamas didelio masto dinamiškoms žiniatinklio svetainėms.[3]

AOL žiniatinklio serveryje nustatymai, tarp jų ir mus dominantys katalogų saugos parametrų nustatymai, taip pat turi aiškią grafinę sąsają. Ši grafinė sąsaja pateikta (11 pav.) Kortelėje “Access Permision” yra nustatinėjami katalogų saugos leidimai, kurie taip pat nereikalauja komandinėje eilutėje rašomų komandų žinojimo.



11 pav. AOL žiniatinklio serverio grafinė sąsaja

1.2.8. Išvados

Išanalizavus „Apache“ žiniatinklio failų sistemos saugos galimybes, galima išskirti tokias saugos savybes ir trūkumus:

1. Duomenų, saugojamų „Apache“ žiniatinklio serverio talpyklose, šifravimo nėra.

2. „Apache” žiniatinklio serveryje failų sistemos duomenų konfindencialumas užtikrinamas naudojant kriptografinį protokolą apsaugant duomenis, kai jie keliauja internetu tarp informacijos vartotojo ir duomenų savininko žiniatinklio serveryje laikomų duomenų. Duomenys pasiekiami per naršyklę jungiantis prie serverio, naudojant kriptografinį protokolą šifravimas gaunasi skaidrus.

3. „Apache” žiniatinklio serveris neturi saugaus failo ištrynimo savybės, todėl išlieka galimybė ištraukti informaciją “iš šiukšlių”.

4. Nėra žurnaliavimo savybės, tai svarbu norint atstatyti duomenis ir užtikrinti duomenų pasiekiamumą jei sistema “lūžtų” ar duomenims pakenktų piktavalis, taip pat ir žmogiškojo faktoriaus atveju, kai vartotojas suklysta.

5. Norint suteikti saugios failų sistemos savybes „Apache” žiniatinklio serveryje saugomiems duomenims, reikia suteikti galimybę kiekvienam informacijos savininkui pačiam įtraukti vartotojus ir jiems suteikti tam tikro lygio prieigą.

6. Prieiga valdoma *.htaccess* failo pagalba, bet nekiekvienas gali pasinaudoti šia galimybe, nes reikia nemažai žinių kaip šį failą konfigūruoti. Saugos savybių suteikimas paprasto vartotojo duomenims, kai tai atlieka pats duomenų savininkas, tampa per daug sudėtingas ir dažniausiai duomenys lieka neapsaugoti.

7. Viena iš pagrindinių problemų norint suteikti saugios failų sistemos savybes saugomiems duomenims, tai šių savybių suteikimas, nes jas reikia suteikti rašant komandines eilutes ir konfigūruojant konfigūracinius failus. Šiai problemai išspręsti reikalinga grafinė sąsaja skirta suteikti. saugos parametrus.

2. SAUGIOS „APACHE“ ŽINIATINKLIO SERVERIO FAILŲ SISTEMOS MODELIS

Pagal analizės išvadose nustatytus „Apache“ žiniatinklio serverio failų sistemos trūkumus, bei pagal tai kaip turi atrodyti saugi failų sistema, projektuojamas saugios „Apache” žiniatinklio serverio failų sistemos modelis. Iškeliami reikalavimai kaip turi atrodyti serverio saugi failų sistema, aprašoma kokios naujos savybės prisidės, kokie standartiniai konfigūraciniai failai bus naudojami ir kt.

2.1. Tikslai

Pagrindinis tikslas yra sukurti saugią „Apache” žiniatinklio serverių failų sistemą, kurios pagalba būtų suteiktas saugumas serverio paslaugomis besinaudojančio vartotojo duomenims.

„Apache” žiniatinklio serverio vartotojas, norėdamas suteikti saugios failų sistemos savybes savo serveryje laikomiems duomenims, galės prisijungti prie saugos parametrų suteikimo grafinės sąsajos lango ir nustatyti norimus parametrus.

2.2. Reikalavimų specifikavimas

2.2.1. Funkciniai reikalavimai

- Informacija užšifruojama nuo informacijos savininko duomenų iki informacijos vartotojo
- Informacija turi būti užšifruota duomenų saugojimo metu
- Šifravimas yra skaidrus
- Saugus prisijungimas naudojant SSL
- Palaikomas žurnaliavimas
- Palaikomas auditas
- Saugus duomenų ištrynimas, užrašant atsitiktine bitų seka
- Vartotojas gali keisti savo patalpintų katalogų ir failų saugos nustatymus, pagal duomenų svarbą nenaudojant komandinių eilučių ir nekonfigūruojant tekstinių failų
- Vartotojo registracija
- Vartotojo autentifikacija
- Galimybė įtraukti naują vartotoją
- Galimybė sudaryti vartotojų grupes
- Vartotojai įtraukiami į grupes su nustatytom prieigos teisėm

- Sistemoje naudojami stiprus slaptažodžiai.
- Įvestas laiko ribojimas, kad užmiršus atsijungti, sistema vartotoją atjungtų automatiškai.
- Pagalbos ir paaiškinimų apie galimas saugos savybes suteikimas

2.2.2. Nefunkciniai reikalavimai

- *Kokybė* – turi būti aiškiai išdėstyti galimi saugos nustatymų pasirinkimai, vartotojas turi lengvai orientuotis sukurtoje grafinėje sąsajoje.
- *Patikimumas* – turi būti patikimas duomenų įrašymas.
- *Saugumas* – turi būti apsaugota grafinė sąsaja, kad negalėtų kas nors įsilaužti ir nurodyti klaidingus duomenis.
- *Minimalus vėlinimas* – saugumo modulio programiniame kode turi būti naudojami algoritmai ir scenarijai (script), kurie kuo mažiau apkrauna sistemą ir gaunamas mažesnis vėlinimas.
- *Grafinė sąsaja* – lankstesnė aplinka, kurios pagalba aiškiau suteikiamos saugumo savybės.

2.2.3. Reikalavimai vartotojo sąsajai

1. Vartotojo sąsaja turi būti suprantama kiekvienam vartotojui;
2. Vartotojo sąsaja turi būti neperkrauta aprašymais;
3. Lengvai pasiekiama pagalba vartotojui;

2.2.4. Sistemos vartotojai

1. Duomenų laikomų serveryje savininkas
2. Vartotojai turintys prieigą prie duomenų.
3. Vartotojų grupės su priskirtom tam tikrom teisėm, turinčios prieigą prie duomenų.

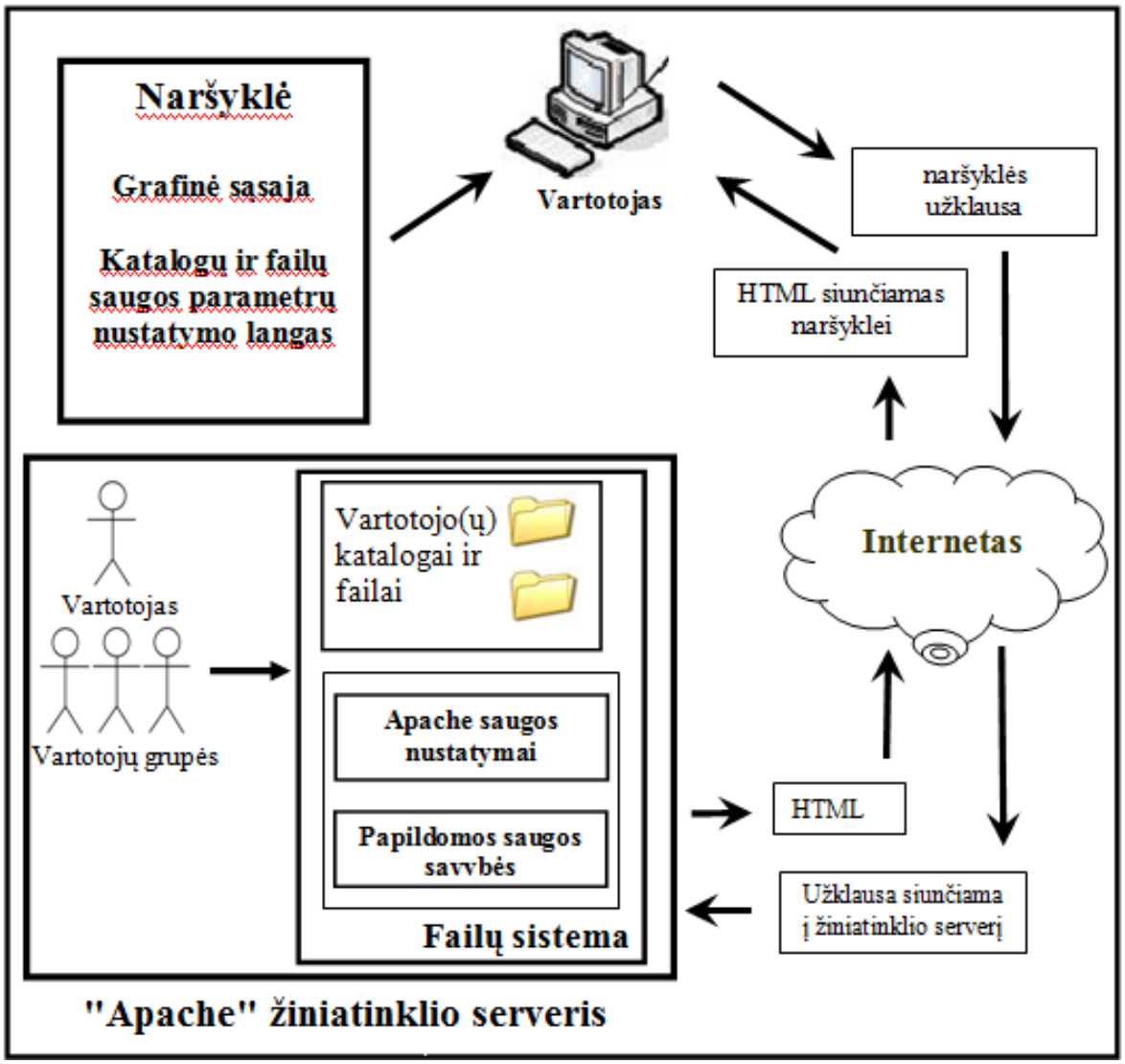
2.3. Sistemos architektūra

Atvaizduojamas projektuojamos saugios „Apache" žiniatinklio serverio failų sistemos modelis ir jį sudarantys elementai.

2.3.1. Koncepcija

Koncepcija (12 pav.) atvaizduoja kokie pagrindiniai elementai sudaro bendrą sistemą ir kaip šie elementai susiję tarpusavyje. Pagrindiniai elementai sistemoje yra: naršyklė, internetas ir „Apache" žiniatinklio serveris, bei prie serverio saugomų resursų besijungiantys vartotojai.

Vartotojų duomenims suteikiamos saugios failų sistemos savybės, užtikrinančios duomenų saugumą.



12 pav. projektuojamos saugios failų sistemos eksperimentinio modelio koncepcija

Vartotojas jungiasi prie grafinės sąsajos ir keičia saugos parametrus pasirinktiems katalogams, failams „Apache" žiniatinklio serveryje.

„Apache žiniatinklio serveris priima naršyklės užklausas ir generuoja HTML kodą, kuris internetu siunčiamas vartotojui.

2.3.2. „Apache" žiniatinklio serverio saugios failų sistemos architektūra

Eksperimentinę failų sistemą sudaro tokie pagrindiniai elementai:

1. Naudojami moduliai
    1.1 Autentifikacijai
    1.2 Duomenų šifravimui (duomenis persiunčiant)
    1.3 Scenarijų (Script) valdymui
    1.4 Žurnaliavimo modulis
    1.5 Šifravimo modulis (duomenis saugant)
    1.6 Failų ištrynimo modulis
    1.7 Vartotojų valdymo modulis

    1.1-1.3 standartiniai „Apache" žiniatinklio moduliai naudojami užtikrinti saugios failų sistemos savybes. Už autentifikavimą atsakingas *mod_auth* modulis, duomenų konfindencialumui užtikrinti ir saugiam duomenų užšifravimui nuo informacijos vartotojo iki duomenų savininko laikomų rinkmenų bei aplankų atsakingas modulis *mod_ssl*. Taip pat saugios failų sistemos savybes užtikrinti naudojami ir skriptų valdymo moduliai.

    1.4-1.7 papildomi moduliai užtikrina papildomas naujas projektuojamas savybes, tai žurnaliavimą, duomenų šifravimą kai duomenys saugomi serveryje, saugų failų ištrynimą bei vartotojų grupių sudarymą, kai įtraukiami nauji vartotojai, vartotojai skirstomi į grupes su skirtingom teisėm.

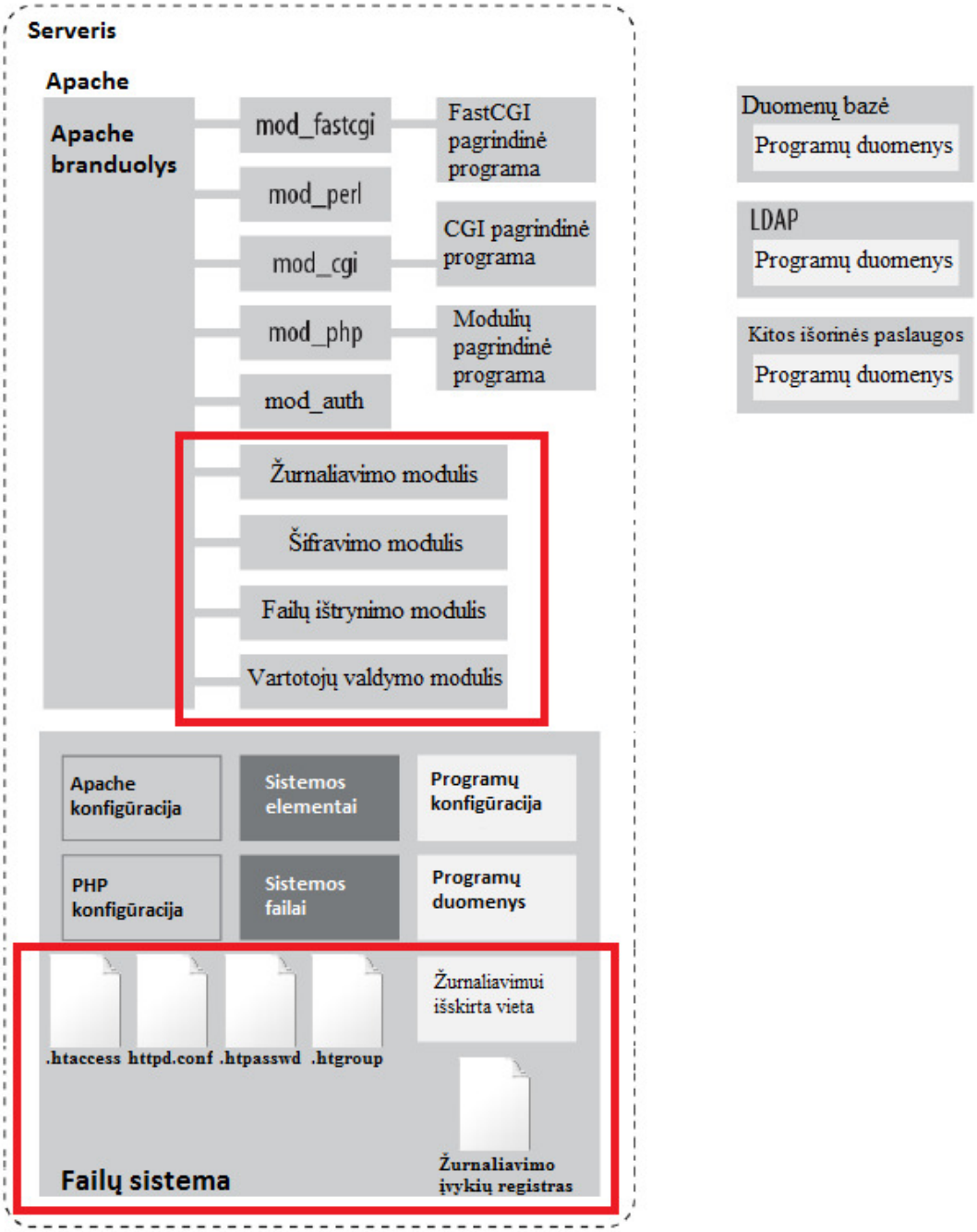
2. Naudojami konfigūraciniai failai
    2.1 Vartotojų registracijai
    2.2 Vartotojų autentifikacijai
    2.3 Grupių sudarymui
    2.4 Rinkmenų ir aplankų saugos savybių suteikimui
    2.5 Naudojamų modulių įjungimui

Šioje failų sistemoje naudojami tokie standartiniai „Apache" žiniatinklio serverio konfigūraciniai failai, kaip *httpd.conf, .htaccess, .htpasswd, .htgruop*. [18]

Vartotojų registracijai, grupių sudarymui, naujo vartotojo įtraukimui naudojami *.htpasswd* bei *.htgroup* failai.

Svarbių modulių įjungimui naudojamas httpd.conf konfigūracinis failas.[9]

Prieigai apriboti ir suteikti svarbias saugos savybes užtikrinant saugesnę failų sistemą, naudojamas *.htaccess* failas.



13 pav. Eksperimentinio modelio struktūra

3. Failų sistemoje išskirta vieta
    3.1 Vartotojų duomenims
    3.2 Žurnaliavimui išskirta duomenų saugojimo vieta

Failų sistemoje yra naudojama vieta vartotojo duomenims saugoti, bei išskirta vieta naujai failų sistemos savybei – žurnaliavimui. Tai žurnalas, kuriame saugomi metaduomenys ir failų bei katalogų architektūra. Redaguojant failus ar sukuriant naujus ar ištrinant visi pakeitimai pirma įrašomi į šį žurnalą ir tik paskui į pačią sistemą.

Meta-duomenys – tai duomenys apie duomenis, specifinė įvairių formatų dokumentuose saugoma informacija. Meta-duomenyse gali būti saugomas dokumento autoriaus vardas, jį sukūrusios organizacijos pavadinimas, programinės arba aparatinės įrangos paliktos žymės, dokumentų modifikavimo istorija, ir t.t. Meta- duomenyse gali būti saugomi net ištisi teksto, anksčiau buvusio dokumento dalimi, bet vėliau ištrinti, gabalai.[12]

Failų sistemoje pridedamos keturios papildomos savybės:

1. Saugus failo ištrynimas (apsauga nuo ištraukimo iš “šiukšlių”)
2. Galimybė įjungti žurnaliavimo savybę taikomajame lygmenyje.
3. Duomenų šifravimas kai duomenys saugomi serveryje.
4. Informacijos savininkas įtraukia naujus vartotojus ir iš esamo vartotojų sąrašo suskirsto vartotojus į grupes su skirtingom teisėm.

2.4. UML Diagramos

Schematiškai atvaizduojama kaip vartotojas savo duomenims gali suteikti saugios „Apache” žiniatinklio failų sistemos savybes. Kokius veiksmus gali jis atlikti ir kokie procesai dalyvauja.

2.4.1. Use Case diagrama

14 pav. prie saugos modulio prisijungusių vartotojų operacijos

Schemoje (14 pav.) parodoma kokias operacijas gali atlikti žiniatinklio serverio paslaugomis besinaudojantys vartotojai. Vartotojo duomenys patalpinti serveryje. Vartotojas prisijungia prie saugos parametrų nustatymo lango ir gali suteikti saugios failų sistemos savybes savo patalpintiems duomenims. Galimi veiksmai yra šie:

1. Saugos parametrų nustatymas naudojant *.htaccess*
2. Katalogo apsauga slaptažodžiu naudojant *.htaccess* ir *.htpasswd*
3. Prieigos teisių nustatymas vartotojams, suskirstant juos į grupes.
4. Naujo vartotojo įtraukimas
5. Grupių sudarymas
6. Saugus failų ištrynimas
7. Žurnaliavimo įjungimas
8. Duomenų šifravimas
9. Audito įjungimas
10. Pagalbos lango iškvietimas su paaiškinimais
11. Atsijungimas

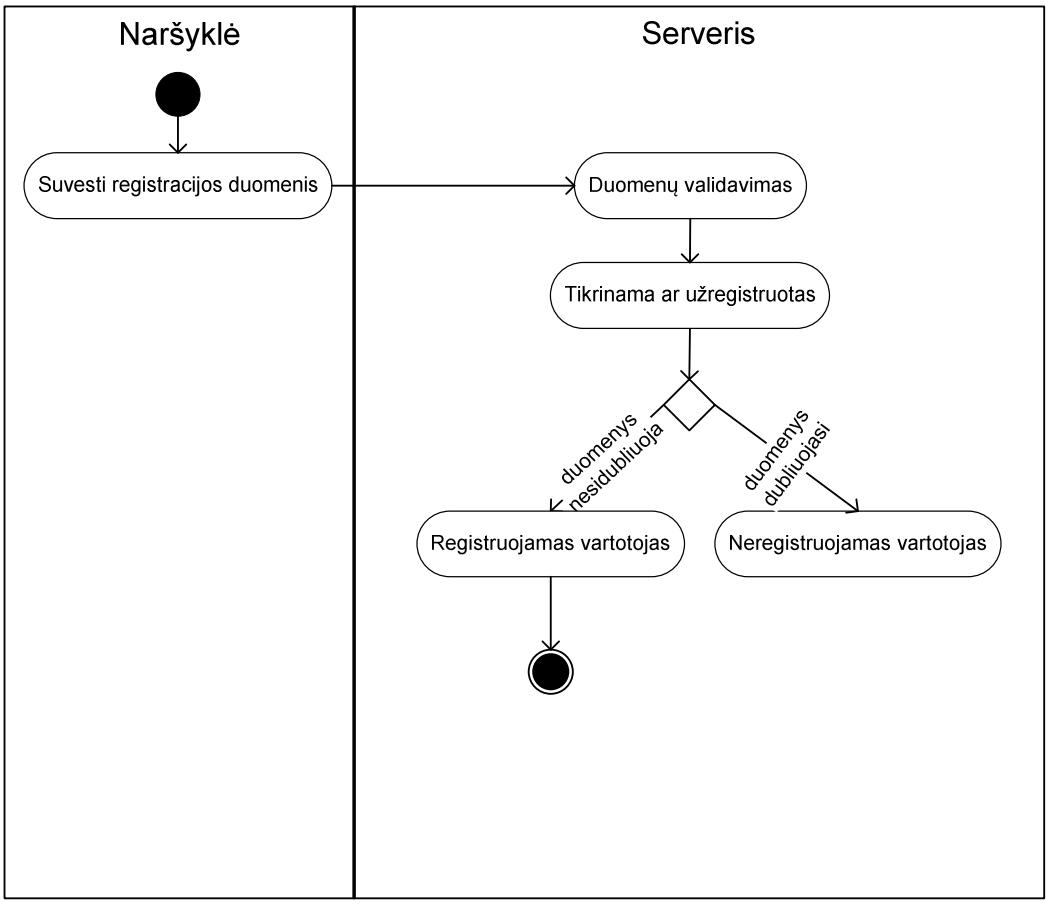
Visi veiksmai atliekami vartotojui prisijungus iš savo kompiuterio naudojant naršyklę ir per šifruotą HTTPS protokolą, kurį užtikrina „Apache" modulis *mod_ssl*. Todėl visi atliekami veiksmai yra saugūs ir nematomi tarpiniuose tinklo įrenginiuose.

2.4.2. Registravimo procesas

Žemiau pavaizduotoje schemoje (15 pav.) atvaizduojamas registracijos procesas. Registracijos procesą sudaro tokie etapai:

1. Suvedami registracijai reikalingi duomenys, tai vartotojo vardas, slaptažodis, bei elektroninio pašto adresas.

   Vartotojo slaptažodis užkoduojamas MD5 algoritmu ir kartu su vartotojo vardu bei elektroniniu pašto adresu saugojamas serveryje slaptažodžių faile
2. Duomenys validuojami, tikrinama ar nesielgiama piktaviliškai. Taip pat apribotas informacijos įvedimo laukelių įvedamų simbolių kiekis.



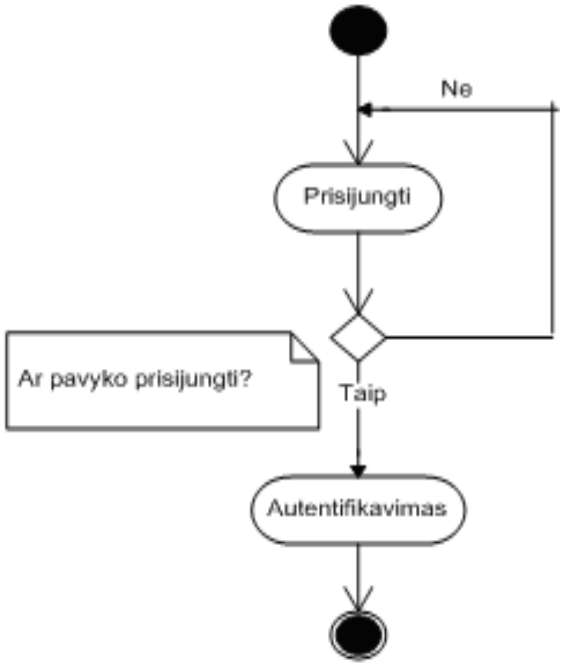
15 pav. vartotojo registracija

3. Tikrinama ar nesikartoja registracijos duomenys, tikrinimas reikalingas, kad sistemoje nesidubliuotų registruoti vartotojai. (tikrinama pagal vartotojo vardą ir elektroninio pašto adresą.
4. Jei tikrinant nustatoma, kad:
    a.) duomenys dubliuojasi, registracijos procesas nutraukiamas.
    b.) duomenys nesidubliuoja, tuomet priregistruojamas naujas vartotojas.
5. Registracijos proceso pabaiga.

2.4.3. Prisijungimo operacija

Žemiau pavaizduotoje schemoje (16 pav.) atvaizduojama prisijungimo operacija. Vartotojas suveda duomenis vartotojo vardą ir slaptažodį.

Suvestas slaptažodis užkoduojamas MD5 algoritmu ir siunčiamas kartu su vartotojo vardu į serverį, kur prisijungimui suvestas naujas užkoduotas slaptažodis palyginamas su serveryje išsaugotu taip pat užkoduotu slaptažodžiu. Jei sutampa – tuomet autentifikacija sėkminga.



16 pav. Prisijungimo operacija

2.4.4. Prieigos teisių suteikimas

Schemoje (17 pav.) pavaizduotas prieigos teisių vartotojams suteikimo procesas. Yra fiksuotos aštuonios grupės į kurias galima įtraukti vartotojus pagal pasitikėjimo lygį ir pagal tai kokias teises galima ar reikia suteikti vartotojui. **Šios grupės turi tokius leidimus:**

1. GRUPĖ RWX
2. GRUPĖ R – –
3. GRUPĖ RW –
4. GRUPĖ R – X
5. GRUPĖ – W –
6. GRUPĖ – – X
7. GRUPĖ – W X
8. GRUPĖ – – –

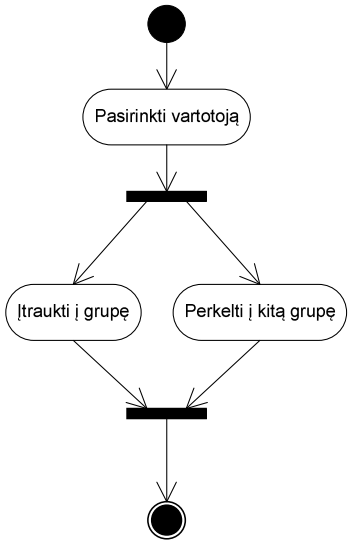
Duomenų savininkas pasirenka vartotojus iš galimų vartotojų sąrašo ir gali atlikti tokius veiksmus:

1. Įtraukti vartotoją į vieną iš aštuonių grupių

Duomenų savininkui nereikia nustatinėti kiekvienam vartotojui atskirai teisių, tereikia priskirti vienai iš aštuonių skirtingo saugumo grupių.

2. Perkelti iš vienos grupės į kitą.

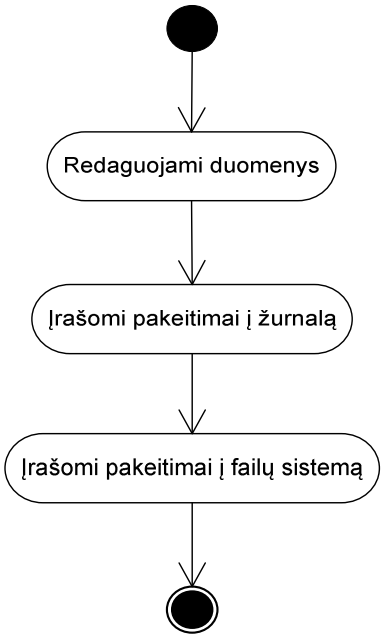
Įtraukus vartotoją į vieną iš grupių bet kada galima jį perkelti į aukštesnio lygio prieigos teises turinčią grupę arba permesti į žemesnio lygio grupę kur teisių bus mažiau.



17 pav. prieigos teisių suteikimas

2.4.5. Žurnaliavimo procesas

Schemoje (18 pav.) matome žurnaliavimo procesą. Šio proceso metu atvaizduojama kaip vyksta žurnaliavimas failų sistemoje. Redaguojant duomenis visi pakeitimai pirmiausiai įrašomi į žurnalą ir tik paskui įrašoma į failų sistemą.

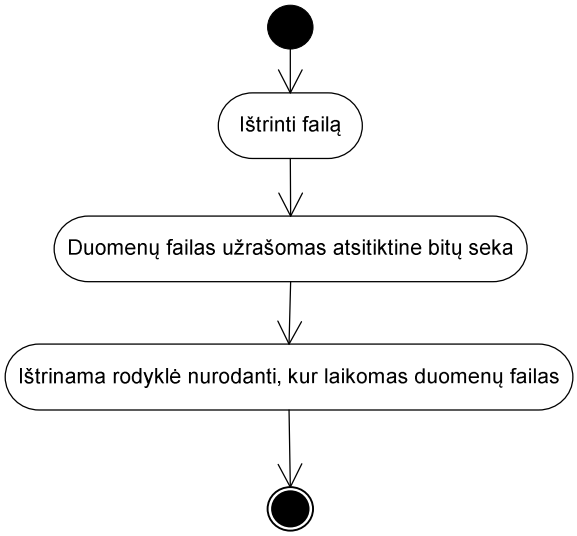


18 pav. žurnaliavimo eiga

2.4.6. Saugaus duomenų ištrynimo procesas

Schemoje (19 pav.) parodomas pagrindinis principas kaip saugiai galima ištrinti duomenis projektuojamoje saugioje „Apache" žiniatinklio serverio failų sistemoje.

Pirmiausiai duomenų failas užrašomas atsitiktine bitų seka ir tik paskui ištrinama rodyklė į failų sistemos vietą kur laikomas failas. Tokiu atveju, kad ir paleidus specialias programas tikrinti serverio diskų paviršių nebus rasti konfindencialūs duomenys.



19 pav. Saugus duomenų ištrynimas

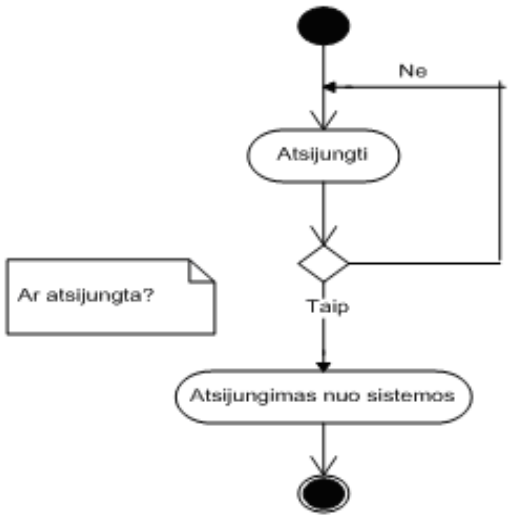
2.4.7. Pagalbos iškvietimo procesas

Labai svarbi failų sistemos savybė tai informacijos apie galimas saugos galimybes suteikimas. Todėl projektuojamoje saugioje failų sistemoje taip pat yra numatytas pagalbos suteikimas ir aiškus informacijos išdėstymas. Schemoje (20 pav.) atvaizduojamas pagalbos suteikimo procesas. Pagalba suteikiama iškviečiant langą ir vartotojas gali pasiskaityti ir greitai išsiaiškinti iškilusius klausimus.

20 pav. Pagalbos lango iškvietimas

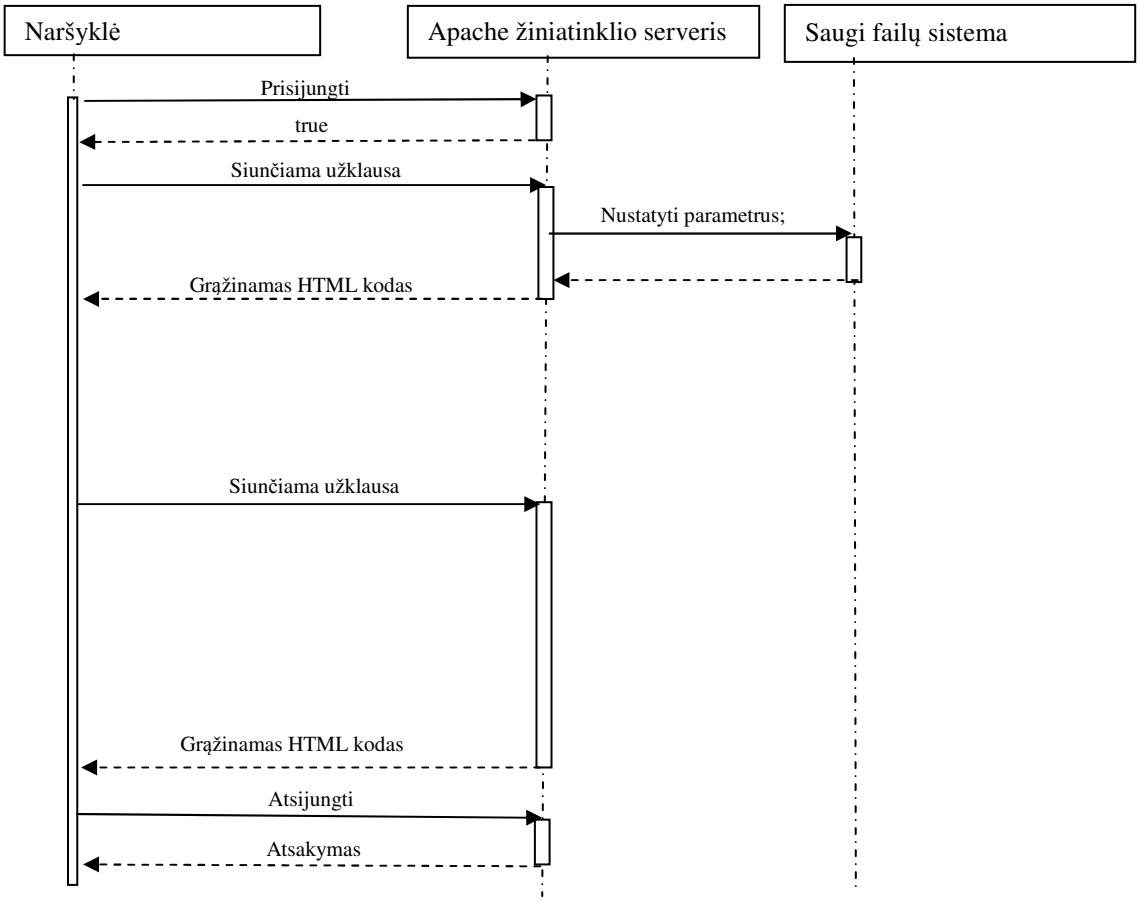
2.4.8. Atsijungimo nuo sistemos procesas

Schemoje (21 pav.) pavaizduotas atsijungimo nuo grafinės sąsajos procesas. Vartotojas norėdamas atsijungti nuo sistemos pasirenka mygtuką atsijungti ir sistema išmeta lentelę ir paklausia ar tikrai norite atsijungti. Patvirtinus – atsijungiama nuo sistemos



21 pav. Atsijungimas nuo sistemos

2.5. Sekų diagrama

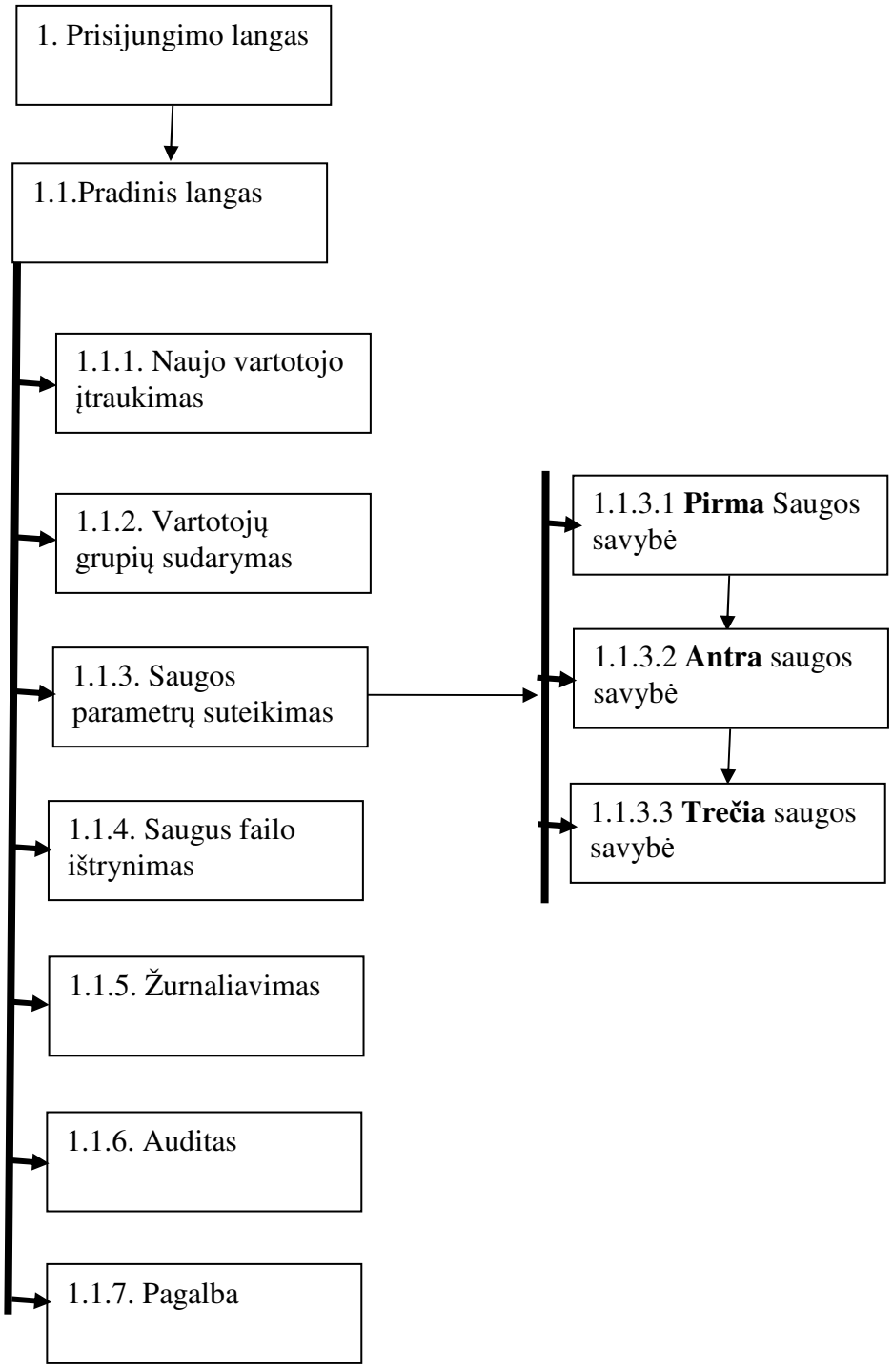


22 pav. sekų diagrama

Sekų diagramoje parodyta kaip tarpusavyje susieti sistemos, kurioje projektuojamas saugi failų sistema, objektų tarpusavio ryšiai. Matome, kad vartotojas prisijungia ir tada veiksmai atliekami siunčiant užklausas į serverį ir gaunamas atsakymas HTML pavidalu. Taip pat nustatomi saugos parametrai.

2.6. Langų planas

Žemiau pateiktoje schemoje (23 pav.) pateiktas grafinės sąsajos langų tarpusavio ryšių planas. Čia pavaizduota kaip vartotojas gali suteikti tam tikras saugios failų sistemos savybes.



23 pav. Vartotojo sąsajos langų planas

2.7. Išvados

„Apache” žiniatinklio failų sistemai nustatyti funkciniai ir nefunkciniai reikalavimai, šie reikalavimai parodo kaip turi atrodyti saugi „Apache” žiniatinklio failų sistema.

Suprojektuotoje sistemoje atsiranda keturi nauji moduliai, tai žurnaliavimo, duomenų saugojamų serverio talpyklose šifravimas, saugaus failų ištrynimo ir vartotojų valdymo moduliai. Kiekvienas modulis atsakingas už vieną papildomą saugios failų sistemos savybę.

Saugioje „Apache“ žiniatinklio failų sistemoje informacijos savininkas gali vartotojus išskirstyti į aštuonias grupes, kurios kiekviena turi konkrečias teises, kurios nekinta. Norint pakeisti vartotojo teises į kitokias, vartotojas keliamas į kitą grupę turinčią kitokias teises. Vartotojų prieigos valdymas duomenų savininkui tampa lankstesnis ir paprastesnis.

Informacijos savininkas pats be administratoriaus pagalbos suteikia saugios failų sistemos savybes savo duomenims ir pats nustato prieigą pasirinktiems vartotojams.

3. SAUGIOS „APACHE” ŽINIATINKLIO SERVERIO FAILŲ SISTEMOS MODELIO REALIZAVIMAS IR TYRIMAS

Realizuojama saugios „Apache” žiniatinklio serverio failų sistemos savybių suteikimo aplinka. Šio parametrų nustatymo lango galimybėmis duomenų savininkas gali nustatyti saugios failų sistemos teikiamas savybes savo serveryje laikomiems duomenims.

3.1. Sistemos prototipo realizavimo įrankio pasirinkimas

Kadangi realizuojama sistema bus pasiekiama naršyklės pagalba, todėl įrankius pasirenkame skirtus žiniatinklio puslapiams kurti. HTML ir CSS kalbos bus naudojamos grafinės aplinkos formavimui, o PHP kalba bus naudojama saugios failų sistemos funkcijom atlikti.

3.2. Vartotojo aplinka

Pirmiausiai vartotojas prisiregistruoja. Tam reikalinga suvesti registracijos duomenis, tai vartotojo vardas, slaptažodis ir elektroninio pašto adresas. (24 pav.)

Slaptažodį prašoma pakartoti, tai būtina, nes jei nebūtų prašoma pakartoti vartotojas gali suklysti įvedinėdamas norima slaptažodį ir registracija bus sėkminga bet norint prisijungti slaptažodis netiks.

24 pav. Vartotojo registracija

Taip pat reikia nurodyti ir elektroninio pašto adresą, šiuo adresu bus nusiunčiami registracijos duomenys. Ši registracija reikalinga vartotojui, kuris nori serveryje patalpinti savo duomenis ir kartu suteikiama papildoma paslauga, tai saugios failų sistemos savybės, kurias prisiregistravęs vartotojas gali suteikti savo duomenims.

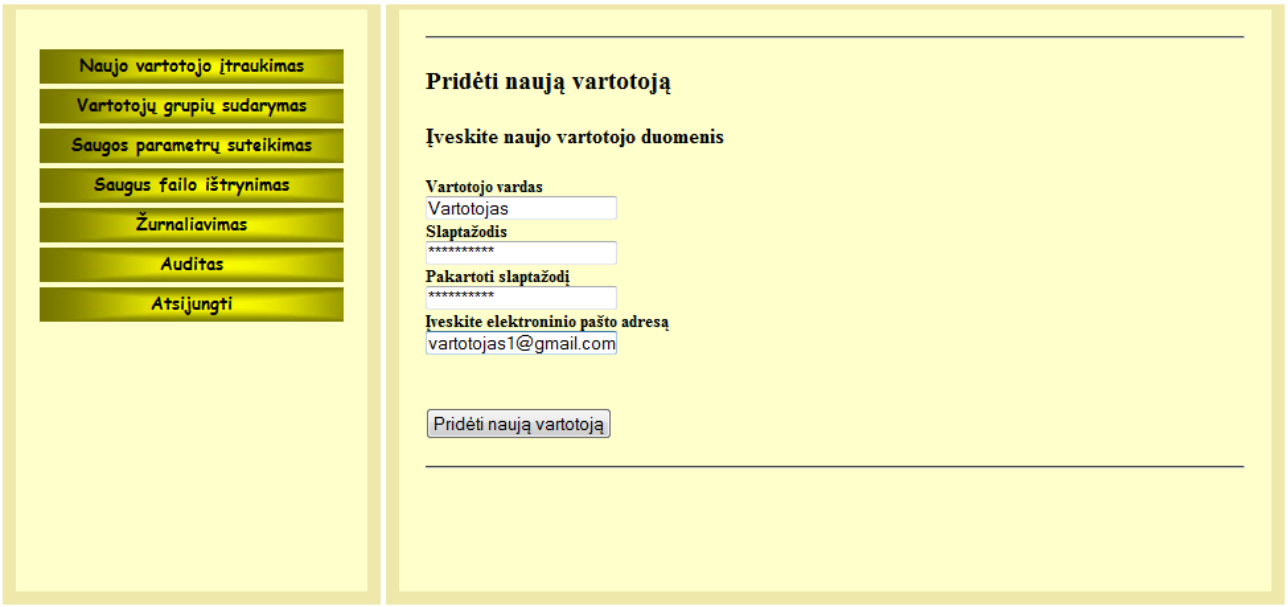
Sėkmingai užsiregistravus, vartotojas gali prisijungti prie vartotojo duomenų saugos parametrų nustatymo lango. Tam reikalinga suvesti vartotojo vardą ir slaptažodį. (25 pav.) Jei duomenys suvesti neteisingai išmetamas pranešimas "Nepavyko prisijungti, bandykite dar kartą".

25 pav. Vartotojo autentifikacija

Sėkmingos autentifikacijos metu, atsiranda kairėje pusėje didesnis meniu. Šio meniu pagalba galima pasirinkti kokia funkciją atlikti.

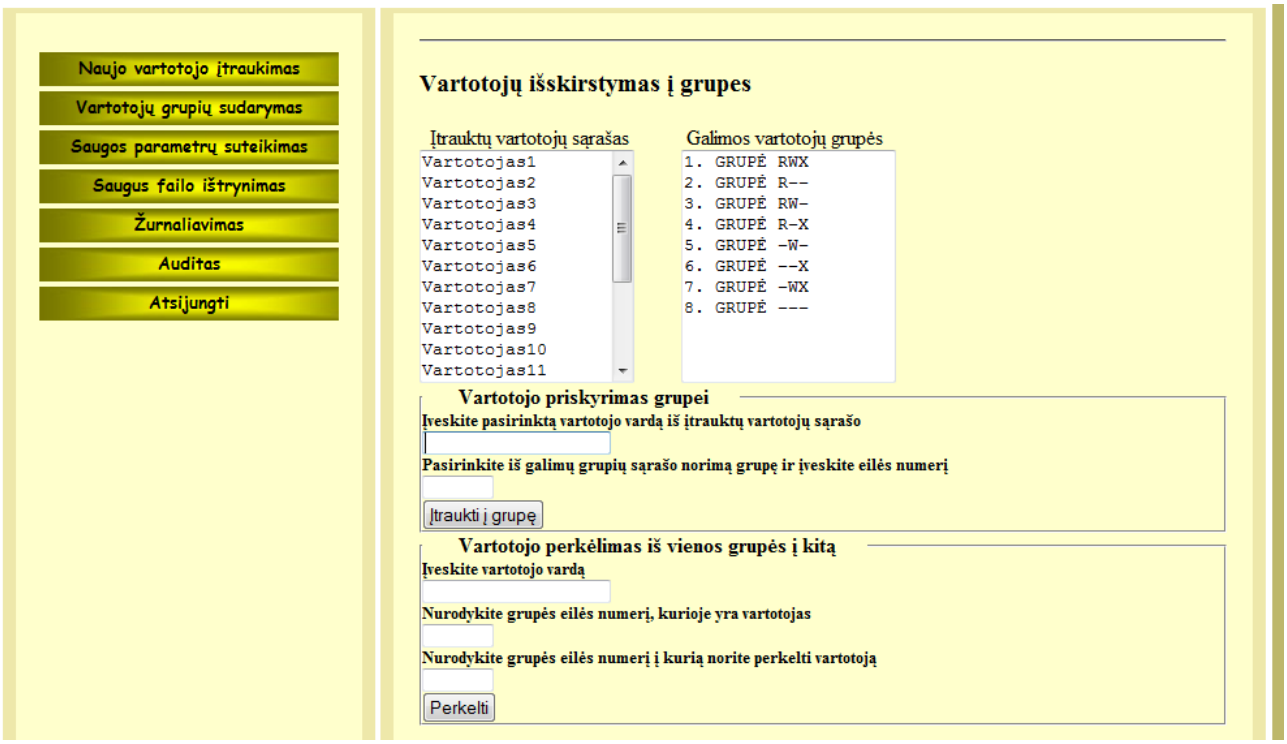
Pirmasis pasirinkimas yra "Naujo vartotojo įtraukimas" (26 pav.). Duomenų savininkas gali įtraukti naują vartotoją suteikdamas jam slaptažodį arba generuojamas atsitiktinis slaptažodis, o registracijos duomenys bus nusiusti į to vartotojo elektroninį pašto adresą.

Gavęs elektroninį laišką informacijos naudotojas galės prisijungti pagal žiniatinklio adresą prie šio vartotojo duomenų.



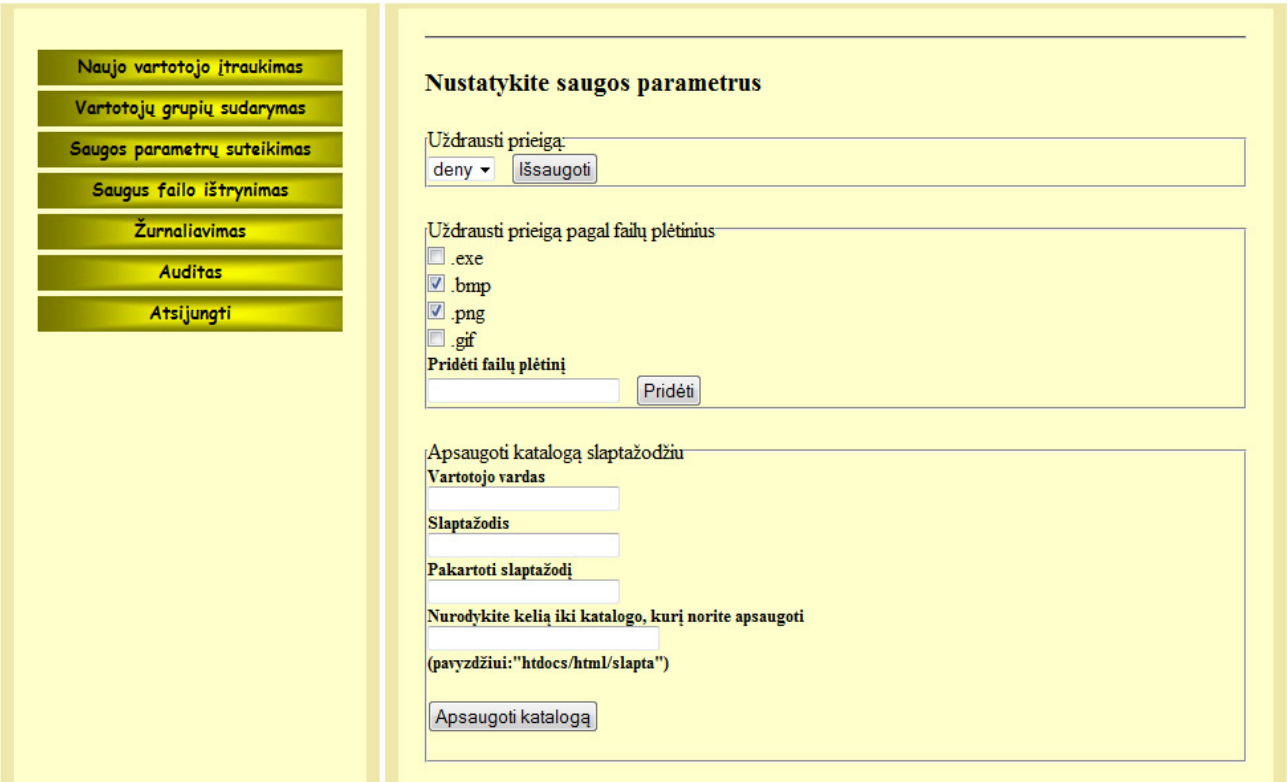
26 pav. Naujo vartotojo įdėjimas

Pridėtas naujas vartotojas gali prisijungti prie duomenų, bet teisės pagal nutylėjimą galioja tokios kokias turi pats failas ne savininko ir ne savininko grupės, o likusių vartotojų teisės. Kad būtų suteikta lankstesnė prieiga informacijos savininkas iš esamų vartotojų sąrašo suskirsto vartotojus į grupes su tam tikrais leidimais, kurie yra nekintantys (27 pav.)



27 pav. Vartotojų paskirstymas į grupes

Duomenų savininkas norėdamas nustatyti saugos parametrus gali pasirinkti trečia meniu punktą “saugos parametrų suteikimas”(28 pav.). Čia galima uždrausti prieigą pagal failų plėtinius, taip pat galima uždrausti bet kokią prieigą prie katalogų, galima apsaugoti pasirinktą katalogą slaptažodžiu. Taip pat galima drausti prieigą pagal valandas , pagal IP adresą, pagal domeną.



28 pav. Saugos parametrų nustatymas

Ketvirtame meniu punkte galime saugiai ištrinti pasirinktą failą (29 pav.). Nurodžius failą pirmiausiai jis bus užrašomas atsitiktine bitų seka ir taip bus sugadinami failo duomenys kurie saugojami failų sistemos talpykloje. Ir tik po to ištrinama rodyklė į failo turinio vietą diske. Tokiu atveju net ir panaudojus specialias programas, kurios skaito talpyklos paviršių, failas bus atkurtas, bet duomenų perskaityti nepavyks.

29 pav. Saugus failo ištrynimas

Dar viena saugos galimybė tai "žurnaliavimo" savybė (30 pav.). Galima pasirinkti įjungti arba išjungti žurnaliavimą. Įjungus žurnaliavimą prieš įrašant pakeitimus į failų sistemą duomenys saugomi į tam paskirtą failų sistemos vietą – žurnalą.

30 pav. žurnaliavimo savybės įjungimas/išjungimas

3.3. Sistemos prototipo tyrimas

Tiriamos sistemos realizuotos savybės, padaromos išvados ir apibendrinimai kokie rezultatai gauti.

3.3.1. Vartotojų registracija

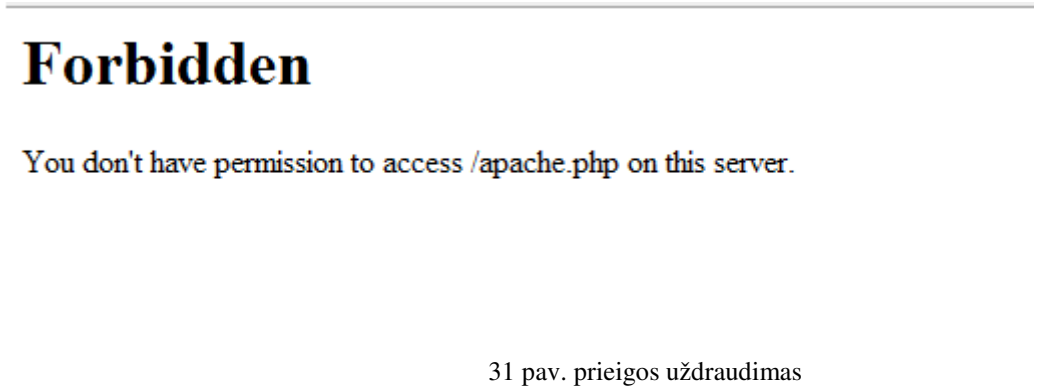
Užregistravus naujus vartotojus, slaptažodžių faile atsirado tokie įrašai:

Vartotojas1:Xt9pH2rvbUWTk
Vartotojas2:pPoy0M2TPj3Ws
Vartotojas3:l3uV8538mIN9s
Vartotojas4:GKtkLqlKCwmV2
Vartotojas5:RtG637TZAcQ5c

Matome penkių vartotojų įrašus, kurių slaptažodžiai užkoduoti MD5 algoritmu. Sukūrus naują vartotoją atsiranda naujas įrašas, tai vartotojo vardas ir per dvitaškį atskirtas vartotojo slaptažodis. Kiekvieną kartą kuriant naują vartotoją, prie esamų įrašų prisideda nauja eilutė.

3.3.2 Prieigos uždraudimas

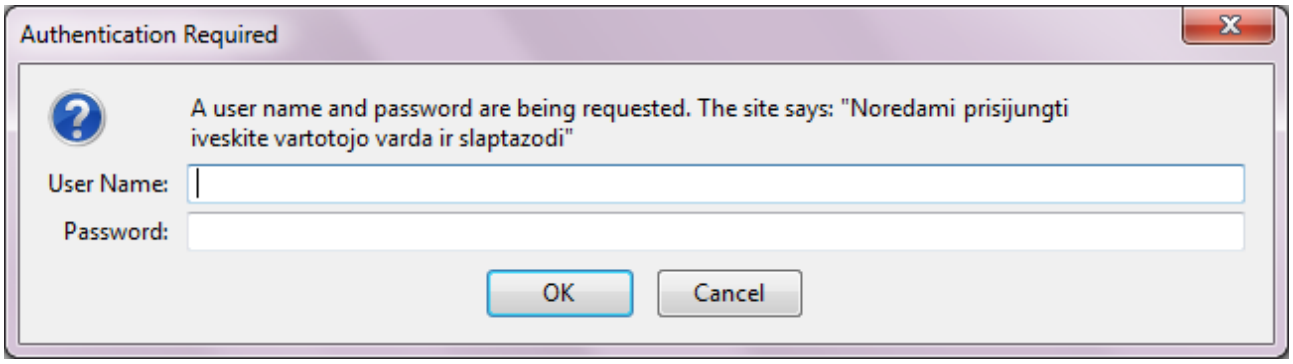
Uždraudus bet kokią prieigą naršyklėje išmetamas pranešimas (31 pav.), kad neturime prieigos teisių pasiekti šį dokumentą, Prie dokumento neprileidžiama ir informuojamas vartotojas, kodėl jis neprileidžiamas.



31 pav. prieigos uždraudimas

3.3.3. Katalogo apsauga slaptažodžiu

Apsaugojus katalogą slaptažodžiu naudojant .htpasswd ir .htaccess failiukus, norint prieiti prie šio katalogo prašoma suvesti vartotojo vardą bei slaptažodį, suvedus neteisingus duomenis pranešama, kad reikalinga autorizuotas vartotojas.



32 pav. katalogo apsauga slaptažodžiu

3.3.4. Žurnalizacijos tyrimas

Realizavus taikomajame lygmenyje saugios failų sistemos savybę – žurnalizavimą, buvo atliktas tyrimas. Tyrimo metu buvo paleistas scenarijus (script), kuris sukurdavo 500 katalogų ir juose po 3 failus, kai sukurdavo visus katalogus ir failus, tuomet juos ištrindavo.

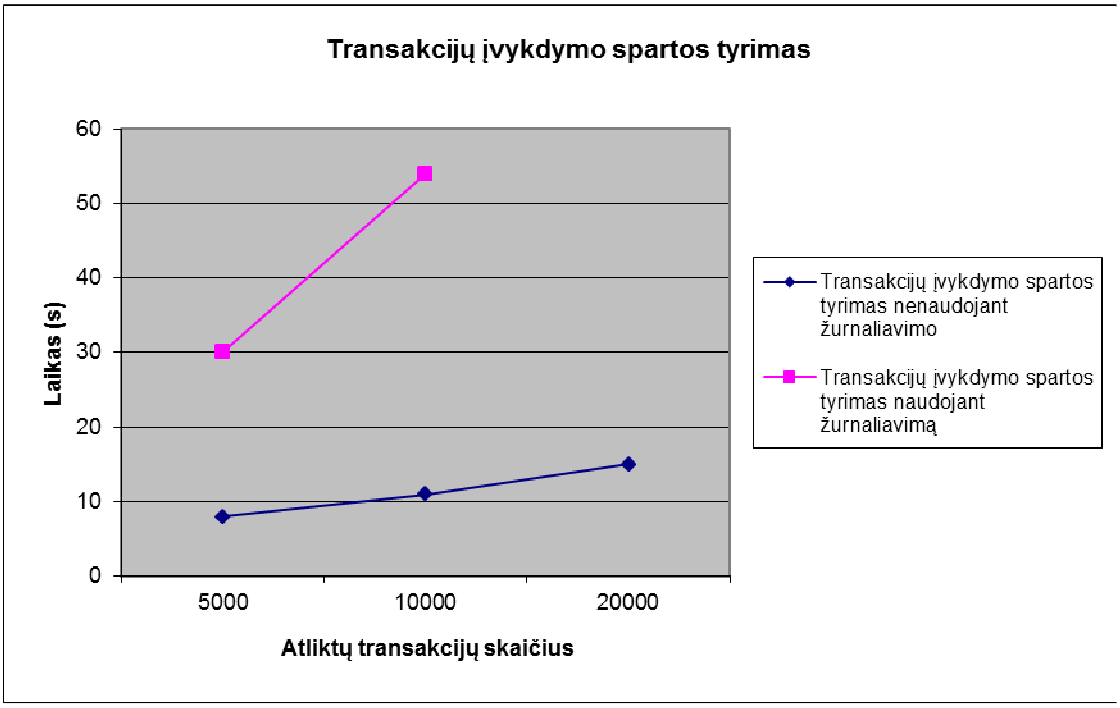
Buvo nustatomas laikas per kiek laiko įvykdys 5000, 10000 ir 20000 transakcijų įjungus žurnaliavimą ir išjungus realizavimą. Rezultatai pateikiami žemiau 5 ir 6 lentelėse, bei (31 pav.)

5 lentelė. Transakcijų įvykdymo spartos tyrimas nenaudojant žurnaliavimo

Transakcijos	Laikas(s)
5000	8
10000	11
20000	15

6 lentelė. Transakcijų įvykdymo spartos tyrimas naudojant žurnaliavimą

Transakcijos	Laikas(s)
5000	30
10000	54



31 pav. transakcijų įvykdymo spartos tyrimas

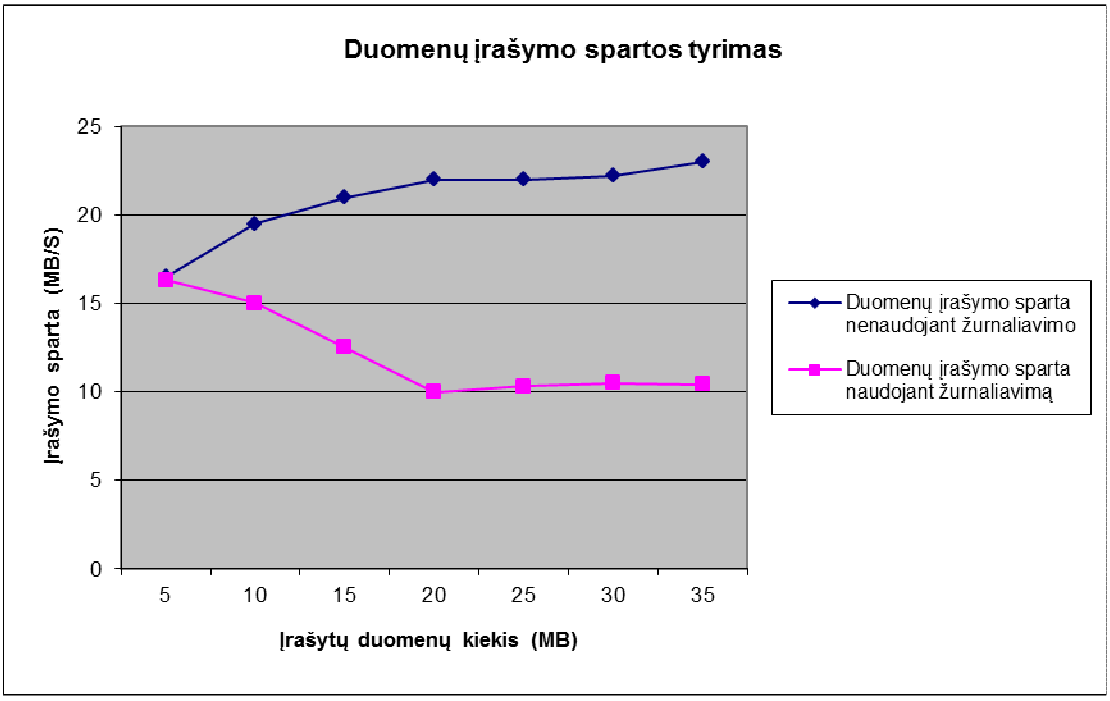
Taip pat buvo atliktas tyrimas, kuris parodo kaip kinta duomenų įrašymo sparta įjungus žurnalizaciją ir išjungus žurnalizaciją. Rezultatai pateikiami 7 ir 8 lentelėse bei (32 pav.)

7 lentelė. Duomenų įrašymo sparta nenaudojant žurnaliavimo

Sparta MB/S	Kiekis MB
16.5	5
19.5	10
21	15
22	20
22	25
22.2	30
23	35

8 lentelė. Duomenų įrašymo sparta naudojant žurnaliavimą

Sparta MB/S	Kiekis MB
16.3	5
15	10
12.5	15
10	20
10.3	25
10.5	30
10.4	35



32 pav. duomenų įrašymo spartos tyrimas

Pirmajame grafike (31 pav.) matome, kad įjungus žurnalizacijos savybę transakcijų įvykdymo laikas ženkliai išauga nuo 3,75 karto prie 5000 transakcijų įvykdymo, o prie 10000 transakcijų įvykdymo laikui, skirtumas išauga net iki 4,9 karto.

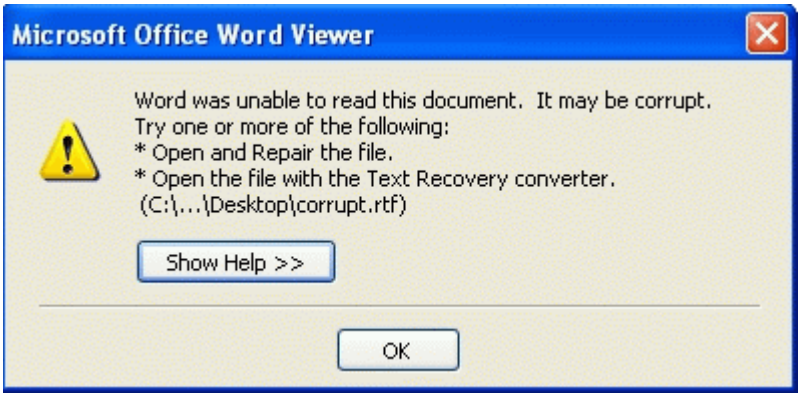
Antrame grafike (32 pav.) atvaizduojama kaip kinta duomenų įrašymo sparta didėjant įrašytų duomenų kiekiui. Matome, kad tik pradėjus įrašinėti duomenų įrašymo sparta ženkliai krinta.

3.3.5. Saugaus failo ištrynimas

Buvo sukurtas Microsoft Office Word failas ir įrašyti keli žodžiai: „Labas labas labas“.

Pirmiausiai failas buvo paprastai ištrintas naudojant standartines „Apache“ žiniatinklio funkcijas, tuomet panaudojant duomenų atstatymo programą failas buvo sėkmingai atkurtas ir duomenis pavyko perskaityti.

Panaudojus saugaus failo ištrynimo savybę failas buvo ištrintas užrašant jį atsitiktine bitų seka. Panaudojus duomenų atstatymo programą failas buvo rastas, bet norint jį atidaryti ir peržiūrėti jame įrašytą informaciją gavome pranešimą, kad negalima perskaityti failo turinio nes failas yra sugadintas (pav. 33)



33 pav. pranešimas jog failas sugadintas

3.4. Išvados

Realizuotoje saugioje „Apache“ žiniatinklio serverio failų sistemoje yra tokios savybės:

1. Realizuotoje saugios failų sistemos grafinėje sąsajoje, pagrindinis vaidmuo nustatant duomenų saugos savybes tenka informacijos savininkui, o informacijos naudotojas tik naudojasi informacija su savininko nustatytom teisėm. Šitaip daug lanksčiau galima užtikrinti saugios failų sistemos savybes, nes kiekvienas informacijos savininkas yra labiau suinteresuotas apsaugoti savo duomenis.
2. Realizuota vartotojo registracija, vartotojo autentifikacija, galimybė įtraukti naują vartotoją, bei grupių sudarymas.
3. Realizuojamos standartines *.htaccess* failo savybės užtikrinančios prieigos apribojimus, bei katalogų apsauga reikalaujant slaptažodžio.
4. Realizuotas žurnaliavimas bei saugus failo ištrynimas.
5. Vartotojui suteikta galimybė paprasčiau nustatyti saugos parametrus savo duomenims, nes dažną vartotoją, kuris nori apsaugoti savo duomenis atbaido sudėtingas saugos nustatymų būdas konfigūruojant failus ir rašant komandines eilutes. Kuo aiškiau bus galima nustatyti saugos parametrus, tuo labiau išnaudojamos „Apache“ saugios failų sistemos savybės ir vartotojo duomenys bus saugesni.

Eksperimente buvo ištirta, kad:

1. Naudojant žurnalizacijos savybę sumažėja ženkliai ir duomenų įrašymo sparta ir per tam tikrą laiką galimų įvykdomų transakcijų skaičius. Šią savybę norint naudoti, kur aptarnaujamas didesnis kiekis vartotojų ir yra mažesni serverio resursai, reikėtų naudoti atsargiau. Geriau šią savybę naudoti tokiuose serveriuose, kuriuose yra didesni resursai.
2. Ištrinant failą paprastom „Apache“ žiniatinklio serverio failų sistemos savybėm, duomenis pavyko atstatyti su specialiom duomenų atstatymo programom ir pavyko perskaityti faile esančią informaciją, ištrynus failą užrašant jį atsitiktine bitų seka, failą duomenų atstatymo programa rado, bet jo paleisti ir nuskaityti jame esančios informacijos nepavyko, nes failas buvo sugadintas.

4. MD bendros išvados.

Failų sistemų analizėje išanalizuotos įvairios failų sistemų saugos savybės ir nustatyta, kokios savybės turi būti realizuotos norint, kad failų sistema būtų saugi.

„Apache“ žiniatinklio serverio failų sistemos analizėje, buvo ištirtos šio serverio failų sistemos galimybės ir palygintos su kitų failų sistemų saugos galimybėmis ir nustatyta kad „Apache“ žiniatinklio serverio failų sistemos saugos galimybės yra ribotos. Siekiant praplėsti ribotas „Apache“ žiniatinklio serverio failų sistemos galimybes ir padidinti saugumą, buvo pridėtos tokios papildomos savybės:

1. pridedama žurnalizacijos savybė realizuota taikomajame lygmenyje
2. pridedamas šifravimas duomenims , kurie saugojami serverio talpyklose
3. pridedama saugaus failo ištrynimas apsaugant nuo duomenų ištraukimo „iš šiukšlių“.
4. suteikiamas lankstesnis vartotojų prieigos valdymas

Projektinėje dalyje buvo suprojektuotas saugios „Apache“ žiniatinklio serverio failų sistemos modelis, iškelti reikalavimai kaip turi atrodyti saugi serverio failų sistema, aprašyta kokie bus naudojami moduliai bei konfigūraciniai failai.

Realizuotoje grafinėje sąsajoje, suteikiama galimybė duomenų savininkui pačiam suteikti saugios failų sistemos savybes savo duomenims apsaugoti ir pats savininkas riboja vartotojų prieigą ir nustato ką informacijos naudotojas galės su tais duomenimis atlikti.

Eksperimentinėje dalyje ištirtos realizuotos savybės. Nustatyta, kad žurnaliavimo savybė gana ženkliai apkrauna serverį, sumažėja įvykdomų transakcijų kiekis per tam tikrą laiką, bei krenta duomenų įrašymo sparta, todėl šią savybę geriau naudoti serveriuose atsižvelgiant į jų techninius resursus, bei serverio aptarnaujamų vartotojų kiekį.

Terminų ir santrumpų žodynas

HTML (Hyper text Markup Language „Hiperteksto žymėjimo kalba“) – tai kompiuterinė žymėjimo kalba, naudojama pateikti turinį internete.

CSS (***angl.*** Cascading Style Sheets) – kalba, skirta nusakyti kita struktūrine kalba aprašyto dokumento vaizdavimą. Dažniausiai CSS aprašomas ***HTML*** dokumentų pateikimas

PHP – plačiai paplitusi dinaminė interpretuojama programavimo kalba (en: Hypertext Preprocessor), sukurta 1997 m. ir specialiai pritaikyta interneto svetainių kūrimui.

IP adresas – kompiuterio identifikatorius IP tinkluose. Tai tam tikrame tinkle unikalus skaičius, naudojamas vienareikšmei duomenų paketo siuntėjo ir gavėjo identifikacijai ir skiriamas žmogaus ar organizacijos

SSL - (Secure Sockets Layer) - Kriptografinis protokolas, skirtas informacijos, sklindančios internete apsaugojimui šifruojant SSL šifravimui naudoja tiek simetrinę, tiek ir asimetrinę kriptografiją.

HTTPS - specialus serverio protokolas, kuris, norint apsaugoti vartotojus, užkoduoja konfidencialią užsakymo informaciją.

MD5 - (Message-Digest algorithm 5) – žinutės santraukos algoritmas, plačiai naudojama kriptografijos maišos funkcija su 128 bitų (16 baitų) maišos reikšme.

EFS - Koduojamųjų failų sistema (EFS) yra Windows funkcija, kuri leidžia įrašyti informaciją į standųjį diską užšifruotu formatu. Šifravimas yra griežčiausia apsauga, kurią pateikia sistema Windows, kad apsaugotų informaciją.

5. Literatūra

1. .htaccess [interaktyvus], [žiūrėta 2009-12-10]. Prieiga per internetą: http://www.freewebmasterhelp.com/tutorials/htaccess/
2. .htaccess galimybių panaudojimas [interaktyvus], [žiūrėta 2009-11-17]. Prieiga per internetą: http://www.coders.lt/straipsniai/apache_.htaccess_galimybes_panaudojimas
3. AOL [interaktyvus], [žiūrėta 2010-03-05]. Prieiga per internetą: http://aolserver.sourceforge.net/
4. Apache [interaktyvus], [žiūrėta 2010-03-17]. Prieiga per internetą: http://en.wikipedia.org/wiki/Apache_HTTP_Server
5. Apache sauga [interaktyvus], [žiūrėta 2010-03-17]. Prieiga per internetą: http://www.coders.lt/straipsniai/
6. Cambel, F. WEB server system and method United States Patent, 2004
7. Delaney, P. Method for the acceleration and simplification of file system logging techniques using storage device snapshots. United States Patent, 2001.
8. htpasswd [interaktyvus], [žiūrėta 2009-12-10]. Prieiga per internetą: http://httpd.apache.org/docs/2.0/programs/htpasswd.html
9. httpd, the Apache HTTP Server Configuration [interaktyvus], [žiūrėta 2009-10-17]. Prieiga per internetą: http://svnbook.red-bean.com/en/1.0/ch06s04.html
10. Hughes, P. Architecture of Secure File System, 2000
11. Klientų identifikavimas [interaktyvus], [žiūrėta 2009-05-17]. Prieiga per internetą: http://orion.kobra.ktu.lt/kraitis/1/autentifikacija.html
12. Metaduomenys [interaktyvus], [žiūrėta 2010-04-10]. Prieiga per internetą:http://www.straipsniai.lt/Hakeriai/puslapis/8341
13. mod_auth [interaktyvus], [žiūrėta 2010-05-01]. Prieiga per internetą: http://httpd.apache.org/docs/2.0/mod/mod_auth.html
14. Permission [interaktyvus], [žiūrėta 2009-11-24]. Prieiga per internetą: http://www.abysssec.com/blog/2008/10/is-your-apache-in-safe-mode/
15. Protecting Data by Using EFS to Encrypt Hard Drives [interaktyvus], [žiūrėta 2009-09-10]. Prieiga per internetą: http://technet.microsoft.com/en-us/library/cc875821.aspx
16. RISTIC, I. *Apache Security.* United States of America, 2005. 398 p. ISBN 978-0-596-00724-9

17. SSL/TLS [interaktyvus], [žiūrėta 2010-03-17]. Prieiga per internetą: http://hadoop.apache.org/core/docs/current/hdfs_permissions_guide.html
18. The Apache Directory Structure [interaktyvus], [žiūrėta 2009-05-17]. Prieiga per internetą: http://devdaily.com/unix/edu/UnixSysAdmin/node169.shtml
19. VENČKAUSKAS, A.; IR TOLDINAS, E. *Kompiuterių ir operacinių sistemų sauga 2008. 198 p.*
20. Zeus [interaktyvus], [žiūrėta 2010-02-05]. Prieiga per internetą: http://www.zeus.com/