LATVIJAS UNIVERSITĀTE
DATORIKAS FAKULTĀTE

PROGRAMMATŪRAS „ĀRSTA BIROJS 3” DROŠĪBAS RISINĀJUMI

BAKALAURA DARBS

Autors: **Dārta Krampe**
Stud. apl. dk06022
Darba vadītājs: Dr. Dat. Darja Šmite

RĪGA 2010

ANOTĀCIJA

Mūsdienās informācijas tehnoloģijas tiek plaši izmantotas jebkādas nozares organizācijās, bieži biznesa darbība ir atkarīga no kādas datorizētas sistēmas. Veselības aprūpes iestādes apstrādā sensitīvus datus, tādēļ aktuāla problēma ir pacienta datu drošība. „Ārsta Birojs 3" ir Latvijā izplatītākā slimnīcas informācijas sistēma, kas tiek izmantota vairākās Latvijas slimnīcās un poliklīnikās, kā arī citās veselības aprūpes iestādēs.

Šajā darbā ir apkopoti starptautiski atzītie informācijas un programmatūras drošības standarti, ieteikumi un vadlīnijas. Tiek sniegts novērtējums programmatūras „Ārsta Birojs 3" drošības risinājumu atbilstībai šiem standartiem un pieņēmumiem. Rezultātā izstrādāti vispārīgi ieteikumi paaugstinātas drošības programmatūras izstrādei.

Atslēgvārdi: programmatūras drošība, informācijas drošības standarti.

ABSTRACT

Information technology is widely used in any industry as business often depends on a computerized system. Health care institutions handle sensitive data, so the patient data security problem is acute. "Doctor's office 3" is widespread information system for health care institutions, which is used in most hospitals and clinics as well as other health care institutions in Latvia.

This paper brings together popular and widely used information and software security standards, recommendations and guidelines. Evaluation of software "Doctor's office 3" security solutions is given, complying with these standards and assumptions. As a result, general recommendations for increased security software development are provided.

Keywords: software security, information security standards.

SATURS

APZĪMĒJUMI UN SAĪSINĀJUMI **5**
IEVADS **6**
1. PROGRAMMATŪRAS DROŠĪBA **8**
1.1. INFORMĀCIJAS DROŠĪBA 9
1.1.1. ISO/IEC 17799:2005 ..... *10*
1.1.2. ISF *14*
1.1.3. ITIL *17*
1.1.4. Programmatūras drošības nodrošināšana *17*
1.1.5. CobIT *18*
1.1.6. LR normatīvie akti *21*
1.2. KOPSAVILKUMS 22
2. DROŠĪBAS PĀRSKATI **25**
2.1. IEEE PROGRAMMATŪRAS APSKAŠU STANDARTS 25
2.2. ISACA IETEIKUMI 25
2.3. PERSONAS DATU APSTRĀDES SISTĒMU AUDITA ROKASGRĀMATA 26
2.4. OSSTMM ROKASGRĀMATA 28
2.5. KOPSAVILKUMS 28
3. RISKU PĀRVALDĪBA **30**
3.1. RISKU PĀRVALDĪBAS METODES 30
3.2. RISKU ANALĪZES METODES 34
4. PROGRAMMATŪRAS „ĀRSTA BIROJS 3" DROŠĪBA **35**
4.1. DROŠĪBAS PĀRBAUDE 36
4.1.1. Programmatūras „Ārsta Birojs 3" risku pārvaldība *36*
4.1.2. Informācijas un datu riski *37*
4.1.3. Programmatūras ieviešanas un uzturēšanas riski *42*
4.1.4. Izstrādes procesa riski *44*
4.1.5. Cilvēka radītie riski *46*
4.2. KOPSAVILKUMS 47
NOBEIGUMS **50**
IZMANTOTĀ LITERATŪRA **51**

PIELIKUMI..... 55

APZĪMĒJUMI UN SAĪSINĀJUMI

ĀB3 – Programmatūra „Ārsta Birojs 3”

VNC - Veselības norēķinu centrs

VEC – Veselības ekonomikas centrs

IT – Informācijas tehnoloģijas

CEN – (*European Committee for Standardization*) Eiropas standartizācijas komiteja

ASTM – (*American Society for Testing and Materials*) pasaulē atzīta standartu izstrādes kompānija

ISF – (Information Security Forum) Informācijas drošības forums izstrādā vadlīnijas informācijas drošībai

ITIL – (*Information Technology Infrastructure Library*) Informācijas tehnoloģiju infrastruktūras bibliotēka

CobIT - (*Control Objectives for Information and related Technology*) pasaulē atzīta informācijas un tehnoloģiju kontroles rokasgrāmata

SSA – (*Software Security Assurance*) Programmatūras drošības nodrošināšana

IEEE – (*Institute of Electrical and Electronics Engineers*) Elektriķu un elektroniku inženieru institūts

ISACA – (*Information Systems Audit and Control Association*) Informācijas sistēmu audita un kontroles asociācija

CISA – (*Certified Information Systems Auditor*) Sertificēts informācijas sistēmu auditors

OSSTMM – (*Open Source Security Testing Methodology Manual*) Atvērtā koda drošības testēšanas metodoloģijas rokasgrāmata

SQL - (*Structured Query Language*) Strukturēto vaicājumu valoda

IEVADS

Mūsdienās informācijas tehnoloģijas tiek plaši izmantotas jebkādas nozares organizācijās, bieži biznesa darbība ir atkarīga no kādas datorizētas sistēmas. Aizvien vairāk veselības aprūpes iestādēs pacientu datu pārvaldībai tiek izmantotas informāciju sistēmas. Veselības aprūpes iestādes apstrādā sensitīvus datus, tādēļ aktuāla problēma ir pacienta datu drošība. Tā kā datus apstrādā, izmantojot informācijas sistēmu, par pacienta datu drošību rūpējas ne tikai iestādes darbinieki, veicot fiziskos drošības pasākumus, bet arī pati sistēma.

„Ārsta Birojs 3" ir Latvijā izplatīta veselības aprūpes iestādes informācijas sistēma, kas tiek izmantota vairākās Latvijas slimnīcās un poliklīnikās, kā arī citās veselības aprūpes iestādēs. Programmatūra nodrošina veselības aprūpes iestādes ietvaros esošās informācijas centralizētu uzkrāšanu un apriti elektroniskā formātā. Tādā veidā tiek samazināts administratīvo darba uzdevumu apjoms, kā arī atvieglota informācijas atkārtota izmantošana un pieejamība veselības aprūpes iestādes ietvaros. Programmatūra tiek uzturēta atbilstoši VNC un VEC normatīvajiem dokumentiem un datu apmaiņas kārtībai. Pilnu programmatūras „Ārsta Birojs 3" funkcionalitāti skatīt 1.pielikumā.

Šī darba mērķis ir apkopot esošus programmatūru drošības standartus un ieteikumus, novērtēt programmatūras „Ārsta Birojs 3" drošības atbilstību šiem standartiem un pieņēmumiem, kā arī izstrādāt vispārējus ieteikumus paaugstinātas drošības programmatūru izstrādei.

Lai sasniegtu mērķi, tika izvirzīti sekojoši uzdevumi:

- Iepazīties ar likumdošanu, kas attiecas uz veselības aprūpes programmatūras drošību,
- Iepazīties ar starptautiski pieņemtiem programmatūras drošības aizsardzības standartiem un ieteikumiem,
- Iepazīties ar vadlīnijām, kā veicama programmatūras drošības pārbaude,
- Veikt programmatūras „Ārsta Birojs 3" drošības pārbaudi, ņemot vērā iepriekš analizētos drošības pārbaudes ieteikumus, kā arī noteikt drošības risinājumu atbilstību programmatūras drošības standartiem un ieteikumiem,
- Apkopot vispārējus ieteikumus drošu programmatūru izstrādei.

Tā kā programmatūra satur un apstrādā datus, tad tai būtu jāparūpējas par datu drošību. Datu drošība ir datu aizsargātība pret tīšu vai netīšu nesankcionētu rīcību, kas var radīt to modificēšanu, atklāšanu vai bojāšanu (20).

Veselības aprūpes programmatūrās datu drošības jēdziens parasti tiek paplašināts, jo programmatūra apstrādā arī sensitīvus datus. Aktuāls ir jautājums, vai cilvēks ziņkārībā var piekļūt sava kaimiņa slimības vēstures datiem? Var gadīties, ka slimnīcas darbinieks nejauši izdzēš pacienta nosūtījumu pie ķirurga, kā šajā situācijā var palīdzēt programmatūra? Eksistē vairāki pieņēmumi, kas raksturo veselības aprūpes informācijas drošību.

Veselības aprūpes ārsts nedrīkst atklāt pacienta veselības stāvokli vai ārstēšanas metodes, vai jebkurus citus pacienta datus, kas tiek iegūti aprūpes laikā, citām personām (12, 27. lpp).

Mūsdienās ir pieņemts, ka pacientu aprūpē ne tikai viens ārsts, bet ārstu grupa, šādā gadījumā jākontrolē informācijas apmaiņa tikai starp šiem grupas biedriem. Tātad ir jāparūpējas par to, ka cilvēks nevar iegūt veselības stāvokļa informāciju, piemēram, par kādu interesējošu personu.

Programmatūrai jāspēj nodrošināt augstu veselības informācijas kvalitātes līmeni (integritāti un precizitāti) (12, 40. lpp).

Šis pieņēmums ir svarīgs attiecībā uz pacienta veselību. Jāpievērš uzmanība tam, ka atkarībā no programmatūras datiem, tiek veikta pacienta ārstēšana, līdz ar to, nekorektu datu gadījumā, var tikt apdraudēta pacienta dzīvība.

Darbs ir organizēts sekojošās nodaļās: pirmā nodaļa apskata un apkopo esošus programmatūras drošības standartus un ieteikumus, otrajā nodaļā tiek analizēti programmatūras drošības pārbaudes ieteikumi, trešā nodaļā tiek apskatītas risku pārvaldības metodes, ceturtā nodaļa apraksta programmatūras „Ārsta Birojs 3" drošības risinājumu atbilstību iepriekš apskatītajiem drošības standartiem un pieņēmumiem, balstoties uz risku analīzes rezultātā identificētajiem apdraudējumiem, kā arī tiek sniegti ieteikumi programmatūras drošības uzlabošanai.

1. PROGRAMMATŪRAS DROŠĪBA

Programmatūras drošības jēdziens ir attiecināms tieši uz programmatūras izstrādātāju, kuram jāparedz iespējamie programmatūras apdraudējumi un jāsamazina to iespējamība, lai sistēma turpinātu darbību pat tad, kad ir noticis apdraudējums. Šis jēdziens sabiedrībā ir parādījies salīdzinoši nesen, kad 2001. gadā tika publicētas pirmās grāmatas un raksti, kas pievērsa uzmanību tieši nepieciešamībai pētīt programmatūras drošību (3, 80. lpp).

Informācijas sistēma sevī ietver ne tikai programmatūru, bet arī tīkla infrastruktūru, aparatūras un to saziņu, datubāzes un visbeidzot arī lietotājus, līdz ar to bieži tiek piemirsts pievērst uzmanību tieši programmatūras drošībai, kā atsevišķai drošības novērtēšanas daļai. Lai gan pasaulē ar programmatūras drošības jēdzienu var saprast dažādas lietas, cilvēki, kas nav saistīti ar IT jomu, pēc darba autores pieredzes, bieži vien uzskata, ka programmatūras drošība ir pētāma tikai tad, kad tā jau ir izstrādāta un ieviesta, lai parūpētos par datora un datu drošību, nepieciešams tikai ugunsmūris un pretvīrusu programmatūra. Tomēr programmatūras drošība ir atkarīga no tās plānošanas, izstrādes, ieviešanas un visbeidzot uzturēšanas procesa, tātad programmatūras drošība ir jānodrošina visā programmatūras izstrādes dzīves ciklā (3, 81. lpp).

Tā kā drošība tiek pārraudzīta visā programmatūras izstrādes dzīves ciklā, ir nepieciešams plānot drošības kontroles mehānismus, kuri būs nepieciešami. Drošības plānošana tiek veikta programmatūras prasību definēšanas fāzē, kur nepieciešams ietvert arī drošības prasības. Jāpievērš uzmanība tam, ka nepieciešams specificēt ne tikai fizisko, bet arī funkcionālo drošību. Ja netiek uzstādītas konkrētas un nepārprotamas drošības prasības, programmatūras drošība var tikt apdraudēta.

Pēc Gary McGraw domām programmatūras izstrādes etapā nepieciešams izmantot risku analīzes metodi, kas palīdz definēt iespējamos apdraudējumus (3, 83.lpp). Šādā veidā var tikt uzsākta nepieciešamo drošības pasākumu plānošana un arī izstrāde. Jāpiemin, ka risku analīze ir cikliska, tā tiek atkārtota visā programmatūras izstrādes dzīves ciklā, tādā veidā aktualizējot iespējamos drošības apdraudējumus. Svarīgi rūpēties arī par drošu kodu – ieteicams veikt arī koda pārskatus.

Testēšanas fāzē bieži vien tiek veikti funkcionālie testi, tomēr arī drošības testi ir nepieciešami. Testētājam jādomā kā „uzbrucējam", jātestē visus risku analīzes rezultātā identificētos apdraudējumus (5, 50. lpp). Tomēr caurlaidības testiem nevajadzētu būt vienīgajiem, jo tie aptver nepilnīgu intervālu - netiek apskatīti visi programmatūras riski, savukārt, ja kāds no caurlaidības testiem dod negatīvu rezultātu, to var uzskatīt par kritisku kļūdu (3, 82. lpp).

1.1. Informācijas drošība

Informācijas drošība tiek nodrošināta ar trīs pamatjēdzienu palīdzību: konfidencialitāte, integritāte un pieejamība (26). Konfidencialitāte ir informācijas aizsardzība no neautorizētas piekļuves un informācijas atklāšanas. Integritāte ir informācijas aizsardzība no neautorizētas vai nejaušas informācijas mainīšanas. Pieejamība attiecas uz informācijas pieejamību vienmēr, kad tā ir nepieciešama lietošanai.

Informācijas drošības galvenais apdraudētājs ir cilvēks. Bieži vien drošības apdraudēšana ir netīša vai neapdomāta (paroles nodošana it kā neļaunprātīgai personai, paroles glabāšana tuvu darba stacijai, piemēram, uz piezīmju lapiņas, kas piestiprināta pie monitora). Tomēr lietotājs var kalpot arī par drošības incidenta ziņotāju un drošības uzraudzītāju, kā arī bieži vien lietotājs spēj ieteikt drošības risinājumus un uzlabojumus (37).

Šobrīd eksistē divas organizācijas CEN un ASTM, kas visaktīvāk izstrādā standartus veselības aprūpes informācijas sistēmām. CEN/prENV12251: Droša lietotāja identifikācija veselības aprūpes sistēmām ir ieteikuma formas standarts, kura mērķis ir uzlabot autentifikācijas procesu. Standarta autori atdzīst, ka drīz autentifikācijas procesu, kurā izmanto paroles, aizstās elektroniskās kartes vai biometriskās ierīces, tomēr, kamēr tas vēl nav tik izplatīti, ir svarīgi rūpēties par autentifikācijas drošību. Šis standarts norāda uz vairākām prasībām, kas būtu jāizpilda, lai nodrošinātu autentifikācijas procesa drošību (13, 15. lpp):

- Lietotāja unikalitātes nodrošināšana,
- Droša piekļuves procedūra,
- Droša paroles glabātuve un glabāšanas veids,
- Paroles lietošanas termiņa ierobežojumi,

- Nepieļaut atkārtotu paroles lietošanu,
- Paroles stipruma kontrole.

ASTM/E1896-97: Veselības aprūpes informācijas konfidencialitātes privātuma piekļuves un datu drošības principu standarts, iekļaujot datorizētus pacienta veselības datus ir standarts, kas sniedz vadlīnijas pacienta veselības datu konfidencialitātes, privātuma, piekļuves un drošības nodrošināšanai (13, 17. lpp). Šis standarts var tikt pielietots ne tikai elektroniskā datu apstrādes procesā, bet arī papīra veselības kartiņu drošībai.

ASTM/E1988:98: Personu, kurām ir piekļuve veselības informācijai, apmācības standarts sniedz vadlīnijas, kā veicama personāla apmācība, lai nodrošinātu veselības informācijas konfidencialitāti, privātumu un drošību (13, 18. lpp). Apmācības pamatā ir tiešie darba pienākumi.

Eksistē vairāki starptautiski atzīti standarti un vadlīnijas, kas raksturo informācijas drošību un var palīdzēt izprast, kāda ir droša programmatūra. Tālākajās apakšnodaļās apskatīti un analizēti daži no tiem.

1.1.1. ISO/IEC 17799:2005

ISO/IEC 17799:2005 ir starptautiski atzīts Informācijas drošības pārvaldības standarts. Standarts piedāvā vadlīnijas, kas palīdzēs definēt, izstrādāt, uzturēt un uzlabot drošības pārvaldību organizācijā (17, 13. lpp). Standarts piedāvā 39 drošības kategorijas, kas apvienotas 11 drošības kontroles sadaļās (17, 18. lpp):

- Drošības politika,
- Informācijas drošības organizēšana,
- Aktīvu pārvaldība, kas ietver informācijas klasificēšanu, definējot tās vērtību, prasības, sensitivitāti un nozīmīgumu,
- Cilvēkresursu drošība,
- Fiziskā un vides drošība,
- Komunikāciju un operāciju pārvaldība,
- Pieejas kontrole,
- Informāciju sistēmas pieņemšana, izstrāde un uzturēšana,
- Informācijas drošības starpgadījumu pārvaldība,
- Biznesa nepārtrauktības pārvaldība,
- Atbilstība.

Dokumenta struktūra ietver katras drošības sadaļas mērķa definīciju, nepieciešamo kontroles mehānismu un ieviešanas vadlīnijas, kā arī citu papildus informāciju.

Šis standarts piedāvā drošības pārvaldību uzsākt ar risku novērtēšanu. Risku pārvaldība ir pamats, kas nosaka, kāda drošības kontrole nepieciešama, atkarībā no riska prioritātes. Risku prioritāšu novērtējumi parasti balstās uz organizācijas iepriekšējo pieredzi, kā arī risku atbildes ir atkarīgas no biznesa pieņēmumiem un organizācijas apstākļiem. Risku pārvaldības metodes aprakstītas šī darba 3. nodaļā.

Lai norādītu uz programmatūras drošības jēdziena saistību ar šo standartu, turpmākajās apakšnodaļās sīkāk apskatītas dažas no standartā piedāvātajām drošības kontroles sadaļām.

1.1.1.1. Drošības politika

Drošības politikas mērķis ir uzstādīt pārvaldības virzienu un informācijas drošības atbalstu atbilstoši biznesa prasībām un piesaistītajiem likumiem un ierobežojumiem. Drošības politikai vajadzētu izstrādāt dokumentu, kurš būtu apstiprināts un publiski pieejams darbiniekiem un iesaistītajām personām (17, 7. lpp).

Drošības politiku ir jāpārskata regulāri plānotos intervālos, vai arī gadījumos, kad konstatētas izmaiņas, lai nodrošinātu politikas piemērotību, adekvātumu un efektivitāti. Drošības politikas dokumentācijai ir jābūt aktuālai, lai tā spētu nodrošināt efektīvu drošības pārvaldību (17, 8. lpp).

Darba autore uzskata, ka drošības plānošana ir nepieciešama arī programmatūras drošībai, līdz ar to ir ieteicams izstrādāt drošības pārvaldības plānu un, vadoties pēc tā, veikt programmatūras izstrādi. Tomēr šī darba ietvaros laika ierobežojumu dēļ drošības politika netiks izstrādāta.

1.1.1.2. Cilvēkresursu drošība

Cilvēkresursu drošības aspekts ir ļoti nozīmīgs, lai nodrošinātu to, ka darbinieki, vadība un iesaistītās personas apzinātos piešķirtās atbildības un ir piemēroti lomām, lai mazinātu zādzības, krāpšanas riskus vai ļaunprātīgas informācijas izmantošanas iespējamību (17, 35. lpp).

Šīs drošības kontroles ietver atbildību un lomu definēšanu un dokumentēšanu, kā arī regulāru darbinieku atbilstības novērtēšanu. Drošības kontroles laikā jānovērtē

darbinieku un iesaistīto personu informētība par drošības pasākumu ievērošanu, kā arī jānovērtē personu informācijas pieejamības līmeni. Ieteicams veikt drošības apmācības, darbinieku disciplinēšanu un izstrādāt motivējošas programmas.

Tā kā programmatūras drošību būtiski ietekmē tās lietotājs, pēc darba autores domām, pētot programmatūras drošības līmeni, nepieciešams apskatīt arī cilvēkresursu drošības ieviestās kontroles.

1.1.1.3. Komunikāciju un operāciju pārvaldība

Komunikācijai un operāciju procesiem ir ļoti būtiska nozīme informācijas sistēmas darbībā. Informācijas sistēmai ir jānodrošina komunikācija, kurai būtu zema ievainojamības pakāpe. Visām procedūrām ir jābūt dokumentētām, dokumentā iekļaujot rezerves kopiju apstrādes procesu, auditēšanas pierakstu procesu, sistēmas žurnālu pārvaldību (17, 37. lpp).

Pienākumiem jābūt sadalītiem, lai mazinātu iespējamību nejaušai informācijas labošanai. Tātad, nepieciešama informācijas vienību pieejas kontrole, ko parasti panāk ar lietotāju lomu un tiesību sadalījumu.

Lai nodrošinātu informācijas integritāti un pieejamību, jāplāno rezerves kopiju veidošana un uzturēšana. Rezerves kopiju veidošanai jābūt regulārai un kopijas fiziski jānovieto atdalītās sistēmas daļās vai pat uz attālinātiem serveriem (17, 44. lpp). Jābūt izstrādātam informācijas atjaunošanas plānam, kurš izmanto rezerves kopijas.

Sistēmas lietošanas auditēšana ir nepieciešama, lai novērstu ļaunprātīgu un nelikumīgu informācijas izmantošanu, kā arī nejaušu informācijas zaudēšanu. Auditācijas pierakstiem jāsniedz informācija par lietotāju – identifikators, laiks, kad darbība veikta, veiktās darbības raksturojums, izmantotās datnes (17, 55. lpp). Nepieciešams plānot un ieviest efektīvu sistēmas kļūdu apstrādi, sākot ar pietiekamu kļūdas paziņojumu parādīšanu lietotājam, kļūdas esamības paziņošanu izstrādātājam, precīzu un saprotamu kļūdas paziņojumu sniegšanu izstrādātājam un beidzot ar efektīvu kļūdas novēršanu. Arī šie drošības pasākumi nepieciešami programmatūras drošībai.

1.1.1.4. Pieejas kontrole

Informācijas pieejas kontrolei jābūt dokumentētai un regulāri kontrolētai atbilstoši biznesa un drošības pieejas prasībām (17, 60. lpp). Šī sadaļa apraksta

nepieciešamās lietotāju autorizācijas kontroles, apskatot drošu lietotāju reģistrācijas procedūru. Procedūra ietver lietotāja pieejas līmeņa definēšanu, atbilstošu tiesību piešķiršanu, lietotāja informēšanu par pieejas tiesībām, kā arī visu lietotāju pieejas tiesību saraksta regulāru pārskatīšanu, neaktīvo lietotāju bloķēšanu vai dzēšanu, lietotāju datu drošības nodrošināšanu.

Lietotāja paroles drošības kontrole ietver lietotāja informēšanu par paroles neatklāšanu citām personām un drošas paroles nepieciešamību. Lietotāja paroles sistēmā nedrīkst glabāt atklātā veidā. Ieteicams sistēmā iestrādāt paroles derīguma intervālu, jeb regulāru paroles maiņas politiku (17). Darba autore uzskata, ka pieejas kontrole, tātad arī droša parole, ir pamats programmatūras drošībai.

1.1.1.5. Informāciju sistēmas pieņemšana, izstrāde un uzturēšana

Šīs drošības kontroles sadaļas mērķis ir nodrošināt to, ka drošība ir kā integrēta informācijas sistēmas daļa. Nepieciešams analizēt drošības prasības, tās specificēt un dokumentēt (17, 77. lpp).

Informācijas apstrādes laikā jāņem vērā cilvēciskais faktors, tādēļ programmatūrai jāveic ievaddatu validācija, kas novērsīs iespējamās cilvēciskās kļūdas. Lai izstrādātu efektīvāku un pēc iespējas drošāku datu validāciju, izstrādātājam ieteicams standartizēt datu pievienošanas, labošanas un dzēšanas funkcijas, kas izstrādes procesā iegūtu lielāku kvalitāti (17, 78. lpp).

Šifrēšanas metožu pielietošanai vajadzētu izstrādāt plānu, kādos gadījumos, kādiem datiem šifrēšana tiek veikta (17, 80. lpp).

1.1.1.6. Uzņēmējdarbības nepārtrauktības pārvaldība

Šīs sadaļas mērķis ir nodrošināt pēc iespējas lielāku procesu nepārtrauktības līmeni un mazināt sekas, ja noticis pārtraukums svarīgā uzņēmējdarbības procesā. Mērķi palīdzēs sasniegt uzņēmējdarbības procesu nepārtrauktības pārvaldības plāns, kurā ieteicams analizēt iespējamos riskus un izstrādāt šo risku atbildes, plānam jābūt dokumentētam (17, 90. lpp). Tā kā „Ārsta Birojs 3” ir veselības aprūpes programmatūra, nepieciešams nodrošināt nepārtrauktu tās pieejamību, tādēļ tiks apskatīti arī programmatūras nepārtrauktības nodrošināšanas risinājumi.

1.1.2. ISF

Informācijas drošības forums (ISF) ir izstrādājis Informācijas drošības labās prakses standartu (*The Standard of Good Practice for Information Security*). Standarts ir izstrādāts no uzņēmējdarbības perspektīvas, sniedzot pamatinformāciju par organizācijas informācijas drošības pasākumu novērtēšanu (39, 1. lpp). Standarts ir sadalīts 6 galvenajās daļās:

- Drošības pārvaldība (uzņēmuma mēroga),
- Datoru instalācijas,
- Tīkli,
- Kritiskās biznesa lietotnes,
- Sistēmas izstrāde,
- Gala lietotāja vide.

Katra no šīm daļām sniedz detalizētu un konkrētu rīcību ieteikumu kopu, kopumā sniedzot 166 norādījumus. Izceļot ar programmatūras drošību saistītās kontroles, tālāk sekojošās apakšnodaļās detalizētāki apskatītas daļas: drošības pārvaldība, kritiskās uzņēmējdarbības lietotnes, sistēmas izstrāde un gala lietotāja vide.

1.1.2.1. Drošības pārvaldība

Drošības pārvaldības nodaļa apraksta vadlīnijas, kā nodrošināt efektīvu informācijas drošības pārvaldību. Pirmkārt, informācijas drošību var nodrošināt tikai tad, ja kāds ir apņēmies to darīt, tas ietver drošības pārvaldības metodes izvēli, profesionālu atbildīgo personu iecelšanu, drošības pārvaldības politikas izstrādi (39). Arī šis standarts iesaka veikt informācijas drošības risku analīzi, kas palīdzēs noteikt iespējamos apdraudējumus. Būtu ieteicams izvēlēties piemērotāko risku analīzes metodoloģiju, kura tiek dokumentēta, regulāri pārskatīta un apstiprināta organizācijā.

Informācijai jābūt klasificētai, skaidri definējot tās lomu un izmantojamību organizācijas darbībā, īpaši pievēršot uzmanību informācijas konfidencialitātes, integritātes un pieejamības prasībām (39).

Informācijas privātums ir ļoti svarīgs drošības aspekts, tādēļ jābūt dokumentētām procedūrām, kuras nodrošina informācijas privātumu. Nepieciešams

definēt personu tiesības, par kurām informācija tiek uzturēta, nodrošināt informācijas pieejamību tikai personām, kurām ir tiesības šo informāciju apstrādāt.

Drošības politikai jāapskata arī lietotāja identifikācijas un piekļuves procesa drošību, kas ietver lietotāju kontu administrēšanas drošību. Lietotāja identifikācijas un piekļuves mehānisms parasti ietver (39):

- Lietotāja reģistrācijas procesu, kurā lietotājam tiek piešķirts piekļuves konts un sistēmas lietošanas tiesības,
- Lietotāja piekļuves process, kurš tiek izpildīts katru reizi, kad lietotājs cenšas piekļūt sistēmai.

Lai nodrošinātu biznesa nepārtrauktību, jābūt dokumentētiem nepārtrauktības nodrošināšanas plāniem. Plāns ietver kritisko punktu identifikāciju, pamatojoties uz risku analīzes rezultātiem. Rīcības plāniem jātiek regulāri atjaunotiem. Nepieciešams parūpēties par rezerves kopiju veidošanu, glabāšanu atdalītā vietā (39).

Nepieciešams nodrošināties pret ļaunprātīgiem uzbrukumiem, tajā skaitā vīrusiem, Trojas zirgiem un ļaunprātīgiem kodiem. Tas ietver ne tikai pretvīrusu programmatūru uzstādīšanu, bet arī lietotāja drošības ievērošanu, piemēram, nebūtu vēlams atvērt aizdomīgus e-pasta pielikumus, uzstādīt programmatūras, kuras izstrādājušas nezināmas organizācijas, atvērt tīmekļa saites no nezināmiem avotiem.

1.1.2.2. Kritiskas uzņēmējdarbības lietotnes

Kritiskās uzņēmējdarbības lietotnes pieprasa stingrāku drošības uzraudzību. Lietotnes nozīmīguma līmeni palīdz noteikt uzņēmējdarbības ietekmes analīze, gadījumā ja notiek informācijas konfidencialitātes, integritātes vai pieejamības zaudējumi (39).

Lietotnes pārvaldības process ietver nepieciešamo drošības pasākumu definīcijas. Ja lietotnes procesos tiek veiktas izmaiņas, nepieciešama izmaiņu kontrole, veicot arī testēšanu. Šeit jāņem vērā arī versiju kontroles nepieciešamību, kas nodrošinās iespēju atgriezties pie iepriekšējām lietotnes versijām.

Sensitīvu datu apstrādei jābūt drošai, ņemot vērā arī datu fizisko atrašanās vietu. Informācijas pārraides procesam jābūt drošam, informācijai jābūt pieejamai tikai autorizējoties (39). Arī veselības aprūpes programmatūra apstrādā sensitīvus datus, tātad jāņem vērā arī šo datu apstrādes drošība.

Pieejas kontrole tiek veikta, izmantojot autorizācijas vai autentifikācijas procesu. Arī lietotājam ir tiesības zināt par to, kādi drošības pasākumi tiek ievēroti, tādēļ pēc veiksmīgas piekļuves veikšanas, vēlams sniegt lietotājam informāciju par to, kā tiek aizsargāta informācija, sniegt informāciju par lietotāja tiesībām un pienākumiem (lietotājam jābūt informētam par veicamajiem drošības pasākumiem, lietojot sistēmu), kā arī informāciju par datumu un laiku, kad lietotājs pēdējo reizi ir veiksmīgi piekļuvis sistēmai. Gan veiksmīgas gan neveiksmīgas piekļuves gadījumiem jātiek reģistrētiem, tā nodrošinot drošībai nepieciešamos auditācijas pierakstus. Vēlams bloķēt piekļuvi lietotāja vārdam līdz ar trešo neveiksmīgo pieslēgšanās mēģinājumu. Lietotāja paroles nevar tikt glabātas atklāta teksta veidā (39).

1.1.2.3. Sistēmas izstrāde

Veicot drošības kontroli sistēmas izstrādes laikā ir rentablāk kā veicot to pēc sistēmas izstrādes, tātad drošībai jātiek kontrolētai visā sistēmas izstrādes dzīves ciklā (39). Lai efektīvi izstrādātu sistēmu, nepieciešams vadīties pēc izvēlētas izstrādes metodoloģijas. Izvēlētajai metodoloģijai jābūt dokumentētai, regulāri pārskatītai un atjaunotai, kā arī jāveic regulārs pārskats par izstrādes procesa atbilstību metodoloģijai (39).

Ļoti būtiski ir specificēt un dokumentēt sistēmas prasības, kuras iekļautu arī drošības prasības. Šādā veidā tiek nodrošināta klienta un izstrādātāja sadarbība, drošība, ka sistēma tiks izstrādāta atbilstoši klienta vajadzībām (39). Pēc darba autores domām, drošības prasību specificēšana un dokumentēšana ir obligāti nepieciešama, lai parūpētos par to, ka klientam tiek nodota droša programmatūra.

Testēšanas process ir viennozīmīgi nepieciešams un var tikt uzskatīts par obligātu drošības kontroles procesa daļu. Veicot testēšanu, jātestē visus sistēmas elementus – sistēmas lietotnes, aparatūras, servisus; lai tiktu nodrošināta sistēmas funkcionalitāte un drošības prasību atbilstība (39).

1.1.2.4. Gala lietotāja vide

Kā jau iepriekš tika minēts, lietotājam ir jāapzinās drošības pasākumu ievērošanas nepieciešamība. To palīdz nodrošināt lietotāja apmācības, kurās

jāpaskaidro, kā pareizi lietot sistēmu, kādus pasākumus veikt, lai nodrošinātu informācijas konfidencialitāti un integritāti (39).

Datubāzei jābūt aizsargātai pret neautorizētu piekļuvi. Datiem jābūt validētiem, nevajadzētu pieļaut noklusētās vērtības (39).

1.1.3. ITIL

Informācijas tehnoloģiju infrastruktūras bibliotēka (ITIL) ir Informācijas tehnoloģiju pakalpojumu pārvaldības un izstrādes procedūru un jēdzienu kopums. ITIL sniedz visaptverošu procedūru un kontrolsarakstu kopumu, kuru jebkura organizācija var pielāgot savām vajadzībām. ITIL ir publicētas vairākas grāmatas, kuras aptver dažādas IT pārvaldības daļas (27). ITIL otrajā versijā ir publicētas 8 grāmatas, tostarp arī Drošības pārvaldība.

Drošības pārvaldība ir balstīta uz pasaulē vispārpieņemto labas prakses standartu ISO 27002, kas tika apskatīts 1.1.1. nodaļā. Drošības pārvaldības pamatā ir informācijas drošība, kur tiek apskatīta informācijas konfidencialitāte, integritāte un pieejamība. Drošības pārvaldības mērķis ir sadalīts divās daļās (27):

1) Drošības prasību realizācija, kas tiek definēta servisa līmeņa līgumos un vienošanās dokumentos,
2) Pamata drošības elementu realizācija.

Drošības pārvaldības process tālāk tiek iedalīts sekojošās daļās: kontrolē, plānošanā, ieviešanā, novērtēšanā un uzturēšanā. Procesa shēmu skatīt 3. pielikumā.

Plānošanas process ietver aktivitātes, kas veicamas, lai nodrošinātu līgumā noteiktos drošības pasākumus (27). Procesa rezultātā tiek sastādīts drošības pārvaldības plāns. Kā jau iepriekš tika minēts, arī programmatūras drošībai, pēc autores domām, nepieciešama drošības plānošana.

Ieviešanas process nodrošina to, ka tiek realizētas visas prasības un veicamās procedūras, kas definētas drošības pārvaldības plānā. Šis process ietver sekojošas aktivitātes: lietotņu klasifikācija un pārvaldība, personāla drošības ieviešana, lai novērstu krāpšanas un zādzību iespējamību, konkrētu drošības prasību definēšana un dokumentēšana, piekļuves mehānismu definēšana un dokumentēšana (27).

1.1.4. Programmatūras drošības nodrošināšana

Programmatūras drošības nodrošināšana (*Software Security Assurance*) ir process, kas nodrošina programmatūras izstrādi atbilstoši drošības līmenim, kas aizsargā pret potenciāliem apdraudējumiem, kas varētu rasties datu zaudēšanas, nekorektuma, nepieejamības vai ļaunprātīgas izmantošanas gadījumā (31, 1. lpp). Šo procesu ir izstrādājis Informācijas nodrošināšanas tehnoloģiju analīzes centrs (*Information Assurance Technology Analysis Center*).

Šis dokuments sniedz drošas programmatūras definīciju: *droša programmatūra ir programmatūra, kas ir spējīga pretoties vairumam uzbrukumu, sadarboties ar uzbrukumiem, kuriem tā nevar pretoties, un ātri atveseļoties ar minimāla postījuma sekām no tiem dažiem uzbrukumiem, ar kuriem tā nevar sadarboties* (31, 2. lpp).

Droša programmatūra ir izstrādāta, ieviesta, konfigurēta un uzturēta, lai panāktu, ka (31, 22. lpp):

- tā turpina pareizi darboties vairumā uzbrukumu gadījumu vai nu aizliedzot uzbrukumam piekļuvi, vai apstrādā kļūdas, kas radušās no šiem uzbrukumiem;
- izolē, aptur un samazina ietekmi, kas radusies no uzbrukuma, kuru programmatūra nav spējusi aizliegt vai apstrādāt, un pēc iespējas ātrāk atjauno darbību.

Kā minimumu Programmatūras drošības nodrošināšanas procesam vajadzētu parūpēties par to, ka (31):

- ir veikts programmatūras drošības novērtējums,
- ir izstrādātas programmatūras drošības prasības,
- ir izstrādātas programmatūras izstrādes un uzturēšanas procesa drošības prasības,
- katrs programmatūras audits ietver programmatūras drošības prasību novērtējumu,
- eksistē konfigurācijas un labojumu pārvaldības plāns,
- fiziskā drošība tiek adekvāti ievērota.

1.1.5. CobIT

CobIT (*Control Objectives for Information and related Technology*) ir pasaulē atzīta un plaši lietota rokasgrāmata, kurā apkopota praktiska pieredze informācijas tehnoloģiju pārvaldīšanai. Tā parāda savstarpējo saistību starp biznesa riskiem,

tehniskiem jautājumiem un kontroles nepieciešamību. CobIT ir IT pārvaldības procesu novērtēšanas un pilnveidošanas metode, kurai ir biznesa orientācija (18).

CobIT efektīva izmantošana uzņēmumā liecina par augstu uzņēmuma kultūras līmeni, tā ir pieeja, kura liek mainīties uzņēmuma vadībai, jo šī metode ir uzņēmuma vadības pārvaldības rīks un, tikai mainoties un domājot CobIT valodā, uzņēmums var panākt atbilstību labākajai praksei (18).

CobIT ir definējis informācijas kritērijus, pēc kuriem vadoties var noteikt informācijas derīgumu (24, 10. lpp):

- Lietderīgums (*effectiveness*) – informācija ir atbilstoša biznesa procesam, tā tiek piegādāta laikā un izmantojamā veidā,
- Efektivitāte (*efficiency*) – informācijas piegādāšanai resursi tiek optimāli izmantoti,
- Konfidencialitāte (*confidenciality*) – jūtīga informācija ir aizsargāta pret neatļautu izpaušanu,
- Integritāte (*integrity*) – informācija ir pareiza un precīza,
- Pieejamība (*availability*) – informācija ir pieejama, kad tā ir vajadzīga biznesa procesam,
- Atbilstība (*compliance*) – informācija ir saskaņā ar likumiem un līgumu prasībām,
- Uzticamība (*reliability of information*) – vadība var uzticēties informācijai, kā arī finansu pārskatiem.

Lai nodrošinātu informāciju, kas uzņēmumam nepieciešama, tā mērķu sasniegšanai, CobIT piedāvā sagrupēt IT procesus loģiskai IT resursu pārvaldībai. CobIT izšķir 4 grupās sagrupētus 34 procesus, kas tālāk satur definētus kontroles mērķus (18), tiek pieminēti ar programmatūras drošību saistītie procesi:

1) Plānošana un organizācija (**PO** – *Planning & Organization*)

- **PO2** Informācijas arhitektūras definēšana - informācijas arhitektūras modelis, datu klasifikācijas shēma un drošības līmenis. Lai uzzinātu, kādu informāciju apstrādā programmatūra, nepieciešama tās klasifikācija;
- **PO9** Risku noteikšana un pārvaldība - biznesa risku apzināšana, risku noteikšanas pieeja, risku identificēšana, risku vērtēšana, risku vadības plāns, risku akceptēšana, kontroļu izvēle. Nepieciešams identificēt un pārvaldīt iespējamos programmatūras apdraudējumus;

2) Iegāde un ieviešana (**AI** – *Acquisition & Implementation*)

- **AI2** Lietojumprogrammatūras iegāde un uzturēšana - programmu specifikācija, ievada un izejas prasību definēšana un dokumentēšana, saskarņu definēšana, kontrolējamība, sistēmas pieejamības nodrošināšana, IT integritātes nodrošināšana programmatūras programmu aplikācijās, programmatūras aplikāciju testēšana, lietotāju instrukcijas un atbalsta materiāli. Programmatūras drošība lielā mērā ir atkarīga no definētajām drošības prasībām, tādēļ visas prasības ir jāspecificē;

3) Piegāde un atbalsts (**DS** – *Delivery & Support*)

- **DS3** Veiktspējas pārvaldība - pieejamības nodrošināšanas plāni, veiktspējas pārraudzība, veiktspējas modelēšanas un prognozēšanas rīki, resursu pieejamības un kapacitātes nodrošināšanas plāni. Šīs kontroles attiecas arī uz programmatūras pieejamības nodrošināšanu;
- **DS5** Sistēmu drošības nodrošināšana - drošības pasākumi, konfidencialitātes un slepenības prasības, autorizācija, autentifikācija un pieejas kontrole, tiešas pieejas datu drošība, lietotāju identifikācija un autorizācijas profili, tiesību piešķiršana pēc vajadzības, tiesību piešķiršanas kontrole, kriptogrāfisko atslēgu vadība, drošības incidentu apstrāde, atskaites un izpildes sekošana, rīki politikas ievērošanas testēšanai un novērtēšanai, incidentu ievērošanai un automātisku atskaišu sagatavošanai;
- **DS7** Lietotāju izglītošana un apmācība - apmācības nepieciešamības identificēšana, apmācības organizēšana, drošības principu apmācība. Lietotājs var ne tikai programmatūru apdraudēt, bet arī aizsargāt;
- **DS11** Datu pārvaldība - datu sagatavošanas procedūras, datu izvades autorizācijas procedūras, precizitātes, pareizības un autorizācijas pārbaude, datu ievades kļūdu apstrāde, datu apstrādes integritāte, datu ievades labošana un apstiprināšana, datu apstrādes kļūdu labošana, izvades datu saglabāšana, izvades pareizības saskaņošana, izvades kļūdu apstrāde, sensitīvu datu aizsardzība pārraides un transportēšanas laikā, izdzēstu sensitīvu datu aizsardzība, datu uzglabāšanas vadība, uzglabāšanas periods un glabāšanas nosacījumi, rezerves kopēšana un datu atjaunošana, rezerves kopēšanas stratēģija, arhivēšana, autentifikācija un integritāte, datu integritātes nepārtrauktība.

1.1.6. LR normatīvie akti

Latvijas republikas normatīvajos aktos darba autore neatrod atbilstošus likumus vai ieteikumus veselības aprūpes programmatūras drošībai, tomēr eksistē Fizisko personu datu aizsardzības likums, kur 10.pants (1) norāda uz personas datu apstrādes ierobežojumiem (19): *„Lai aizsargātu datu subjekta intereses, pārzinis nodrošina:*

1) *godprātīgu un likumīgu personas datu apstrādi;*
2) *personas datu apstrādi tikai atbilstoši paredzētajam mērķim un tam nepieciešamajā apjomā;*
3) *tādu personas datu glabāšanas veidu, kas datu subjektu ļauj identificēt attiecīgā laikposmā, kurš nepārsniedz paredzētajam datu apstrādes mērķim noteikto laikposmu;*
4) *personas datu pareizību un to savlaicīgu atjaunošanu, labošanu vai dzēšanu, ja personas dati ir nepilnīgi vai neprecīzi saskaņā ar personas datu apstrādes mērķi"*.

Šajā likumā tiek dotas norādes par sistēmas drošības pārbaudes nepieciešamību - 26.panta 2.punktā teikts (19): *„Valsts un pašvaldību institūcijas reizi divos gados iesniedz Datu valsts inspekcijai audita atzinumu par personas datu apstrādi, ietverot tajā arī riska analīzi, un pārskatu par informācijas drošības jomā veiktajiem pasākumiem. Prasības audita atzinumam nosaka Ministru kabinets"*.

Personas datu aizsardzības obligātās tehniskās un organizatoriskās prasības 3. punktā tiek minēts (23): *„Personas datu obligāto tehnisko aizsardzību īsteno ar fiziskiem un loģiskiem aizsardzības līdzekļiem, nodrošinot:*

3.1. aizsardzību pret fiziskās iedarbības radītu personas datu apdraudējumu;

3.2. aizsardzību, kuru realizē ar programmatūras līdzekļiem, parolēm, šifrēšanu, kriptēšanu un citiem loģiskās aizsardzības līdzekļiem".

Šī likuma 5.punktā minēts (23): *„Pārzinis, apstrādājot personas datus, izstrādā iekšējos datu apstrādes aizsardzības noteikumus, kuros nosaka: [..]*

5.2. personas datu aizsardzības klasifikāciju atbilstoši to vērtības un konfidencialitātes pakāpei; [..]

5.8. paroles garumu un uzbūves nosacījumus (minimālais paroles garums ir astoņi simboli);

5.9. paroles lietošanas kārtību, kā arī laikposmu, pēc kura nomaināma parole;

5.10. rīcību, ja parole vai kriptoatslēga kļuvusi zināma citai personai;

5.11. personas datu lietotāju tiesības, pienākumus, ierobežojumus un atbildību”.

1.2. Kopsavilkums

Pirmajā nodaļā tika apskatīts programmatūras drošības jēdziens, kā arī informācijas drošības un drošības pārvaldības pasaulē atzītie standarti. Visos apskatītajos standartos informācijas drošības definīcija iekļauj informācijas konfidencialitātes, integritātes un pieejamības pārvaldības nepieciešamību. Šīs trīs pamata īpašības tiks ņemtas vērā, pētot izvēlētās programmatūras „Ārsta Birojs 3” drošību.

Apkopojot standartos minēto informāciju, darba autore secina, ka programmatūras drošības pārvaldības procesam ir jābūt plānotam, dokumentētam un tas ir veicams visā sistēmas izstrādes dzīves ciklā. Lai detalizētāki apkopotu iegūto informāciju par drošības pārvaldību, darba autore piedāvā apkopojošu tabulu, kurā tiek attēlota apskatītā standarta vai ieteikuma iekļauto drošības kontroļu izmantošana (skatīt 1.2.1. tabulu).

Autore secina, ka visplašāk programmatūras drošību apraksta starptautiski atzītais standarts ISO/IEC 17799:2005 un kontroles ieteikumi CobIT. Vairākumā apskatīto dokumentu ir iekļauts apraksts par piekļuves drošības kontrolēm – autorizāciju un drošu paroli, kas liecina par šīs kontroles esamības obligātumu programmatūras drošības nodrošināšanai. Lielākajā daļā dokumentu ieteikts veikt drošības plānošanu, risku analīzi un personāla drošības apmācības. Pēc darba autores domām, šie četri kritēriji ir drošības kontroles pamatā, bet arī pārējie kritēriji kopumā nosaka programmatūras drošības līmeni, un visas drošības kontroles tiks vērtētas programmatūras „Ārsta Birojs 3” pārskatā.

1.2.1. tabula

Standartos iekļauto drošības risinājumu apkopojums

Drošības kontrole / Dokuments	Drošības pārvaldība		Konfidencialitāte		Integritāte			Pieejamība	Izstrādes vides drošība		
	Drošības plāns	Risku analīze	Autori-zācija, droša parole	Lomu un tiesību sadalījums	Datu validācija	Auditācijas pieraksti	Personāla drošības apmācība	Darījumprocesu nepārtrauktības nodrošināšana	Testēšana	Meto-doloģija	Koda inspekcija
CEN/ prENV 12251			✓								
ASTM/ E1896-97			✓								
ASTM/ E1988:98							✓				
ISO/IEC 17799: 2005	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
ISF	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓
ITIL	✓		✓				✓		✓		✓
SSA	✓	✓						✓	✓	✓	
CobIT	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
LR normatīvie akti		✓	✓				✓	✓			

✓ - ieteikumos iekļauta drošības kontrole

2. DROŠĪBAS PĀRSKATI

Tā kā viens no darba uzdevumiem ir programmatūras drošības novērtēšana, nepieciešams veikt programmatūras drošības pārskatu. Lai efektīvāk un kvalitatīvāk veiktu drošības pārskatu, darba autore veic esošo, starptautiski atzīto programmatūras un informācijas sistēmu drošības auditu un novērtējumu ieteikumu apskati.

2.1. IEEE programmatūras apskašu standarts

IEEE programmatūras apskašu standarts (*IEEE Standard for Software Reviews*) ir apstiprināts 1997. gada 9. decembrī. Standarts definē apskates mērķi: *programmatūras apskates mērķis ir sniegt neatkarīgu novērtējumu programmatūras un tās procesu atbilstībai likumiem, standartiem, labajai praksei, plāniem un procedūrām* (16, 25. lpp).

Standarts pieņem, ka programmatūras apskati jeb auditu veic auditoru grupa, tādēļ tiek definētas audita procesa atbildības un sekojošas lomas: galvenais auditors, reģistrētājs, auditori, iniciators, auditētā organizācija. Galvenā auditora pienākumos ietilpst audita plāna sagatavošana, auditoru grupas sastādīšana, audita atskaites sagatavošana (16, 26. lpp).

Apskates pamatā vajadzētu būt plānam, kurš tiek izstrādāts, balstoties uz auditoru un klienta sanāksmes spriedumiem. Apskates process tiek veikts atbilstoši plānā pieņemtajiem standartiem. Par apskates rezultātiem ieteicams veidot sapulci, kurā pārrunāt rezultātus, problēmas, ar kurām nācās saskarties apskates procesā, ieteikumus drošības uzlabošanai un vispārējos apskates rezultātus (16, 30. lpp).

2.2. ISACA ieteikumi

Starptautiskā organizācija ISACA ir dibināta 1967. gadā un šobrīd tā apvieno vairāk kā 65 000 profesionāļus informācijas tehnoloģiju pārvaldībā, drošībā un auditā no vairāk kā 140 valstīm. ISACA mērķis ir kļūt par pasaulē atzītu līderi IT vadībā, kontrolē un pārbaudē. ISACA ir starptautiski atzīta sertifikācijas programma IS audita, kontroles un drošības jomā, ir atzinību guvušas profesionālas publikācijas, kas sniedz informāciju par jaunākajiem pētījumiem, konkrētu situāciju analīzi un praktiskus padomus (28).

ISACA piedāvā arī 42 vadlīnijas audita veikšanai (29), divas no tām var palīdzēt programmatūras pārskata procesā:

- **G38** Pieejas kontrole – auditoram jāspēj novērtēt pieejas kontroles risinājumus;
- **G42** Nepārtrauktības nodrošināšana – vadlīnijas paredzētas, lai veiksmīgi novērtētu nepārtrauktības nodrošināšanu plānošanas, ieviešanas un uzturēšanas procesā.

ISACA apskata vairākus rīkus un tehnikas (30), kas palīdz audita veikšanas procesā. Uz programmatūras drošību vairāk attiecas ieteikumi risku un drošības novērtēšanai:

- **P1** IS risku novērtējuma mērījumi – dokuments ietver IS audita risku novērtējuma definīciju, norādījumus, kā izmantot risku novērtēšanas metodiku iekšējā audita veikšanā, un vadlīnijas par riska pakāpes kritēriju izvēli. Tiek paskaidrota risku bāzēta audita pieeja un risku novērtēšanas tehnikas, tiek sniegti vairāki konkrēti risku novērtēšanas procesu piemēri;
- **P8** Drošības novērtējums – caurlaidības testi un ievainojamības analīze – procedūra paskaidro ieteicamos soļus, kas veicami testēšanas procesā.

ISACA Sertificēto informācijas sistēmu auditoru (*Certified Information Systems Auditor*) programma ir visā pasaulē atzīts standarts, saskaņā ar kuru tiek novērtēti sasniegumi IS audita, kontroles un drošības jomā (22). Lai kvalitatīvāk veiktu programmatūras pārskatu, darba autore iepazīstas ar CISA ieteikumiem, kas attiecas uz programmatūras drošības pārskatu.

CISA rokasgrāmata ietver informāciju par IS audita metodēm, drošības un audita standartiem, risku analīzes metodēm, kā arī ir kā mācību materiāls, jo piedāvā dažādus uzdevumus katras nodaļas beigās (11).

2.3. Personas datu apstrādes sistēmu audita rokasgrāmata

Šo rokasgrāmatu izstrādājusi Datu valsts inspekcija un tā var tikt uzskatīta par standartu personas datu apstrādes sistēmu audita veikšanai. Šī rokasgrāmata apskata audita veikšanu, lai piemērotu Fizisko personu datu aizsardzības likuma prasības. Šī rokasgrāmata paredzēta ne tikai Datu valsts inspekcijas informāciju sistēmu audita veikšanai, bet arī jebkuras citas personas datu apstrādes informācijas sistēmas audita veikšanai. Rokasgrāmata definē konkrētus audita mērķus, kurus būtu jāsasniedz (14):

- Jāpārbauda, vai auditējamajā sfērā eksistē oficiāla (t.i., dokumentēta un atbilstoši precizēta) personas datu aizsardzības sistēma;
- Jāpārbauda, vai šajā sfērā strādājošais ar personas datu aizsardzību saistītais personāls: zina, ka eksistē personas datu aizsardzības sistēma, izprot personas datu aizsardzības sistēmu, izmanto personas datu aizsardzības mehānismu;

- Jāpārbauda, vai šajās sfērā personas datu aizsardzības mehānismi reāli funkcionē un ir efektīvi.

Arī šajā rokasgrāmatā tiek uzsvērts, ka audits ir regulāri veicams un to veic profesionāls speciālists, kurš ir no organizācijas neatkarīga persona. Dokumentā tiek norādīts, ka eksistē sekojoši audita veidi: iekšējais un ārējais audits, kā arī atbilstības audits un saskaņas audits.

Atbilstības audita mērķis ir pārliecināties par to, ka pastāv dokumentēts datu aizsardzības mehānisms, kas pilnībā atbilst visām Fizisko personu datu aizsardzības likuma prasībām (14, 13. lpp). Ja atbilstības audita rezultāti liecina, ka organizācijai ir dokumentēts datu aizsardzības mehānisms un tajā pastāv tikai dažas nepilnības vai trūkumi, auditors uzsāk saskaņas audita procesu.

Saskaņas audita mērķis ir noskaidrot, kādā veidā datu aizsardzības mehānisms tiek izmantots un cik efektīvs tas ir. Saskaņas auditam eksistē divas pamata metodoloģijas (14, 16.lpp):

- Funkcionālais jeb vertikālais audits – audita laikā tiek pārbaudīti visi personas datu aizsardzības aspekti kādā noteiktā darbības sfērā, funkcijā vai organizatoriskā struktūras daļā. Funkcionālā audita laikā tiek ieteikts veikt personāla intervijas;
- Procesa jeb horizontālais audits – audits tiek veikts ar mērķi izpētīt kāda noteikta procesa norisi no sākuma līdz procesa beigām. Audita laikā tiek pētīta vairāku sfēru, funkciju vai struktūrvienību mijiedarbība šī procesa ietvaros.

Rokasgrāmata piedāvā audita procesa definīciju, kura ietver 5 galvenos etapus, kas veicami secīgi (14, 24. lpp):

1) Audita plānošana, kas sevī ietver risku novērtējumu, audita grafika sastādīšanu, apstiprināšanu un publiskošanu, audita grafika ievērošana, auditora izvēli (iekšējais vai ārējais auditors), pirms audita anketas izdalīšanu, sagatavošanās sanāksmes noturēšanu, audita vadības veidlapas uzsākšana,
2) audita sagatavošana – atbilstības audita veikšana, audita grafika apstiprināšana, veidlapu sagatavošana, audita plāna sastādīšana,
3) saskaņas audita veikšana – atklāšanas sanāksmes noturēšana, vides sagatavošana, audita veikšana,
4) saskaņas audita pārskatu sastādīšana – Nesaskaņas protokola sastādīšana, saskaņas audita pārskata ziņojuma sastādīšana, noslēguma sanāksmes noturēšana, pārskata ziņojumu izplatīšana,
5) audita sekojums – tiek veikts audita sekojums ar mērķi novērst nesaskaņas, tiek parakstīti audita protokoli, audits tiek noslēgts.

Kā jau tika minēts, audita laikā ieteicams veikt personāla intervijas, šādā veidā noskaidrojot esošo situāciju no personāla skata punkta. Intervijas laikā auditors iepazīstina ar sevi un audita mērķiem, ļauj izteikties intervējamai personai, pēc tam ar jautājumu palīdzību, tiek vākta nepieciešamā informācija. Pēc intervijas būtiski ir veikt pierādījumu novērtējumu, nosakot avotu uzticamības līmeni (14, 49. lpp).

Veicot šīs rokasgrāmatas pārskatu, darba autore secina, ka būtiska loma pārskata veikšanā ir pārskata dokumentēšanai, tomēr ierobežotā laika dēļ šī darba izstrādes ietvaros netiks veikta pārskata dokumentēšana. Tomēr ieteikums par intervijas izmantošanu, lai iegūtu informāciju par programmatūras drošības risinājumiem, tiks ņemts vērā.

2.4. OSSTMM rokasgrāmata

OSSTMM ir atvērtā koda drošības testēšanas metodoloģijas rokasgrāmata, ko piedāvā ISECOM (*Institute for Security and Open Methodologies*). Pirmā rokasgrāmatas versija tika publicēta 2001. gadā, tā sastāvēja no 14 lapaspusēm, šobrīd aktuālā versija 3 RC 28 ir vairāk kā 150 lapaspuses. Rokasgrāmata ietver vadlīnijas drošības testēšanai piecās daļās, viena no tām ir Informācijas drošība, kuras mērķis ir noteikt interneta biznesa aptvērumu, noteikt datu vākšanas sistēmas un tehnikas, kā arī definēt savākto datu klasifikāciju (32).

Rokasgrāmata piedāvā arī veidnes audita dokumentēšanai. Rokasgrāmata apskata pilnu drošības testēšanas ciklu, ietverot tīkla drošības izvērtēšanu un fizisko drošību, tomēr šī rokasgrāmata neapskata detalizētas programmatūras drošības testēšanas vadlīnijas. Pielikumā tiek ieteiktas personas datu drošības vadlīnijas, uzsverot, ka testēšanas jeb audita procesā izmantotajiem datiem jābūt aizsargātiem (32, 121. lpp):

- Personas datus drīkst izmantot tikai ar personas piekrišanu, izņemot gadījumus, kad personas datus ļauj izmantot valsts likums,
- Testēšanā izmantotos datus vajag atbilstoši iznīcināt pēc testēšanas pabeigšanas,
- Dati nevar tikt atklāti citām personām,
- Arī testēšanas procesā jārūpējas par drošību, veicot visus nepieciešanos drošības pasākumus.

2.5. Kopsavilkums

Šajā nodaļā tika apskatīti ieteikumi, kas palīdz veikt informācijas sistēmu auditus. Eksistē dažādi standarti, katram no tiem tiek ir izvirzīts kāds konkrēts mērķis, tomēr

vienojošais elements ir informācijas, programmatūras, tīkla un fiziskā drošība. Apskatot populārākos un pasaulē pieņemtākos standartus darba autore secina, ka audita process visos standartos ir līdzīgs, kas liecina par to, ka ir izstrādāts kvalitatīvs un pasaulē atzīts procesa standarts.

Ieteikumos iekļautie programmatūras pārskatā veicamie soļi ir apkopoti tabulā (skatīt 2.5.1. tabulu).

2.5.1. tabula

Ieteikumos iekļauto programmatūras pārskata soļu apkopojums

Solis / Dokuments	**Plānošana**	**Sagatavošana, dokumentēšana**	**Intervijas**	**Risku novērtējums**	**Drošības novērtēšana**	**Atskaites sapulce**	**Atskaite un ieteikumi**
IEEE	✓	✓	✓	✓	✓	✓	✓
ISACA	✓	✓	✓	✓	✓	✓	✓
PDASAR*	✓	✓	✓		✓	✓	✓
OSSTMM	✓				✓		✓

* - Personas datu apstrādes sistēmu audita rokasgrāmata

Ņemot vērā šajā nodaļā apskatītos ieteikumus pārskata veikšanai, darba autore programmatūras pārskatā izmanto interviju, lai iegūtu informāciju par programmatūras drošības risinājumiem, kā arī veic risku novērtējumu, lai identificētu iespējamos apdraudējumus. Datu aizsardzības novērtējums, pēc autores domām, ir obligāti iekļaujams programmatūras novērtējumā, tādēļ tiks apskatīti informācijas drošības kritēriji – konfidencialitāte, integritāte un pieejamība.

Visos apskatītajos dokumentos tiek ieteikts veikt pārskata plānošanu un pārskata atskaites izstrādi. Darba autores veicamajā programmatūras pārskatā par plānošanas daļu var uzskatīt šajā nodaļā veikto ieteikumu apskati, apkopošanu un veicamā procesa definēšanu. Par pārskata atskaiti var uzskatīt šī darba 4. nodaļu, kurā aprakstīti pārskata rezultāti.

CISA rokasgrāmatā minēts, ka, izmantojot riska novērtēšanas metodi, iespējams identificēt svarīgākās lietas, kuras vajadzētu auditēt, kas palīdzētu efektīvi noteikt ierobežotos audita resursus, nodrošinātu visu līmeņu auditu un sniedz apkopojumu par to, kā audita subjekts ir saistīts ar sistēmu un tās procesiem (11, 38. lpp). Ņemot vērā šos ieteikumus, darba autore uzskata, ka programmatūras drošības vērtēšanai nepieciešams identificēt pētāmās programmatūras riskus, tādēļ nākamajā nodaļā tiek apskatītas risku pārvaldības un analīzes metodes.

3. RISKU PĀRVALDĪBA

Iepriekšējās nodaļās tika apskatīti dažādi standarti un vadlīnijas programmatūras drošības pārvaldībā un pārskatu veikšanā. Gandrīz visi no šiem dokumentiem kādā no soļiem iekļāva arī risku pārvaldības vai risku analīzes procesu. Darba autore secina, ka drošības novērtējumu var sniegt tikai tad, kad ir veikta iespējamo apdraudējumu analīze, šo uzdevumu veic risku pārvaldība.

Terminu vārdnīca sniedz paskaidrojumu terminam „risku pārvaldība": *Apsteidzoša rīkošanās ar riska noteikšanu un analīzi. Riska pārvaldības galamērķis ir pavērt ceļu labvēlīgai ietekmei uz pasāktiem darbiem, vienlaikus samazinot ar attiecīgo risku saistīto nelabvēlīgo ietekmi* (20).

Kaspars Līcis rakstā „Riska vadības stratēģijas uzņēmumā (1. daļa)" uzskata: *„Par risku dēvē – visu, kas var aizkavēt, samazināt un ietekmēt biznesa mērķus"* (21). Darba autore piekrīt šai definīcijai un uzskata, ka tā ir pietiekama un precīza.

North D. W. uzskata: „*Risku pārvaldība ir process, kad tiek identificēti un izpildīti pasākumi, kas var mazināt risku līdz pieņemamam līmenim, un dokumentēti galvenie nozīmīgākie lēmumi"* (35).

Darba autore secina, ka riska pārvaldības mērķis ir identificēt, novērtēt un mazināt riskus, kas apdraud biznesa mērķu sasniegšanu.

Savukārt „riska novērtējums" ir jēdziens, kurš bieži tiek lietots tieši saistībā ar drošības kontroles ieviešanu atbilstoši riska novērtējumam. Tomēr risku novērtēt var tikai subjektīvi, jo nevienā grāmatā nav definēta konkrētā riska iespējamība vai ietekme. Tātad riska novērtēšana ir process, kurā tiek pēc iespējas objektīvāk novērtēta riska iestāšanās varbūtība un iespējamā ietekme.

3.1. Risku pārvaldības metodes

Risku pārvaldība ietver riska novērtēšanas un riska mazināšanas stratēģiju izstrādes procesus, kuru rezultātā tiek izstrādātas drošības politikas, standarti, procedūras, biznesa nepārtrauktības plāni, katastrofu atveseļošanās plāni, uzticamības nodrošinājumi (37, 2. lpp).

Neeksistē noteiktu risku pārvaldības metožu saraksts, kurā varētu ieskatīties un izvēlēties kādu no atbilstošākajiem, tomēr ir vairāki pieņēmumi par to, kā organizēt risku pārvaldību uzņēmumā, kādi ir veicamie soļi, kādi mērķi sasniedzami.

CISA rokasgrāmata piedāvā risku pārvaldības metodi, izmantojot dažus veicamos soļus (11, 372. lpp). Tiek uzsvērts, ka nepieciešams izstrādāt risku pārvaldības programmu, kas ietver risku pārvaldības mērķa definēšanu un par risku pārvaldību atbildīgo personu iecelšanu, kuras nodrošinās pārvaldības plāna izpildi. Pirmais risku pārvaldības solis ir informācijas resursu identifikācija un klasifikācija, kur informācijas resursi ir tie resursi, kuriem nepieciešamas drošības kontroles. Klasifikācija var būt dažāda atkarībā no organizācijas pieņemtajiem standartiem un sistēmas uzbūves, taču tipiskā klasifikācija ir sekojoša (11, 372.lpp):

- Informācija un dati,
- Aparatūra,
- Programmatūra,
- Pakalpojumi,
- Dokumenti,
- Personāls.

Otrais solis ir iespējamo draudu un ievainojamību novērtēšana, kas attiecas uz pirmajā solī identificētajiem informācijas resursiem, un to iespējamības noteikšana. Draudi parasti tiek klasificēti (11, 373. lpp):

- Kļūdas,
- Ļaunprātīgs uzbrukums,
- Krāpšana,
- Zādzība,
- Aparatūru, programmatūru atteice.

Draudu un ievainojamību rezultātā rodas ietekme uz biznesa procesiem, parasti tie var būt finansiāli zaudējumi, reputācijas zaudēšana, biznesa aktivitāšu traucēšana, klientu zaudēšana. Riska novērtējums parasti tiek izteikts formā: iespējamība x ietekme, jeb iespējamie zaudējumi.

Kad ir veikta riska novērtēšana, seko nepieciešamo pretsoļu definēšana, piekārtošana katram no riskiem. Eksistē 4 pretsoļu veidi (17, 5. lpp):

- Riska mazināšana – piemērojot līdzekļus riska mazināšanai,
- Riska pieņemšana – pieņemt riska iestāšanos, ņemot vērā organizācijas drošības politiku (plānot resursu rezerves),

- Izvairīšanās no riska – nepieļaut riska iestāšanos,
- Riska novirzīšana – riska novirzīšana citām pusēm, piemēram, apdrošināšana.

Microsoft ir radījis karkasu drošības risku pārvaldībai, kurā tiek piedāvāts risku pārvaldību veikt ar 5 aktivitāšu palīdzību (36, 98. lpp):

- Ieviešana – projekta definīcijas ieviešana: visām drošības pārvaldībā iesaistītajām personām jādefinē mērķi, pieņēmumi un ierobežojumi. Rezultātā iegūstot projekta vīzijas un sfēras apstiprinājumu;
- Plānošana – drošības pārvaldības procesa novērtējums un analīze: organizācijas novērtējumi, aktīvu novērtējums, draudu identifikācija, ievainojamību novērtējumi, drošības risku novērtējumi. Rezultātā iegūstot projekta plānu;
- Izstrāde – drošības korekciju izstrāde: vienībtestēšana, pretsoļu kvalitātes validācija;
- Stabilizēšana – drošības korekciju testēšana, resursu funkcionālā testēšana;
- Uzstādīšana – drošības politikas un pretsoļu uzstādīšana.

Programmatūras izstrādes institūta (*Software Engineering Institute*) tehniskajā ziņojumā „Programmatūras risku pārvaldība" tiek sniegts paraugs risku pārvaldības metodei, uzsverot, ka risku pārvaldības process ir nepārtraukts, tiek definētas sekojošas aktivitātes (33, 28. lpp):

- Identifikācija – lai risks varētu tikt pārvaldīts, tas ir jāidentificē, pirms risks kļūst par problēmu;
- Analīze – analīzes aktivitāte ir riska datu pārvēršana lēmumu pieņemoša riska informācijā;
- Plānošana – plānošanas aktivitāte riska informācijas pārvēršana lēmumos un darbībās, riska pretsoļu definēšana;
- Sekošana – aktivitāte iekļauj riska statusa apsekošanu un uzlabojošo darbību veikšanu;
- Kontrole – riska kontrole labo novirzes no plānotajām riska aktivitātēm, kontrolē riska darbību. Riska kontrole ir svarīgs riska pārvaldības solis, kas uzlabo riska pārvaldības procesu;
- Komunikācija – komunikācija ir riska pārvaldības centrā, ņemot vērā to, ka šī aktivitāte ir saskatāma visos pārvaldības soļos un tā ir kritiski svarīga, bez komunikācijas nav iespējama kvalitatīva un uzticama riska pārvaldība.

Žurnāla „IS Control Journal" 2008. gada 6. Izdevumā iekļauts raksts „Vai IT risku ir vērts kontrolēt?", uzsverot, ka grūtākais posms izstrādes procesā ir investīciju sadalīšana riska pārvaldībai (7). Rakstā definētas riska pārvaldības fāzes (7):

1. Uzbrukumu un ievainojamību identificēšana – nosakot apskatāmo intervālu un vides atkarības, tiek identificēti iespējamie uzbrukumi un ievainojamības;

2. Riska iespējamības novērtēšana – riski tiek klasificēti un, balstoties uz ekspertu spriedumiem un iepriekšējo pieredzi, tiek noteikta riska iespējamība;
3. Riska ietekmes novērtēšana – pēc iespējas objektīvāk novērtēt riska iespējamo ietekmi uz sistēmas darbību;
4. Risku prioritāšu sadalīšana – risku prioritāšu sadalīšana notiek, izmantojot risku iespējamības un risku ietekmes novērtējumus, veidojot risku iespējamības un ietekmes matricu;
5. Risku pretsoļu definēšana – organizācija definē piemērotākos pretsoļus katram riskam;
6. Atlikušo risku novērtēšana – pēc riska mazināšanas kontroles procesa, tiek novērtēts atlikušais risks.

Informācija par apskatītajām risku pārvaldības metodēm darba autore ir apkopojusi tabulā, kurā tiek apskatīti metodēs izmantotie pārvaldības soļi un to izmantošana apskatītajos dokumentos (skatīt 3.1.1. tabulu).

3.1.1. tabula

Ieteikumos iekļauto riska pārvaldības soļu kopsavilkums

Resurss / Solis	**CISA rokasgrāmata**	**Microsoft**	**Programmatūras izstrādes institūts**	**Žurnāla *Control* raksts**
Informācijas resursu identifikācija	✓			
Draudu klasifikācija	✓			
Risku identifikācija	✓	✓	✓	✓
Risku novērtēšana	✓	✓	✓	✓
Risku pretsoļu definēšana	✓	✓	✓	✓
Pretsoļu uzstādīšana		✓	✓	
Risku sekošana un kontrole			✓	✓

✓ - resursā tiek izmantots risku pārvaldības solis

Kā redzams, vienojošie soļi ir risku identifikācija, novērtēšana un pretsoļu definēšana. Ņemot vērā iepriekš apskatītos ieteikumus, darba autore definē riska pārvaldības soļus, kuri veicami veselības aprūpes programmatūras risku pārvaldības procesā:

1. Riska pārvaldības mērķa, metodes un sagaidāmo rezultātu definēšana – šis solis netiek ieteikts, tomēr autore uzskata, ka solis paskaidro riska pārvaldības nepieciešamību,
2. Apdraudējumu klasifikācija – tiek ņemts vērā CISA rokasgrāmatas ieteikums, pēc autores domām, tas palīdz efektīvāk identificēt riskus,
3. Risku identifikācija – identificēt programmatūras drošības riskus,

4. Risku ietekmes un iespējamības novērtēšana.

Ierobežotā darba izstrādes laika dēļ netiks apskatīta pretsoļu definēšana un uzstādīšana, kā arī to sekošana.

3.2. Risku analīzes metodes

Pēc CISA rokasgrāmatas riska analīze ir audita plānošanas daļa, kas palīdz identificēt riskus un ievainojamības, pēc kuriem auditors var noteikt nepieciešamās kontroles, lai šos riskus mazinātu (11, 28. lpp).

Risku analīzei pamatā ir divas metodes: kvalitatīvā un kvantitatīvā. Kvalitatīvā riska analīze izceļ sistēmas problēmas un pašreizējo stāvokli. Kvantitatīvā analīze identificē vietas, kur būtu nepieciešamas drošības kontroles. Eksistē arī trešā riska analīzes metode, kas tiek saukta par hibrīdo metodi, jo tā ir abu iepriekšminēto metožu apvienojums (34).

Eksistē vairākas kvalitatīvās analīzes tehnikas. Tilman Resche rakstā „Riska analīzes metodes - pārskats" ir salīdzinātas riska analīzes tehnikas, par pamatu ņemot tādus kritērijus kā sistēmas izstrādes dzīves cikla fāze, plusi, mīnusi, iegūstamie rezultāti, nepieciešamās prasmes, izmaksas un citus. Kopā analizētas 10 dažādas riska analīzes tehnikas: „Ja nu", Neveiksmju un seku analīze (FMEA & FMECA), Briesmu un operāciju pētīšana (HAZOP), Cilvēku kļūdu analīze, Uzticamības bloku diagrammas, Trūkumu koka analīze, Notikumu koka analīze, Pirmās kārtas uzticamības metodes (F.O.R.M.), Monte Carlo metodes, Risku iespējamības un drošības novērtēšana (PRA & PSA) (38). Ņemot vērā šī raksta novērtējumus, lētākā tehnika ir „Ja nu" tehnika, kuras rezultātā tiek iegūts risku un drošības apdraudējumu saraksts. Par visdārgāko, toties efektīvāko tehniku tiek uzskatīta Risku iespējamības un drošības novērtēšana, kuras rezultātā tiek krasi mazināta riska iestāšanās iespējamība un būtiski uzlabota sistēmas drošība.

Programmatūras „Ārsta Birojs 3" risku analīzei tiks izmantota „Ja nu" metode, kas ļauj identificēt un novērtēt riskus. Pielietojot šo metodi, tiks izpildīts uzstādītais risku pārvaldības mērķis.

4. PROGRAMMATŪRAS „ĀRSTA BIROJS 3” DROŠĪBA

Bahatia M. grāmatā ir definētas galvenās drošības pārvaldības daļas, kuras iekļauj drošības politikas mērķa definēšanu, apskata fizisko, vides, administratīvo un personāla drošību, tiek uzsvērta katastrofas plāna nepieciešamība un organizācijas drošības nodaļas nepieciešamība, kā arī sistēmas lietotāju apmācības svarīgums (10, 251. lpp). Novērtējot programmatūras „Ārsta Birojs 3” drošības risinājumus, tiks ņemti vērā dažādu organizāciju un autoru publicētie ieteikumi un starptautiski atzītie standarti, kas tika apskatīti iepriekšējās nodaļās.

Programmatūras drošības pārbaude tiks veikta, balstoties uz risku analīzes procesā identificētajiem apdraudējumiem (skatīt 4.1.1. nodaļu). Rezultāti un secinājumi iegūti, veicot interviju ar izstrādātāju un pārskatot programmatūras dokumentāciju un darbību.

Intervija ar izstrādātāju tika veikta programmatūras „Ārsta Birojs 3” izstrādes birojā, intervijā piedalījās divi pārstāvji no izstrādātāja puses – ieviešanas nodaļas vadītājs un izstrādes nodaļas vadītājs, kā arī darba autore kā intervētājs. Intervijas laikā tika apspriesti programmatūras drošības jautājumi, tika sniegtas atbildes uz darba autores sastādītajiem jautājumiem. Pilnu intervijas protokolu skatīt 2. pielikumā.

Lai novērtētu informācijas un datu drošību, nepieciešams definēt, kāda informācija un kādi dati tiek aizsargāti, tātad kādus datus apstrādā programmatūra „Ārsta Birojs 3”. Intervijas laikā tika definēta sekojoša informācijas uzkrāšana:

1) Pacienta dati,
2) Ārsta dati,
3) Pacienta ambulatorās ārstēšanas dati,
4) Pacienta stacionārās ārstēšanas dati,
5) Pacienta nosūtījumi uz izmeklēšanu,
6) Veikto izmeklējumu dati.

Tā kā tiek uzkrāta informācija par pacienta veselību, dati tiek uzskatīti par sensitīviem, līdz ar to ir piemērojami stingri drošības nosacījumi. Informācijai ir jānodrošina konfidencialitāte, integritāte un pieejamība.

4.1. Drošības pārbaude

Drošības pārbaude tiek veikta, lai novērtētu programmatūras „Ārsta Birojs 3" drošības risinājumus un to atbilstību vispārpieņemtajiem standartiem un ieteikumiem, kas tika apskatīti 1. nodaļā. Šīs drošības pārbaudes mērķis ir novērtēt kādā līmenī tiek novērsti vai mazināti programmatūras drošības riski, kā arī ieteikt drošības uzlabojumu risinājumus.

Drošības pārbaude ir organizēta pa soļiem, kurus definējusi darba autore:

1) Risku pārvaldība, ietverot risku identifikāciju un novērtēšanu,
2) Programmatūras drošības risinājuma novērtējums katram no identificētajiem riskiem, grupējot tos pēc draudu klasifikācijas, ņemot vērā pirmajā nodaļā apskatītos drošības risinājumu ieteikumus (skatīt 2.1.1. tabulu).

4.1.1. Programmatūras „Ārsta Birojs 3" risku pārvaldība

Programmatūras „Ārsta Birojs 3" risku pārvaldība tiks veikta atbilstoši 3.1. nodaļā definētajiem riska pārvaldības soļiem.

4.1.1.1. Riska analīzes mērķis, metode, sagaidāmie rezultāti

Risku analīzes mērķis ir identificēt programmatūras „Ārsta Birojs 3" riskus, novērtējot to iespējamību un ietekmi, izmantojot kvalitatīvo risku analīzes metodi. Risku pārvaldības rezultātā tiek izcelti iespējamie drošības apdraudējumi.

Risku analīzi veic darba autore, ņemot vērā iepriekš apskatītos risku analīzes ieteikumus, risku novērtējumi tiek veikti balstoties uz autores pieredzi. Risku identifikācijas procesā tiek ņemti vērā pirmajā nodaļā apskatītie drošības apdraudējumi.

4.1.1.2. Apdraudējumu klasifikācija

Ērtākai iespējamo apdraudējumu identificēšanai, apdraudējumi tiek klasificēti, klasifikācijā tiek iekļauti apdraudējumi, kas ir saistīti ar programmatūras drošības draudiem:

- Informācijas un datu apdraudējumi,
- Programmatūras ieviešanas un uzturēšanas apdraudējumi,
- Izstrādes procesa apdraudējumi,
- Cilvēka radītie apdraudējumi.

4.1.1.3. Risku identifikācija un novērtēšana

Risku identifikācijas procesā riski tiek identificēti grupās, ņemot vērā iepriekš definēto apdraudējumu klasifikāciju. Katram riskam tiek piekārtots identifikators. Riskam tiek novērtēta tā iespējamība un ietekme. Ietekme šajā gadījumā tiek vērtēta attiecībā uz drošības ievainojamību, tātad, kā riska iestāšanās var apdraudēt programmatūras drošību. Identificētos riskus un to novērtējumu skatīt 4.1.1.tabulā.

4.1.1. tabula

Identificētie programmatūras „Ārsta Birojs 3" riski

Identifikators	**Risks**	**Iespējamība**	**Ietekme**
1. Informācijas un datu riski			
R-01	Neatļauta piekļuve datiem	Zema	Augsta
R-02	Dokumentu un informācijas zādzība	Vidēja	Augsta
R-03	Ļaunprātīga informācijas izmantošana	Augsta	Zema
R-04	Datu integritātes zudums	Vidēja	Augsta
R-05	Nejauša datu zaudēšana	Augsta	Augsta
R-06	Dati nav pieejami	Zema	Augsta
2. Programmatūras ieviešanas un uzturēšanas riski			
R-07	Neapejamas kļūdas rašanās	Vidēja	Augsta
R-08	Nekorekta un nepietiekama dokumentācija	Vidēja	Vidēja
R-09	Programmatūras neatbilstība prasībām	Augsta	Augsta
R-10	Neērta lietotāja saskarne	Vidēja	Zema
3. Izstrādes procesa riski			
R-11	Nepilnīga izstrādes procesa un drošības pārvaldība	Vidēja	Augsta
R-12	Neatbilstoša metodoloģijas izvēle	Zema	Zema
R-13	Nekorekts un neoptimāli izstrādāts kods	Augsta	Augsta
R-14	Ļaunprātīga testa datu izmantošana	Zema	Augsta
4. Cilvēka radītie riski			
R-15	Programmatūra tiek nepareizi lietota	Augsta	Augsta
R-16	Programmatūra tiek nepareizi konfigurēta	Augsta	Augsta
R-17	Lietotājs nav informēts par savām tiesībām un pienākumiem	Vidēja	Vidēja
R-18	Klientu neinteresē fiziskās drošības ievērošana (durvju un logu slēgšana, paroles glabāšana pieejamā vietā)	Vidēja	Augsta

4.1.2. Informācijas un datu riski

Pirmajā nodaļā tika apskatīts informācijas drošības jēdziens, kurš iekļauj trīs pamata lietas, kas nosaka to, vai informācija ir droša: konfidencialitāte, integritāte un pieejamība. Šajā apakšnodaļā tiek apskatīti informācijas un datu risku (skatīt 4.1.1. tabulu) drošības risinājumi grupēti pēc šiem kritērijiem.

4.1.2.1. Konfidencialitāte

Lai nodrošinātu informācijas konfidencialitāti, nepieciešams realizēt vairākus drošības pasākumus. Pirmkārt, jānovērtē piekļuves drošību, pētot, kādas autorizācijas vai autentifikācijas metodes tiek lietotas. Piekļuves ierobežošana mazina neatļautas piekļuves risku R-01 un datu zādzības risku R-02 (skatīt 4.1.1. tabulu). Autorizācijas kontrole pamatā nodrošina funkcionalitāti, kas nosaka vai noteiktai lietotāja identifikatora un paroles kombinācijai ir piešķirtas tiesības piekļūt sistēmai, savukārt autentifikācija apstiprina arī to, ka identifikators un parole pieder personai, kura mēģina piekļūt sistēmai (8, 11. lpp). Programmatūras ĀB3 piekļuves nodrošināšanai tiek izmantota autorizācijas metode, nepieciešams norādīt lietotāja vārdu un paroli. Tā kā programmatūras piekļuvei tiek izmantota parole, parolei jātiek glabātai drošā vietā un veidā. ĀB3 paroles tiek glabātas šifrētā veidā datu bāzē, izmantojot SHA-1 algoritmu, piekļuves mēģinājuma laikā parole netiek sūtīta uz datubāzi, bet tiek salīdzinātas šifrētās paroles kontrolsummas.

Ļoti būtiski ir parūpēties par drošu paroli. Tommie W.Singleton rakstā „What Every IT Auditor Should Know About Access Controls" sniedz drošas paroles politikas ieteikumus, sastādot tabulu, kuru darba autore intervijas laikā lūdza aizpildīt ĀB3 izstrādātājiem (skatīt 4.1.2. tabulu) (8, 12. lpp).

4.1.2. tabula

Drošas paroles politikas metodes

Metode	**Paskaidrojums**	**Tiek izmantots ĀB3**
Automātisks laika ierobežojums parolei	Kāds noteikts periods, pēc kura parole ir jāmaina	ĀB3 piedāvā iestatījumus, kuros klientam iespējams norādīt paroles derīguma periodu
Stipra parole	8 vai vairāk zīmes, iekļauti lielie burti, cipari, speciālās zīmes	ĀB3 piedāvā iestatījumus, kuros klientam iespējams norādīt paroles minimālo garumu, ciparu obligātumu parolē
Paroles aizsardzība	Neizmantot piezīmju lapiņas tuvu datoram, neatbildēt uz nezināmas,	Šāda drošības ievērošana ir klienta kompetencē. Piekļuve pēc 3. neveiksmīgā

Metode	Paskaidrojums	Tiek izmantots ĀB3
	nekompetentas personas lūgumu pēc paroles, slēgt piekļuvi pēc 3. neveiksmīgā autorizācijas mēģinājuma	mēģinājuma netiek slēgta
Redzamības ierobežojumi	Rādīt tikai lietotājam nepieciešamos skatus, datus	ĀB3 nodrošina administrēšanas iespējas pa lomām, kur iespējams norādīt skatus, kurus attēlot lomai
Ierobežots administratoru skaits	Ierobežots administratoru skaits un droša piekļuve, neizmantot iebūvētos administratoru kontus	Administratoru skaitu kontrolē klients. Iebūvētie administratoru konti netiek izmantoti
Paroles iznīcināšana	Efektīva paroles slēgšana pēc lietotāja darba līguma beigām	Lietotāja kontu slēdz administrators, izmantojot ĀB3 izstrādātu funkciju

Izvērtējot sniegtās atbildes, darba autore secina, ka piekļuves drošība galvenokārt ir atkarīga no klienta rīcības, kas, pēc autores domām, var tikt uzskatīts par vidējas pakāpes risku, jo klients bieži vien neapzinās drošības nepieciešamību. Tādēļ ieteicams ieviest obligātu paroles derīguma perioda un stipras paroles izmantošanu, kā arī slēgt lietotāja kontu pēc izvēlēta skaita neveiksmīgā pieslēgšanās mēģinājuma. Protams, jāņem vērā arī klienta vēlmes un prasības, tomēr būtu nepieciešams klientam norādīt uz šīs drošības kontroles obligāto nepieciešamību.

Lai nodrošinātu informācijas pieejamību tikai personām, kurām informācija ir nepieciešama, tiek izmantota lomu un tiesību sadalīšana. Arī ĀB3 izmanto lomas, kurām pēc tam tiek definētas tiesības. ĀB3 eksistē loma „Administrators", kuram tiek piešķirtas visas ĀB3 esošās tiesības, kā arī šai lomai ir tiesības veidot citas lomas, kurām tiek pievienoti konkrēti lietotāju konti. ĀB3 tiek definētas atsevišķas tiesības skatīt, pievienot, labot un dzēst informāciju kādā konkrētā skatā jeb informācijas logā.

Atbilstoši Cheryl Traverse rakstītajam: *„Modelī 'Aizliegt visu, piešķirt pēc izņēmuma' (DAPE) katra lietotāja vai lietotāju grupas konts sākumā ir bez jebkādām tiesībām jeb nav redzams neviens skats sistēmā"*, tiek nodrošināta jaunizveidotās lomas tiesību ierobežošana (9, 31. lpp). Arī ĀB3 izmanto šo pieeju visām lomām, kuras izveido administrators.

Svarīga ir arī lietotāju un tiem piešķirto tiesību kontrole. Pasaulē tiek piedāvāti vairāki rīki, kas ģenerē lietotāju, lomu un tiesību sarakstus, piemēram, kā uzskata Tommie W.Singleton: „*Dumpsec rīks savāc informāciju par lietotājiem, to tiesībām, ģenerē tabulu, kurā auditors var ērti pārskatīt lietotāju un tiesību atbilstības, administratoru skaitu, lietotājus, kuru kontiem vajadzētu būt slēgtiem"* (8, 13. lpp). Tomēr piemērot rīku var būt

sarežģītāk, kā izstrādāt atskaiti programmatūras ietvaros, šādi rīkojās arī ĀB3 izstrādātājs, ĀB3 piedāvā atskaiti, kurā redzamas lomas, tām piesaistītās tiesības un lietotāji.

John Moynihan rakstā „Managing the Insider Threat: Data Surveillance" sniedz piemēru datu uzraudzības nepieciešamībai: *„Ziņkārības vadīts, slimnīcas darbinieks apskata populāra atlēta, kuru tikko atpazinis slimnīcas telpās, slimības vēsturi un uzņemšanas ierakstus"* (4, 38.lpp). Šādu datu piekļūšanu nav iespējams programmiski ierobežot, jo iespējams, ka darbinieka pienākumos tiešām nepieciešams piekļūt pacientu uzņemšanas ierakstiem. Tomēr, lai kontrolētu lietotāja darbības, tiek izmantota žurnalēšanas metode jeb auditācija, kas mazina identificēto risku R-03 - Ļaunprātīga informācijas izmantošana (skatīt 4.1.1. tabulu).

ĀB3 nodrošina auditācijas pierakstu veidošanu divos līmeņos. Pirmkārt, sistēma veic jebkuras rīcības auditācijas pieraksta veidošanu un saglabāšanu atsevišķā datubāzē, kur tiek saglabāts lietotāja identifikators, veiktā darbība, datums un laiks, kad darbība tiek veikta. Otrs līmenis parūpējas par datu pievienošanas, labošanas un dzēšanas darbību auditēšanu. Veicot darbību, auditācijas datubāzē tiek saglabāts lietotāja identifikators, pacienta (kura dati tiek mainīti) identifikators, veiktā darbība, darbības datums un laiks. Ja tiek veikta dzēšanas darbība, papildus tiek saglabāti visi dati, kuri tiek dzēsti.

Auditācijas pierakstu veidošanu klients var ieslēgt vai izslēgt, izmantojot ĀB3 iestatījumus, jo šādas ir klienta prasības. Darba autore iesaka ĀB3 izstrādātājam ieviest obligātu auditācijas pierakstu veidošanu, vismaz otrā līmeņa auditācijai, jo klients bieži vien nenovērtē šī drošības pasākuma nepieciešamību, līdz brīdim, kad tiek pazaudēti kādi nozīmīgi dati, kurus bez auditācijas ierakstiem vairs nav iespējams atgriezt. Arī Cheryl Traverse, rakstot par veselības aprūpes sistēmu informācijas validāciju, iesaka: *„Veselības aprūpes sistēmām vajadzētu apsvērt sesijas darbību ierakstīšanas nepieciešamību gan komandrindas, gan grafiskās lietotnēs"* (9, 32. lpp).

Ja tiek izmantota auditācija, tātad nepieciešams arī rīks vai sistēma, kā auditācijas datus apstrādāt vai analizēt. Pasaulē tiek piedāvāti rīki, kas analizē auditācijas pierakstus, piemēram, Masačūsetsas ieņēmumu departaments ir izstrādājis Transakciju izsekošanas sistēmu (*Transaction Tracking System*), kas tik pat kā aizstāj auditācijas pierakstu izmantošanu, jo analizē SQL pieprasījumus, sniedzot konfidenciālo datu izmantošanas atskaites (4, 38. lpp). Eksistē arī organizācijas, kas kā pakalpojumu piedāvā veikt auditācijas pierakstu analīzi. Kā tika noskaidrots intervijas laikā ĀB3 programmatūrā nav realizēta auditācijas pierakstu analīze, auditācijas ierakstu apskati veic izstrādātājs, ja to ir pieprasījis klients. Tā kā auditācijas ierakstu analīzes rīka izstrāde prasa papildus resursus, tad to nepieciešamību galvenokārt nosaka klients.

4.1.2.2. Integritāte

Kā definē Wikipedia: *„Datu integritāte ir dati, kam ir pilnīga struktūra. Lai dati būtu pilnīgi, tiem jāatbilst sekojošiem raksturlielumiem: biznesa prasību likumiem, datu daļu sakarību likumiem, datumiem, definīcijām un ciltsrakstiem”* (25). Ir vairākas metodes, kā nodrošināt datu integritāti (identificētais risks R-04 – skatīt 4.1.1. tabulu). Viena no metodēm ir datu validācija. Datu validācija var tikt kontrolēta datu bāzes līmenī (noteikti datu tipi, garuma, vērtību, formas ierobežojumi) vai programmatūras līmenī, kur tiek izstrādātas datu validācijas kontroles. ĀB3 izmanto abus iepriekš minētos līmeņus, tiek kontrolēta datu atbilstība datu tipam, garuma ierobežojumi, konkrētas formas nepieciešamība, konkrētu vērtību ierobežojumi, kā arī programmatūras līmenī tiek izstrādāti biznesa prasību likumi, piemēram, sākuma datumam jābūt mazākam par beigu datumu, personas vecums nedrīkst būt negatīvs skaitlis, personas kodam jābūt noteiktā formā un citi.

Bieži vien lietotājs saskaras ar nejaušu informācijas zaudēšanu (identificētais risks R-05 – skatīt 4.1.1. tabulu), vai nekorektas informācijas ievadīšanu, tādēļ būtiski ir kontrolēt arī lietotāja darbības. Pēc autores domām, obligāts nosacījums ir lietotāja apstiprinājums pirms datu dzēšanas. ĀB3 ir izstrādājis funkciju, kas lietotājam sniedz apstiprinājuma ziņojumu pirms jebkādu datu dzēšanas, tādā veidā aizsargājoties no nejaušas datu zaudēšanas. Pēc ĀB3 izstrādātāja uzskatiem, ir nepieciešams kontrolēt arī lietotāja nejaušu loga aizvēršanas mēģinājumu, pirms ir saglabāti veiktie labojumi, tādēļ arī pirms loga aizvēršanas, gadījumā ja ir veiktas kādas izmaiņas, tiek parādīts paziņojums ar jautājumu, vai lietotājs vēlas izmaiņas saglabāt. Pēc darba autores domām, nepieciešams lietotāja apstiprinājums arī svarīgu datu labošanas gadījumā, tādēļ ĀB3 izstrādātājam tiek ieteikts ieviest datu labošanas apstiprināšanu kritiskiem datiem, piemēram, pacienta izmeklējuma rezultātu datu labošanas gadījumā.

4.1.2.3. Pieejamība

CobIT vadlīnijas DS4 Pakalpojumu nepārtrauktības nodrošināšanas procesa aprakstā teikts: *„Nepieciešamība nodrošināt nepārtrauktus IT pakalpojumus prasa nepārtrauktības plāna izstrādi, uzturēšanu un testēšanu, attālinātas rezervju kopiju novietnes uzturēšana un nepārtrauktības plāna periodisku apmācības organizēšanu. Efektīvs nepārtrauktības pakalpojuma process mazina iespējamību un ietekmi nopietniem IT pakalpojumu traucējumiem nozīmīgās biznesa funkcijās un procesos”* (24, 113. lpp). Tātad, lai nodrošinātu programmatūras procesu nepārtrauktību un mazinātu datu nepieejamības risku R-06 (skatīt

4.1.1. tabulu), nepieciešams izstrādāt, uzturēt un testēt nepārtrauktības nodrošināšanas plānu, kā arī parūpēties par rezerves kopiju veidošanu un drošu glabāšanu.

ĀB3 eksistē nepārtrauktības plāns, kurā ietverti pasākumi, kas veicami, lai nodrošinātu programmatūras pieejamību 24 stundas dienā. Ja rodas kādi traucējumi, problēma var tikt atrisināta, vadoties pēc šī plāna, vidēji stundas laikā, kad tiek novērsti problēmas cēloņi un veikti informācijas atjaunošanas pasākumi, ja tādi ir nepieciešami. Lai nodrošinātu iespēju atjaunot informāciju, katru dienu tiek veikta rezerves kopiju izveidošana, katras nedēļas beigās tiek izveidota rezerves kopija un novietota attālināti. Rezerves kopiju politika ir stingri noteikta un dokumentēta.

4.1.3. Programmatūras ieviešanas un uzturēšanas riski

Kā zināms, nododot programmatūru klientam, ar lielu iespējamību var rasties kļūdas, tātad svarīgi ir definēt kļūdu apstrādes procesu, kas nodrošinās operatīvu kļūdas labošanu un neapejamas kļūdas rašanās riska R-07 (skatīt 4.1.1.tabulu) mazināšanu. Šim nolūkam tiek sniegta garantija, kuras laikā tiek bez maksas novērstas nepilnības un kļūdas, pēc garantijas beigām parasti tiek noslēgts uzturēšanas līgums, kurš paredz kļūdu labošanu iepriekš norunātās laika robežās. Ir noteikts kļūdu apstrādes un labošanas algoritms. Klients saņemot programmatūras kļūdu, uz ekrāna redz kļūdas paziņojumu un identifikatoru. Šis identifikators tiek nodots izstrādātājam un izstrādātājs kļūdu datubāzē ātri var atrast kļūdu pēc tās identifikatora. Datubāzē glabājas detalizēts kļūdas apraksts, no kā iespējams noteikt kļūdas cēloni. Izstrādātājs kļūdu novērtē un paziņo klientam kļūdas labošanai nepieciešamos laika resursus. Kļūda tiek izlabota norunātajā laikā un atkarībā no kļūdas cēloņa, izstrādātājs nodod klientam jaunu programmatūras versiju vai kļūdainās komponentes jaunu versiju.

Tomēr, lai mazinātu kļūdas rašanās iespējamību, nepieciešams programmatūru testēt. Programmatūra tiek testēta, lai atrastu iespējamās kļūdas un pēc tam tās novērstu. Eksistē vairākas testēšanas metodes un izstrādātājs izvēlas sev piemērotāko un efektīvāko. Testējot programmatūras drošību, nepieciešama cita pieeja, kā uzskata O. Aras un B. L. Ciaramitaro: *„Tradicionāli testētāja mērķis ir apstiprināt, ka programmatūra izpilda visas funkcionālās un operāciju prasības. Tomēr, testējot drošību, testētājam ir jādomā kā uzbrucējam un testa scenāriji jāizstrādā tā, lai pārbaudītu potenciālās ievainojamības. [..] Caurlaidības testi vēsturiski tiek lietoti, lai noteiktu drošības trūkumus”* (5, 51. lpp). Kā tika noskaidrots intervijas laikā, ĀB3 izstrādātājs veic testēšanu, izmantojot Melnās kastes testēšanas metodi. Ir izstrādāti testa piemēri, pēc kuriem vadoties, tiek veikta testēšana. Testēšanas rezultāti tiek dokumentēti bezmaksas kļūdu apsekošanas sistēmā Mantis, kas ērti ļauj pārvaldīt kļūdu

statusu. Kā piemin ĀB3 izstrādātājs, nākotnē plānots ieviest regresīvo testēšanu, izmantojot jau iegādātu speciālu rīku. Autore apstiprina Mantis sistēmas efektivitāti, jo autore šo sistēmu lieto un uzskata, ka tā kvalitatīvi izpilda visas tai paredzētās funkcijas, kā arī veic testēšanas dokumentācijas funkcijas.

Nodrošinot izstrādes procesa kvalitāti un veiksmīgu rezultātu, nepieciešams izstrādāt dokumentus, kas apliecinās projekta nodevumu kvalitāti un funkcionalitātes atbilstību. Izstrādājot kvalitatīvu programmatūras dokumentāciju, tiek mazināts identificētais risks R-08 Nekorekta un nepietiekama dokumentācija (skatīt 4.1.1.tabulu). Zināms, ka uzsākot projektu, tiek izstrādāts programmatūras projektējums, tiek definētas konkrētas prasības, tiek izstrādāta un testēta programmatūra un visbeidzot tā tiek nodota klientam. Šo procesu secība ir atkarīga no metodoloģijas izvēlēs, bet tomēr lielākoties ir pieņemts, ka programmatūrai eksistē dokumentācija, parasti tā ietver: Programmatūras projektējuma aprakstu, Programmatūras prasību specifikāciju, Testēšanas dokumentāciju, Lietotāja dokumentāciju un citus dokumentus, atkarībā no vajadzībām.

Programmatūras projektējuma aprakstā tiek definētas programmatūras galvenās prasības, mērķi, ierobežojumi u.c.. Tā kā programmatūra ĀB3 tiek piedāvāta kā produkts, projekta gaitā netiek izstrādāts pilns programmatūras projektējums, bet tikai papildinājumu projektējums, kurā tiek definēti papildinājumu ierobežojumi un prasības.

Programmatūras prasību specifikācija definē konkrētas programmatūras prasības, uzskaitot visas nepieciešamās funkcionālās, nefunkcionālās un drošības prasības. ĀB3 izstrādātājs norāda, ka Programmatūras prasības specifikācija kā dokuments tiek izstrādāta tikai gadījumā, ja to pieprasa klients, šī dokumenta vietā tiek izmantota bezmaksas kļūdu apsekošanas sistēma Mantis, kurā tiek uzskaitītas visas klienta prasības (prasības tiek definētas, veicot intervijas ar klientu), tālāk izmantojot šo sistēmu, prasības tiek izstrādātas, testētas un atkļūdotas, līdz prasība tiek atzīta par izpildītu. Darba autores pieredze rāda, ka bieži vien IS izstrādātāji Programmatūras prasību specifikāciju vietā izmanto citus risinājumus, kas aizstāj prasību uzskaitīšanu, tomēr tas nespēj aizstāt dokumenta vērtību attiecībā uz nepieciešamo klienta apstiprinājumu. Pastāv risks, ka izstrādātā programmatūra neatbildīs klienta vajadzībām (identificētais risks R-09 – skatīt 4.1.1. tabulu), un ja neeksistē šis dokuments, tad izstrādātājs nonāk juridiski bīstamā situācijā, jo nav pierādījumu par to, kādas prasības klients ir definējis, jo konkrēts dokuments neeksistē un nav apstiprināts. Lai mazinātu šo risku, ieteicams izstrādāt Programmatūras prasību specifikāciju, un programmatūras izstrādi veikt tikai pēc apstiprinātām prasībām. Šajā gadījumā izstrādātājam ieteicams dokumentēt programmatūras ĀB3 papildinājumu prasības.

Tā kā pastāv risks R-10 Neērta lietotāja saskarne (skatīt 4.1.1. tabulu), nepieciešams specificēt arī lietotāja saskarnes prasības, šāda sadaļa ietilpst programmatūras prasību specifikācijā, vai izmantot prototipēšanas metodi izstrādes procesa laikā. Kā minēts rakstā „Kas ir prototipēšana?": *„[..] Prototipi tiek izstrādāti, lai palīdzētu sistēmas dizainerim izstrādāt informācijas sistēmu, kas ir intuitīva un ērti lietojama gala lietotājam. Prototipēšana ir iteratīvs process, kas ir daļa no analīzes fāzes sistēmas izstrādes dzīves ciklā"* (40). ĀB3 izstrādātājs izmanto prototipēšanas metodi, kur lietotāja saskarne pēc iespējas pietuvināti tiek veidota kopā ar klientu interviju laikā, tādā veidā klients programmatūrā redz daļēji paša veidotu lietotāja saskarni, kas nodrošina lietošanas ērtumu un klienta apmierinātību.

4.1.4. Izstrādes procesa riski

Izstrādājot programmatūru, nepieciešams nodrošināt arī šī procesa drošību. Izstrādes procesa drošība ir atkarīga no vairākām komponentēm: projekta pārvaldības, izstrādes metodoloģijas, koda drošības, testēšanas procesa, risku analīzes. Kā zināms projekta vadītājam ir jārūpējas par izstrādes procesa veiksmīgu rezultātu sasniegšanu, jānodrošina komunikāciju starp izstrādātāju un klientu, jāpieņem svarīgi lēmumi attiecībā uz resursu sadali un citi pienākumi, līdz ar to projekta vadītājs ir lielā mērā atbildīgs par izstrādes procesa drošību, kas attiecas uz veiksmīga projekta nodošanu un klienta apmierinātības nodrošināšanu, šādā veidā projekta vadītājs mazina nepilnīga izstrādes procesa un drošības pārvaldības risku R-11 (skatīt 4.1.1. tabulu).

Kā minēts ISO standartā, drošības politikas mērķis ir uzstādīt pārvaldības virzienu un informācijas drošības atbalstu atbilstoši biznesa prasībām un piesaistītajiem likumiem un ierobežojumiem (17, 7. lpp). Pēc ITIL uzskatiem - drošības pārvaldības pamatā ir informācijas drošība, kur tiek apskatīta informācijas konfidencialitāte, integritāte un pieejamība (27). Pēc autores domām, organizācijai obligāti ir jāizstrādā drošības pārvaldības plāns un tas jādokumentē. ĀB3 izstrādātājs atzīst, ka drošības pārvaldība netiek dokumentēta, bet tiek veiktas regulāras drošības pārbaudes, tomēr pēc autores uzskatiem ar to nepietiek, tādēļ ieteicams dokumentēt drošības pārvaldības plānu.

Plašāk tiek apskatīts kā izstrādes procesa drošību ietekmē metodoloģijas izvēle. Dave Henderson rakstā „Issues With Auditing the Systems Development Process" piemin: *„Pareiza sistēmas izstrādes metodoloģijas izmantošana ir nozīmīga, lai mazinātu programmatūras projekta izgāšanās riskus, kas saistīti ar izstrādes procesu. [..] Tā vietā lai punktuāli sekotu metodoloģijai, izstrādātājs parasti modificē izvēlēto metodoloģiju, balsoties*

uz dažiem faktoriem, ieskaitot organizācijas ierasto metodoloģiju, izstrādes saturu, metodoloģijas lomām un citiem" (2).

Tā kā ĀB3 tiek piedāvāts klientam kā gatavs produkts, tad nepieciešams nopietni apsvērt ieviešanas procesa metodoloģijas izvēli. Bieži vien produkts tiek papildināts, atbilstoši klienta prasībām, līdz ar to nepieciešams arī papildinājumu vai uzlabojumu izstrādes process. ĀB3 izvēlētā metodoloģija ir Spējā programmatūras izstrāde, kas ir piemērota tieši izmaiņu izstrādei. Veicot intervijas ar klientu, ĀB3 nepieciešamās izmaiņas tiek identificētas vairākās iterācijās, ko arī atbalsta Spējā programmatūras izstrādes metodoloģija. ĀB3 izstrādes projekta vadītājs uzver, ka katra projekta ietvaros metodoloģija tiek izvērtēta un var tikt mainīta, atkarībā no klienta prasībām (ir izmantots arī ūdenskrituma modelis). Darba autore uzskata, ka neatbilstošas metodoloģijas izvēles risks R-12 (skatīt 4.1.1. tabulu) tiek atbilstoši mazināts, jo metodoloģijas atbilstība projektam tiek regulāri pārskatīta.

Dave Henderson norāda: *„IS auditoram ir jāpārskata dokumentācija, lai pārliecināts par to, ka eksistē dokumentēta metodoloģija un ka piegādātais produkts ir izstrādāts, izmantojot šo metodoloģiju"* (2, 44. lpp). Darba autore uzskata, ka dokumentēta metodoloģija būtiski atvieglo projekta pārvaldības procesu, palīdzot sekot projekta gaitai, un palielina projekta izstrādes procesa kvalitāti. Intervijas laikā tika konstatēts, ka ĀB3 izstrādātājs izmanto projekta pārvaldības plānu, kurā tiek norādīta izstrādes procesa gaita un dokumentēta izvēlētā metodoloģija.

Rakstā „Secure Software Development – The Role of IT Audit" tiek norādīts, ka izstrādes procesā svarīga ir arī koda drošība: *„Programmatūras koda inspekcijas var panākt būtiskus uzlabojumus drošā programmatūras izstrādē, inspekcijas laikā kodu pārskata pieredzējuši programmētāji, kas jau ir pazīstami ar drošības jautājumiem"* (5, 51. lpp). Pēc darba autores domām, nekorekti izstrādāts kods var būtiski ietekmēt programmatūras ātrdarbību, tādēļ, lai mazinātu nekorekta un neoptimāli izstrādāta koda risku R-13 (skatīt 4.1.1. tabulu), darba autore iesaka veikt regulāras koda inspekcijas. Arī ĀB3 izstrādes laikā vecākā programmētāja pienākumos ietilpst koda inspekcijas, kas nodrošina augstāku drošības līmeni.

Lai mazinātu ļaunprātīgu testa datu izmantošanas risku R-14 (skatīt 4.1.1.tabulu), nepieciešams kontrolēt arī testa datu drošību. Ieteicams ir testēšanas laikā neizmantot reālus datus, arī C. Warren Axelrod uzskata: *„[..] Programmētāji un testētāji var darīt savu darbu neredzot personiskus vai sensitīvus datus"* (1, 58. lpp). Lai to panāktu, eksistē rīki, kas ģenerē datu tipam atbilstošus datus, kas protams ir neērti, jo tiem nav saprotami loģiski scenāriji. ĀB3 testēšanas laikā tiek izmantoti testēšanas laikā ģenerētie dati. Tā kā programmatūra apstrādā tādus datus, kam nepieciešama unikalitāte, piemēram, personas kods, ĀB3

izstrādātājs ir parūpējies par unikāla personas koda ģenerēšanu, šim nolūkam tika izstrādāta iekšējai lietošanai paredzēta programma.

Kā jau tika minēts 3. nodaļā, risku analīzes rezultātā iespējams identificēt potenciālos apdraudējumus, tātad pirms tiek veikta programmatūras drošības testēšana, būtu nepieciešams noteikt, kādus apdraudējumus testēt. ĀB3 projekta prasību definēšanas ciklā veic risku analīzi, kuras laikā tiek identificēti iespējamie apdraudējumi. Tomēr netiek veikta risku pārvaldība visā projekta gaitā, līdz ar to netiek identificēti riski, kas var rasties izstrādes laikā. Pamatojoties uz 3. nodaļā apskatītajiem riska pārvaldības nepieciešamības argumentiem, ĀB3 izstrādātājam tiek ieteikts veikt riska pārvaldību visā projekta laikā, visā programmatūras izstrādes dzīves ciklā, kā arī parūpēties par risku pārvaldībai nepieciešamo dokumentāciju, kas atvieglos pārvaldības procesu. Jāņem vērā to, ka riska pieņemšana var dot biznesa ieguvumu, tā vietā lai tā ignorēšanas dēļ saskartos ar lieliem zaudējumiem. Mukul Pareek rakstā „Living With Risk" paskaidro riska un atlīdzības saistību: *„Risks un atlīdzība ir pozitīvi saistīti – jo augstāks pieņemtais biznesa risks, jo augstāks biznesa sagaidāmais ieguvums"* (6).

4.1.5. Cilvēka radītie riski

Kā zināms bieži vien lietotājs neapzinās savas rīcības sekas, vai pat nav informēts par drošības pasākumiem, kurus būtu nepieciešams ievērot. Pastāv risks, ka programmatūra tiks nepareizi lietota vai nepareizi konfigurēta – riski R-15, R-16 (skatīt 4.1.1. tabulu), tādēļ nepieciešamas lietotāja drošības apmācības. CISA rokasgrāmatā teikts: *„Drošības efektivitāte vienmēr būs atkarīga no cilvēka. Drošība var būt efektīva tikai gadījumā, kad lietotājs zina, ko no viņa sagaida, un kādi ir viņa pienākumi"* (11, 217. lpp).

Drošības apmācībās vajadzētu iekļaut tādas daļas kā datora drošības pamatus, fiziskās drošības nepieciešamību un īstenošanu, atbildību un pienākumu definēšanu (10, 267. lpp). Lietotājam jābūt informētam arī par auditācijas pierakstu vākšanu, tas ir, ka lietotāja darbības tiek uzraudzītas, to var realizēt dažādos veidos: iekļaujot punktu līgumā, parādot paziņojumu lietotājam piekļuves laikā, iekļaujot auditācijas nepieciešamības un pielietojamības skaidrojumus lietotāja apmācībās.

Pastāv risks, ka klientu neinteresē fiziskās drošības ievērošana (risks R-18, skatīt 4.1.1. tabulu), piemēram, durvju un logu slēgšana, paroles glabāšana pieejamā vietā, programmatūra netiek aizvērta pēc darba beigšanas. Tā kā šī riska iestāšanās ir tikai un vienīgi atkarīga no lietotāja rīcības, tad darba autore secina, ka riska vienīgais mazināšanas veids ir lietotāja

informēšana par fiziskajiem drošības pasākumiem un brīdināšana par iespējamām sekām, kas var rasties gadījumā, ja šie pasākumi netiek veikti.

M. Bhatia grāmatā „Auditing in a Computerised Environment" definē konkrētus paziņojumus, kurus būtu jāparāda lietotājam, brīdī, kad notiek piekļūšana sistēmai, ieteicams rādīt paziņojumus par to, ka (10, 254. lpp):

- sistēma tiek aizsargāta,
- neautorizēta pieeja ir aizliegta,
- lietotāja darbības tiek uzraudzītas un kā pierādījumi var tikt nodoti tiesā,
- lietotājs var turpināt darbību, ja piekrīt visam iepriekš minētajam.

Protams, šos paziņojumus var rādīt arī jebkādā citā klientam ērtā veidā. Darba autore uzskata, ka lietotāju apmācības un programmatūras paziņojumi par lietotāja tiesībām un pienākumiem mazina lietotāja neinformētības risku R-17 (skatīt 4.1.1.tabulu).

ĀB3 ir izstrādāta lietotāju apmācības programma, kurā tiek iekļauta arī programmatūras drošības pasākumu apmācība. Dokumentā norādīts: *„Mācību procesa apguves laikā izmantojamas šādas metodes:*

- *Demonstrēšana – pasniedzējs demonstrē rādāmo vielu apmācāmajiem, apmācāmie redzēto atkārto;*
- *Praktiskā darbība – apmācāmie veic dažādus uzdevumus patstāvīgi, nepieciešamības gadījumā lūdzot pasniedzēju palīdzēt;*
- *Pārbaudes darbs – apmācāmie veic uzdevumus patstāvīgi, par to iegūstot pasniedzēja vērtējumu, kā rezultātā apmācāmais iegūst sistēmas Ārsta Birojs sertifikātu, kas apliecina viņa spēju strādāt ar sistēmu"* (15).

Intervijas laikā tika noskaidrots, ka apmācībās tiek īpaši uzsvērta paroles droša glabāšana un neatklāšana citām personām, kā arī lietotājs tiek informēts, ka visas veiktās darbības tiek auditētas.

Darba autore secina, ka tiek veiktas kvalitatīvas un atbilstošas programmatūras drošības apmācības, tomēr ĀB3 izstrādātājam būtu ieteicams, pieslēdzoties sistēmai, lietotāju informēt par to, ka tiek veikta darbību uzraudzība.

4.2. Kopsavilkums

Darba autore secina, ka ĀB3 izstrādātājs ir veicis drošības pasākumus, kas nepieciešami lai nodrošinātu programmatūras drošību, tomēr papildus drošībai ieteicams ieviest arī citus risinājumus. Standartos un ieteikumos minēto drošības risinājumu (skatīt 1. nodaļu) un ĀB3

ieviesto drošības risinājumu atbilstība ir apkopota tabulā (skatīt 4.2.1. tabulu), kā arī tiek sniegti darba autores ieteikumi programmatūras drošības uzlabošanai.

4.2.1. tabula

ĀB3 drošības risinājumu atbilstība vispārpieņemtiem standartiem un ieteikumiem

Ieteiktais drošības risinājums	**ĀB3 drošības risinājums**
Izstrādāt drošības pārvaldības plānu, vadoties pēc tā, aizsargāt programmatūras drošību visā izstrādes dzīves ciklā	✓ Tiek veiktas drošības pārbaudes visā programmatūras izstrādes dzīves ciklā **X** Netiek izstrādāts dokumentēts drošības pārvaldības plāns + Ieteicams dokumentēt drošības pārvaldības plānu
Veikt risku pārvaldību visā programmatūras izstrādes dzīves ciklā, lai noteiktu programmatūras apdraudējumus, un piemērot līdzekļus, lai tos mazinātu	✓ Programmatūras prasību definēšanas fāzē tiek veikta risku analīze + Ieteicams veikt risku analīzi iterācijās – visa programmatūras izstrādes dzīves cikla laikā
Izmantot autorizāciju vai autentifikāciju programmatūras piekļuvei, lai nodrošinātu informācijas konfidencialitāti	✓ Tiek izmantota autorizācijas metode, lietotājs norāda lietotāja vārdu un paroli
Izstrādāt drošas paroles politiku (droša parole, paroles lietošanas ierobežojumi, konta slēgšana pēc neveiksmīgiem mēģinājumiem)	✓ Drošas paroles politika ir izstrādāta, iekļaujot drošas paroles nepieciešamību, lietošanas ierobežojumus **X** Drošas paroles politika var tikt izslēgta ar iestatījuma palīdzību + Ieteicams ieviest obligātu drošas paroles politikas izmantošanu + Ieteicams slēgt kontu pēc noteikta skaita neveiksmīgā piekļuves mēģinājuma
Izmantot lomu un tiesību mehānismu, lai nodrošinātu informācijas konfidencialitāti	✓ Izstrādāts lomu un tiesību mehānisms, eksistē loma „administrators", kurai ir pilnas tiesības, kā arī tiesības izveidot jaunu lomu. Jaunajai lomai sākotnēji nav nekādu tiesību.
Veikt lomu un tām piešķirto tiesību, kā arī lomām piesaistīto lietotāju regulāru inspekciju	✓ Izstrādāta ĀB3 atskaite, kurā tiek attēlotas lomas un tām piešķirtās tiesības, piesaistītie lietotāji
Izstrādāt datu validācijas funkcijas	✓ Izstrādātas funkcijas, kas veic datu validāciju: atbilstību datu tipam, maksimālajam garumam, ieviesti klasifikatori, kā arī loģiskās validācijas – datumu pārklāšanās intervāli, informācijas formu kontrole
Izstrādāt lietotāja darbību auditāciju, lai nodrošinātu informācijas integritāti un mazinātu datu zaudēšanas iespējamību, veikt auditācijas pierakstu analīzi	✓ Tiek veikta divu līmeņu lietotāja darbību auditācija: auditē visas lietotāja darbības, auditē informācijas pievienošanas, labošanas un dzēšanas darbības **X** Auditācija var tikt izslēgta ar iestatījuma palīdzību + Ieteicams ieviest obligātu lietotāja datu apstrādes darbību

Ieteiktais drošības risinājums	ĀB3 drošības risinājums
	auditāciju
Veikt personāla programmatūras lietošanas un drošības apmācības	✓ Tiek veiktas lietotāju apmācības: programmatūras lietošana, konfigurēšana, drošības apmācība
Izstrādāt biznesa nepārtrauktības plānu, lai nodrošinātu informācijas pieejamību, ieteicams iekļaut kļūdu un izmaiņu pieprasījumu apstrādes plānus, rezerves kopiju veidošanas politiku	✓ Ir izstrādāts dokumentēts biznesa nepārtrauktības plāns ✓ Ir izstrādāts kļūdu apstrādes mehānisms ✓ Ir izstrādāts izmaiņu pieprasījumu apstrādes mehānisms ✓ Ir izstrādāta rezerves kopiju veidošanas politika un tā tiek dokumentēta
Veikt programmatūras testēšanu, lai mazinātu kļūdu rašanās risku	✓ Tiek veikta programmatūras testēšana, izmantojot melnās kastes metodi ✓ Tiek izmantota kļūdu apstrādes sistēma Mantis ✓ Tiek izmantoti testa piemēri ✓ Plānots ieviest automātisko testēšanu
Veikt programmatūras koda inspekcijas	✓ Vecākais programmētājs veic regulāras koda inspekcijas, vērtējot koda drošību un optimālumu
Testēšanas procesā neizmantot reālus personu datus, lai nodrošinātu informācijas konfidencialitāti	✓ Testēšanas procesā datus ģenerē testētājs ✓ Unikālu personas kodu ģenerēšanai tiek izmantots organizācijā izstrādāts rīks
Izstrādāt un dokumentēt projekta pārvaldības metodoloģiju, lai mazinātu projekta izgāšanās risku	✓ Projekts tiek pārvaldīts pēc Spējās metodoloģijas vai Ūdenskrituma metodoloģijas, atkarībā no klienta prasībām ✓ Metodoloģijas izvēle un atbilstība tiek regulāri analizēta ✓ Metodoloģija tiek dokumentēta projekta pārvaldības plānā
Izstrādāt programmatūras dokumentāciju (programmatūras pārvaldības plānu, prasību specifikāciju, lietotāja rokasgrāmatu)	✓ Tiek izstrādāts Programmatūras projektējuma apraksts veicamajiem papildinājumiem ✓ Visas prasības tiek reģistrētas sistēmā Mantis ✓ Tiek izstrādāta Lietotāja rokasgrāmata **X** Programmatūras prasību specifikācija tiek izstrādāta, ja klients to pieprasa + Ieteicams izstrādāt Programmatūras prasību specifikāciju, neatkarīgi no klienta prasībām

✓ **-** ĀB3 tiek izmantotas drošības kontroles

X – ĀB3 drošības kontroles nav pietiekamas

+ - Darba autores ieteikums

NOBEIGUMS

Programmatūra „Ārsta Birojs 3" ir Latvijā izplatīta veselības aprūpes iestādes programmatūra, kura apstrādā pacienta veselības datus. Tā kā veselības stāvokļa dati ir klasificējami kā sensitīvi, nepieciešams ieviest stingras drošības kontroles.

Šajā darbā tika apskatīti 9 informācijas un programmatūras drošības standarti un ieteikumi, to skaitā ISO/IEC 17799:2005, kas ir pasaulē pieņemts informācijas drošības standarts. Tika pētītas 4 programmatūras pārbaudes vadlīnijas, starp tām arī CISA rokasgrāmata, kas ir pasaulē plaši pielietota programmatūras audita rokasgrāmata.

Autore uzskata, ka šī darba mērķis apkopot programmatūras drošības standartus un novērtēt programmatūras „Ārsta Birojs 3" drošības atbilstību šiem standartiem ir sasniegts. Darba izstrādes rezultātā ir gūti sekojoši secinājumi:

- Pacienta datu drošība ir aktuāla problēma, jo pacienta dati ir sensitīvi un to apstrādei jābūt drošai – konfidenciālai, korektai un pieejamai;
- Eksistē vairāki programmatūras un informācijas drošības pārvaldības standarti un vadlīnijas, kas sniedz aptverošu informāciju par drošības pārvaldības procesu, rīkiem un tehnikām, nepieciešamajiem drošības pasākumiem un to dokumentēšanu;
- Ieteikumi par programmatūras pārbaudes veikšanu kopumā ietver vienojošus veicamos soļus: pārbaudes plānošana, informācijas iegūšana, informācijas analizēšana un pārbaudes veikšana, visbeidzot pārbaudes rezultātu sniegšana;
- Programmatūras „Ārsta Birojs 3" drošības risinājumi atbilsts vispārpieņemtajiem standartiem un ieteikumiem, tomēr drošības pastiprināšanai ieteicams ieviest papildus drošības risinājumus.

Šī darba izstrādes gaitā iegūtie rezultāti tiks nodoti programmatūras „Ārsta Birojs 3" izstrādātājam. Pēc autores domām, sniegtie ieteikumi var palīdzēt pastiprināt ne tikai pētītās programmatūras drošību, bet arī ir derīgi jebkuras programmatūras drošības uzlabošanai.

IZMANTOTĀ LITERATŪRA

Žurnāli

1. **Axelrod, C. W.** Achieving Privacy Through Security Measures. *Information Systems Control Journal*, 2007, vol. 2, p. 56-60.
2. **Henderson, D.** Issues With Auditing the Systems Development Process. *Information Systems Control Journal*, 2008, vol. 6, p. 42-45.
3. **McGraw, G.** Software Security. *IEEE Security and Privacy*, 2004, vol. 2, p. 80-83.
4. **Moynihan, J.** Managing the Insider Threat: Data Surveillance. *Information Systems Control Journal*, 2008, vol. 1, p. 38-39.
5. **Ozem, A., Ciaramitaro, B. L.** Secure Software Development – The Role of IT Audit. *Information Systems Control Journal*, 2008, vol. 4, p. 49-52.
6. **Pareek, M.** Living With Risk. *Information Systems Control Journal*, 2006, vol. 6, p. 35-38.
7. **Sathiyamurthy, S.** Is the IT Risk Worth a Control? *Information Systems Control Journal*, 2008, vol. 6, p. 35-38.
8. **Singleton, T. W.** What Every IT Auditor Should Know About Access Controls. *Information Systems Control Journal*, 2008, vol. 4, p. 11-14.
9. **Traverse, C.** Implementing, Automating and Validating Controls for Privileged Users in Healthcare Organizations. *Information Systems Control Journal*, 2008, vol. 6, p. 30-32.

Grāmatas

10. **Bhatia, M.** *Auditing in a Computerized Environment*. New Delhi: Tata McGraw – Hill Publishing Company Limited, 2002. 654 p.
11. *CISA Review Manual.* IL: Information Systems Audit and Control Association, 2003. 474 p.

Raksti grāmatās

12. **Kokolakis, S., Gritzalis, D., Katsikas, S.** Draft Standard for High Level Security Policies for Healthcare Establishments. **In:** *Security Standards for Healthcare Information Systems*. The Netherlands Publ., E.B. Barber et. al. (Eds.), 2002, p. 23-47.

13. **Kokolakis, S., Gritzalis, D., Katsikas, S., Ottes, F.** Overview on Security Standards for Healthcare Information Systems. **In:** *Security Standards for Healthcare Information Systems*. The Netherlands Publ., E.B. Barber et. al. (Eds.), 2002, p. 13-22.

Dokumenti

14. *Personas datu apstrādes sistēmu audita rokasgrāmata 1.versija.* Rīga: Datu valsts inspekcija, 2004. 193 lpp.
15. **Skladņeva, L.** *Lietotāju un administratoru mācību plāns, materiāli un metodika, 2.versija.* Rīga, 2009. 29 lpp.
16. **IEEE Std 1028-1997**, *IEEE Standard for Software Reviews*, 1997. 38 p.
17. **ISO/IEC FDIS 17799:2005(E)**, *Information Technology – Security techniques – Code of practice for information security management*, 2005. 115 p.

Elektroniskie informācijas avoti

18. **Borzovs, J.** *CobIT* [tiešsaiste]. Rīga: LU, 2009 – [atsauce 05.05.2010.]. Pieejams: http://estudijas.lu.lv/file.php/34/Materiali/CobiT.htm
19. *Fizisko personu datu aizsardzības likums.* 01.07.2009, Rīga: Saeima [atsauce 10.05.2010]. Pieejams: http://www.likumi.lv
20. *Lielā terminu vārdnīca*. [tiešsaiste] – [atsauce 10.05.2010.]. Pieejams: http://www.termini.lv
21. **Līcis, K.** *Riska vadības stratēģijas uzņēmumā (1.daļa).* [tiešsaiste] – [atsauce 01.05.2010.]. Pieejams: http://dll.lv/riska-vadibas-strategijas-uznemuma-1-dala
22. *Par ISACA Latvijas nodaļu.* [tiešsaiste] – [atsauce 10.05.2010.]. Pieejams: http://isaca.biti.lv/content/par-isaca-latvijas-noda-u
23. *Personas datu aizsardzības obligātās tehniskās un organizatoriskās prasības.* MK noteikumi Nr. 40, 01.09.2007, Rīga : Ministru kabinets [atsauce 10.05.2010]. Pieejams: http://www.likumi.lv
24. *CobIT 4.1.* The IT Governance Institute, 2007 – [atsauce 10.05.2010.]. Pieejams: http://www.isaca.org/Template.cfm?Section=COBIT6&Template=/TaggedPage/TaggedPageDisplay.cfm&TPLID=55&ContentID=7981
25. *Data Integrity* **In:** Wikipedia, the Free Encyclopedia. [tiešsaiste] – [atsauce 05.05.2010.]. Pieejams: http://en.wikipedia.org/wiki/Data_integrity
26. *Information security* **In:** Wikipedia, the Free Encyclopedia. [tiešsaiste] – [atsauce 02.05.2010.]. Pieejams: http://en.wikipedia.org/wiki/Information_security

27. *Information Technology Infrastructure Library* **In:** Wikipedia, the Free Encyclopedia. [tiešsaiste] – [atsauce 03.05.2010.]. Pieejams: http://en.wikipedia.org/wiki/Information_Technology_Infrastructure_Library
28. *ISACA Overview and History*. [tiešsaiste] – [atsauce 10.05.2010.]. Pieejams: http://www.isaca.org/template.cfm?section=Overview_and_History
29. *IT Audit and Assurance Guidelines.* [tiešsaiste] – [atsauce 10.05.2010.]. Pieejams: http://www.isaca.org/Template.cfm?Section=Standards&CONTENTID=55948&TEMPLATE=/ContentManagement/ContentDisplay.cfm
30. *IT Audit and Assurance Tools and Techniques.* [tiešsaiste] – [atsauce 10.05.2010.]. Pieejams: http://www.isaca.org/Template.cfm?Section=Standards&CONTENTID=49806&TEMPLATE=/ContentManagement/ContentDisplay.cfm
31. *ITIL security management* **In:** Wikipedia, the Free Encyclopedia. [tiešsaiste] – [atsauce 03.05.2010.]. Pieejams: http://en.wikipedia.org/wiki/ITIL_Security_Management
32. **Herzog, P.** *Open-Source Security Testing Methodology Manual 2.2.* ISECOM [tiešsaiste]. 2006 – [atsauce 29.04.2010.]. Pieejams: http://www.isecom.org/osstmm/
33. **Higuera, R. P., Haimes, Y. Y.** *Software Risk Management*. Pittsburgh: Carnegie Mellon University [tiešsaiste]. 1996 – [atsauce 29.04.2010.]. Pieejams: http://www.sei.cmu.edu/reports/96tr012.pdf
34. **Meritt, J. W.** *A Method for Quantitative Risk Analysis*. [tiešsaiste] – [atsauce 10.05.2010.]. Pieejams: http://csrc.nist.gov/nissc/1999/proceeding/papers/p28.pdf
35. **North, D.W.** *Limitations, definitions, principles and methods of risk analysis.* [tiešsaiste] – [atsauce 25.04.2010.]. Pieejams: http://www.northworks.net/limitations.pdf
36. *Software security assurance*. Information Assurance Technology Analysis Center [tiešsaiste]. 2007 – [atsauce 29.04.2010.]. Pieejams: http://iac.dtic.mil/iatac/download/security.pdf
37. **Stevenson, G.** *Managing Information Privacy & Security in Healthcare* **In:** Healthcare Information and Management Systems Society [tiešsaiste]. 2007 – [atsauce – 30.04.2010.]. Pieejams: http://www.himss.org/content/files/CPRIToolkit/version6/v6%20pdf/D03_Security_Primer.pdf
38. **Rasche, T.** *Risk Analysis Methods – a Brief Review*. [tiešsaiste] – [atsauce 10.05.2010.]. Pieejams: http://www.mishc.uq.edu.au/Publications/Risk_Analysis_Methods_a_Brief_Review.pdf

39. *The Standard of Good Practice for Information Security*. Information Security Forum [tiešsaiste]. 2007 – [atsauce – 15.05.2010.]. Pieejams: http://www.netbotz.com/library/Info_Security_Forum_Standard_Good_Practices.pdf
40. *What is Prototyping?* [tiešsaiste] – [atsauce 13.05.2010.]. Pieejams: http://www.umsl.edu/~sauterv/analysis/prototyping/proto.html

PIELIKUMI

1. pielikums

Programmatūras „Ārsta Birojs 3" funkcionalitāte

1. Pacientu datu bāze

Tiek uzkrāta pacientu/slimnieku pases elektroniska informācija.

Pases informācija par konkrēto personu ir jāievada vienreiz. Pēc tam to izmanto visās nepieciešamajās vietās:

- Reģistratūrai/uzņemšanai sagatavojot ambulatorās kartes/slimības vēstures vākus,
- Statistikas nodaļai ievadot "No stacionāra izrakstītā "mirušā" slimnieka karti" vai "Ambulatorā pacienta talonu" datu bāzē.

2. "Ambulatorā pacienta talonu" datu bāze

- Ērta talonu ievade VSMTA noteiktajā kārtībā;
- Tiek nodrošināta ikmēneša talonu datu eksports uz slimokasi;
- Pārskatu veidošanas iespējas – pa ārstiem, specialitātēm, slimokasēm, diagnozēm, pacientiem, datumiem, manipulācijām, epizodēm, vizītēm u.c.

3. "No stacionāra izrakstīto (mirušo) slimnieku karšu" datu bāze

- Ērta karšu ievade VSMTA noteiktajā kārtībā;
- Stacionāra kartes izdruka;
- Stacionāra kartes arhīvs;
- Datu ievadīšana uzņemšanā;
- Tiek nodrošināts regulārs datu eksports uz slimokasi;
- Pārskatu veidošanas iespējas – pa ārstiem, specialitātēm, slimokasēm, diagnozēm, pacientiem, datumiem, manipulācijām, nodaļām, profiliem u.c.;
- Izmaksu aprēķins pēc gultu dienām un manipulācijām;
- Finanšu atskaites un statistikas pārskati;
- Atskaites apdrošinātājiem.

4. Ārstu datu bāze

Visu iestādē strādājošo mediķu informācija.

Ārstu datus izmanto visās nepieciešamajās vietās:

- Reģistratūrai/uzņemšanai sagatavojot ambulatorās kartes/slimības vēstures vākus;

- Statistikas nodaļai ievadot "No stacionāra izrakstītā "mirušā" slimnieka karti" vai "Ambulatorā pacienta talonu" datu bāzē.

5. Kalendārs

- Reģistratūras pieraksta kalendāra uzturēšana;
- Ārsta skats;
- Reģistratūras skats;
- Brīvās vietas meklēšana;
- Vizīšu un epizožu automātiska veidošana reģistrējot apmeklējumu kalendārā;
- Neatliekamā medicīniskā palīdzība un Traumpunkts;
- NMP karšu ievade;
- NMP karšu arhīvs un statistika;
- Tiek nodrošināts datu eksports uz VIS.

6. Traumpunkta apmeklējuma reģistrācija

- Talona veidošana;
- Nosūtījuma uz RIS veidošana.

7. Klīniskās nodaļas

- Uzņemšana nodaļā;
- Pārvietošana no/uz nodaļu;
- Izrakstīšana no slimnīcas;
- Manipulāciju un operāciju fiksēšana;
- Nosūtījumi uz izmeklējumiem;
- Epikrīzes veidošana, drukāšana;
- U.C

8. Diktofona centrs

- Dokumentu sagatavošana no veidnēm;
- Informācijas ielasīšana dokumentā no datu bāzes;
- Dokumentu rediģēšana (papildināšana ar ārsta diktēto informāciju) ar Microsoft Word, drukāšana un saglabāšana.

9. Kase (*)

- Maksas pakalpojumu definēšana;
- Apdrošinātāju kompensācijas apjomu definēšana;

- Čeku sagatavju veidošana;
- Čeku veidošana un drukāšana;
- Atskaites par sniegtajiem maksas pakalpojumiem.

(*) Elektronisko Kases Aprātu modeļiem CHD 3010T; CHD 5010T; CHD 5510; CHD 3550.

10. Noliktava

Papildmodulis Noliktava nodrošina Ārsta biroja sadarbību ar veselības aprūpes iestādes medikamentu un medicīnas preču noliktavas programmu:

- Medikamentu un medicīnas preču uzskaite par katru pacientu;
- Medikamentu un medicīnas preču uzskaite par katru slimnīcas nodaļu;
- Atskaites.

11. Laboratorijas IS

- Nosūtījuma ievade ārsta kabinetā;
- Nosūtījuma saņemšana laboratorijā;
- Laboratorijas žurnāls;
- Analīžu rezultātu ievade un piesaiste konkrētiem pacientiem un saslimšanas gadījumiem.

12. Radioloģijas IS

- Nosūtījuma ievade ārsta kabinetā;
- Nosūtījuma saņemšana radioloģijas nodaļā;
- Datu nosūtīšana uz radiologa asistenta specializēto darba vietu;
- Rezultātu ievade;
- Dozu/filmu u.c. reģistrs.

2. pielikums

Intervija ar izstrādātāju

Intervijas laiks: 24.03.2010.

Intervijā piedalās: Programmatūras „Ārsta Birojs 3" projektu vadītājs, vecākais programmētājs un darba autore (intervētāja).

Intervijas mērķis: Iegūt informāciju par programmatūras „Ārsta Birojs 3" funkcionalitāti un izmantotajiem drošības risinājumiem, lai šī darba ietvaros novērtētu drošības risinājumu atbilstību vispārpieņemtiem programmatūras drošības standartiem.

Intervijas laikā apspriestie jautājumi:

1. **Īsi pastāstiet par programmatūru „Ārsta Birojs 3", kā tā ir pielietojama?**

   Programmatūra „Ārsta Birojs 3" ir veselības aprūpes iestādes programmatūra, kas nodrošina pacienta veselības datu apstrādi, kā arī uztur ārstu un laboratorijas darbību datubāzes. Programmatūra ir ieviesta daudzās veselības aprūpes iestādēs visā Latvijā, nākotnē plānots programmatūru piedāvāt arī citās valstīs.

2. **Kādu informāciju un datus apstrādā programmatūra „Ārsta Birojs 3"?**

   Programmatūra „Ārsta Birojs 3" apstrādā sekojošus datus:

   1) Pacienta datus,
   2) Ārsta datus,
   3) Pacienta ambulatorās ārstēšanas datus,
   4) Pacienta stacionārās ārstēšanas datus,
   5) Pacienta nosūtījumus uz izmeklēšanu,
   6) Veikto izmeklējumu datus.

3. **Vai eksistē dokumentēts programmatūras drošības pārvaldības plāns?**

   Nē, bet drošība tiek kontrolēta un pārskatīta regulāri, atbilstoši organizācijas pieņemtajiem drošības pasākumiem.

4. **Kā tiek nodrošināta programmatūras ĀB3 pieejamība (rezerves kopiju veidošana datiem, biznesa nepārtrauktības plāns)?**

Ir izstrādāts biznesa nepārtrauktības plāns, kurā ietverts kļūdu apstrādes mehānisms, izmaiņu pieprasījumu apstrādes mehānisms, kā arī rezerves kopiju veidošanas politika. Rezerves kopijas tiek veidotas pa etapiem:

1) Katru dienu tiek veidota datu rezerves kopija un novietota turpat uz servera,
2) Katru nedēļu tiek veidota datu rezerves kopija un novietota attālināti,
3) Katru mēnesi tiek atjaunots datu rezerves kopiju arhīvs, kas glabājas attālināti.

5. Vai datu glabāšanā, pārvietošanā tiek izmatota kāda datu šifrēšanas metode?

Datu šifrēšanas metode tiek pielietota tikai parolēm.

6. Kādas autorizācijas vai autentifikācijas metodes tiek izmantotas ĀB3?

a. Vai eksistē lietotāja vārda ierobežojumi – garums, simboli,

Nē, lietotāja vārda ierobežojumi nav.

b. Kā tiek šifrēta parole,

Parole tiek šifrēta izmantojot SHA-1 algoritmu.

c. Atzīmējiet, paskaidrojiet vai un kā tiek izmantots:

Metode	**Paskaidrojums**	**Tiek izmantots ĀB3**
Automātiskais laika ierobežojums parolei	Kāds noteikts periods, pēc kura parole ir jāmaina	ĀB3 piedāvā iestatījumus, kuros klientam iespējams norādīt paroles derīguma periodu
Stipra parole	8 vai vairāk zīmes, iekļauti lielie burti, cipari, speciālās zīmes	ĀB3 piedāvā iestatījumus, kuros klientam iespējams norādīt paroles minimālo garumu, ciparu obligātumu parolē
Paroles aizsardzība	Neizmantot piezīmju lapiņas tuvu datoram, neatbildēt uz nezināmas, nekompetentas personas lūgumu pēc paroles, slēgt piekļuvi pēc 3. neveiksmīgā autorizācijas mēģinājuma	Šāda drošības ievērošana ir klienta kompetencē. Piekļuve pēc 3. neveiksmīgā mēģinājuma netiek slēgta
Redzamības ierobežojumi	Rādīt tikai lietotājam nepieciešamos skatus, datus	ĀB3 nodrošina administrēšanas iespējas pa lomām, kur iespējams norādīt skatus, kurus attēlot lomai
Ierobežots administratoru skaits	Ierobežots administratoru skaits un droša piekļuve, neizmantot iebūvētos administratoru kontus	Administratoru skaitu kontrolē klients. Iebūvētie administratoru konti netiek izmantoti
Paroles	Efektīva paroles slēgšana pēc lietotāja	Lietotāja kontu slēdz administrators,

Metode	Paskaidrojums	Tiek izmantots ĀB3
iznīcināšana	darba līguma beigām	izmantojot ĀB3 izstrādātu funkciju

7. Kāda metode tiek izmantota lietotāju lomu pārvaldībā (izveidojot lietotāju, tam tiek piešķirtas visas vai neviena tiesība)? Kādas lomas pastāv ĀB3? Kam ir tiesības lietotājam definēt lomas un tiesības?

Eksistē administratora loma, kurai tiek piešķirtas visas tiesības, administrators var veidot jaunas lomas, kurām sākotnēji netiek piešķirta neviena tiesība.

8. Cik sadrumstalota ir datu pieeja? Kā tiek ierobežotas pacienta datu pievienošanas, skatīšanas, labošanas un dzēšanas tiesības?

Katrai lomai tiek noteikta tiesības pievienot, skatīt, labot un dzēst informācijas apgabalu, kas ir ĀB3 skats.

9. Vai un kā tiek ierobežots ĀB3 administratoru skaits?

Administratoru skaitu nosaka klients.

10. Vai tiek izmantots kāds rīks, kas ļauj ģenerēt lietotāju tiesību atskaiti, vai šāda veida atskaites sniedz ĀB3?

ĀB3 ir izstrādāta atskaite, kas sniedz informāciju par eksistējošām lomām, to tiesībām un piesaistītajiem lietotājiem.

11. Vai ĀB3 tiek veikti audita pieraksti? Ja jā, tad kādiem datiem tie tiek veikti, kādām darbībām? Kā lietotājam tiek paziņots par to, ka audita pieraksti tiek veikti (piemēram, punkts līgumā, paziņojums pie ielogošanās, paziņojums pie darbības veikšanas)?

Jā, ĀB3 tiek veikta divu līmeņu auditācija. Pirmkārt, auditācijas pieraksti tiek veidoti par katru lietotāja veikto darbību, reģistrējot lietotāja identifikatoru, veikto darbību, laiku, kad darbība veikta. Otrkārt, tiek veidots auditācijas pieraksts par lietotāja veiktajām informācijas labošanas darbībām, jeb informācijas pievienošana, labošana un dzēšana. Dzēšanas gadījumā tiek saglabāti arī visi dzēstie dati. Papildus tiek saglabāts pacienta identifikators un labotās informācijas identifikators.

Lietotājam apmācības ietvaros tiek sniegta informācija par to, ka lietotāja darbības tiek kontrolētas.

12. **Vai tiek izmantots kāds rīks, kas analizē vai ģenerē atskaites par lietotāju veiktajām darbībām? Vai ir noteikti kādi brīdinājumi vai sodi lietotājam, kuri izmanto pacienta datus ārpus darba vajadzībām?**

Nē, netiek izmantots rīks analizēšanai. Klients nepieciešamības gadījumā sazinās ar izstrādātāju un auditācijas pieraksti pēc klienta prasības tiek pārskatīti. Par lietotāju brīdināšanu un sodīšanu rūpējas klients.

13. **Vai tiek veikta ĀB3 lietotāju drošības apmācība? Ja jā, tad kāds ir apmācības saturs?**

Jā, tiek veikta programmatūras lietošanas apmācība, kuras ietvaros tiek pastāstīts arī par veicamajiem drošības pasākumiem: fizisko drošību, paroles drošību, datu konfidencialitāti.

14. **Kādas datu validācijas metodes tiek izmantotas (piemēram, atbilstība datu tipam, simbolu ierobežojums, loģiski ierobežojumi, citas)?**

Tiek izmantotas gan šīs, gan arī loģiskās datu validācijas, piemēram, personas koda unikalitāte un forma, datumu intervālu nepārklāšanās u.c.

15. **Vai tiek izmantoti brīdinājuma paziņojumi lietotājam pirms datu dzēšanas un labošanas?**

Jā, tiek dots brīdinājums pirms datu dzēšanas. Papildus tiek izdots paziņojums, ja lietotājs mēģina aizvērt formu, kuras dati nav saglabāti.

16. **Vai ĀB3 ir iebūvēta kļūdu apstrāde un kļūdu paziņošana izstrādātājam? Cik ātri iespējams reaģēt kļūdas saņemšanas gadījumā? Kādu informāciju ietver kļūdas paziņojums lietotājam? Kādu informāciju ietver kļūdas paziņojums izstrādātājam?**

Jā, eksistē datubāze, kur tiek reģistrētas visas kļūdas. Uz kļūdas rašanos iespējams reaģēt 5 minūšu laikā, tomēr tas atkarīgs no klienta, jo par kļūdas rašanos informē klients. Lietotājam tiek parādīts saprotams un īss paziņojums, kā arī kļūdas identifikators.

Izstrādātājam redzams pilns kļūdas paziņojums, kurā redzama arī kļūdas rašanās apsekošana, lietotājs, kuram kļūda ir radusies un laiks.

17. **Vai eksistē drošības speciālisti vai drošības nodaļa ĀB3 izstrādātāja uzņēmumā? Vai tiek veikti drošības auditi?**

Nē, drošības nodaļa neeksistē, par drošības pārvaldību rūpējas izstrādes nodaļas vadītājs un projekta vadītājs. Izstrādes laikā tiek veiktas drošības pārbaudes.

18. **Definējiet ĀB3 testēšanas procesu. Kāda testēšanas metode tiek izmantota? Vai izstrādātāja uzņēmumā ir testēšanas nodaļa? Kā testēšana tiek dokumentēta, vai tiek izmantoti kādi testēšanas standarti?**

Eksistē testēšanas nodaļa, kura testē programmatūru, izmantojot melnās kastes testēšanas metodi. Testēšanas rezultāti tiek reģistrēti un pārvaldīti bezmaksas kļūdu apsekošanas sistēmā Mantis. Nākotnē ir plānots ieviest automātisko testēšanu, šim mērķim ir iegādāts speciāls rīks.

19. **Vai testēšanas procesā tiek izmantoti reāli pacienta dati? Vai tiek izmantots kāds rīks, kurš ģenerē testa datus?**

Reāli dati netiek izmantoti, testēšanas datus ievada testētājs. Netiek izmantots rīks testa datu ģenerēšanai.

20. **Kāda metodoloģija tiek izmantota ĀB3 izstrādē? Vai metodoloģija tiek dokumentēta? Kādi ir iemesli metodoloģijas izvēlē? Vai izstrādes uzņēmumā ir kāds darbinieks, kurš pārrauga izstrādes procesa norisi atbilstoši izvēlētajai metodoloģijai?**

Jāpiebilst, ka ĀB3 tiek piedāvāts kā produkts, tādēļ projekta gaitā parasti norit produkta ieviešana un papildinājumu izstrāde. Projekta pārvaldībai pārsvarā tiek izmantota spējā metode, jo izstrādātāja sadarbība ar klientu notiek interviju veidā vairākās iterācijās. Tomēr atkarībā no klienta prasībām ir izmantota arī ūdenskrituma metode. Metode netiek dokumentēta. Tomēr metodoloģijas atbilstība tiek regulāri analizēta.

21. Kāda programmatūras dokumentācija ir izstrādāta ĀB3 (piemēram, PPS, PPA, Lietotāja dokumentācija)?

Programmatūras papildinājumiem tiek izstrādāti projektējuma apraksti, tiek izstrādātās lietotāju rokasgrāmatas. Programmatūras prasību specifikācijas funkcijas veic sistēma Mantis, kurā tiek reģistrētas un apsekotas visas klienta prasības.

22. Kāda drošības kontroles metodoloģija tiek izmantota (piemēram, pirms ieviešanas drošības prasību definēšana, paralēlā kontrole, pēcizstrādes analīze)?

Drošība tiek kontrolēta visā projekta gaitā.

23. Vai tiek veiktas koda inspekcijas?

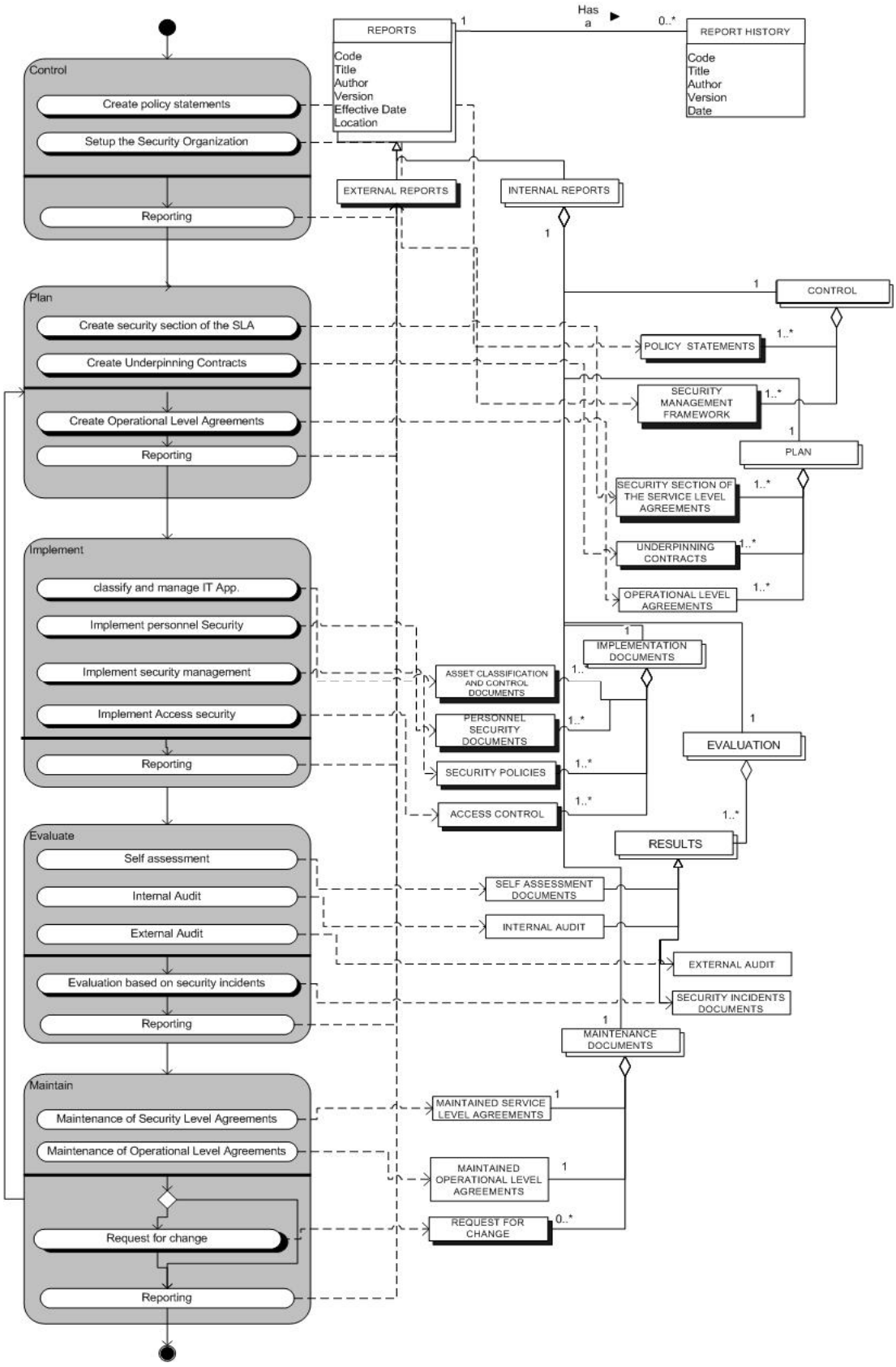
Jā, vecākais programmētājs regulāri veic koda inspekcijas.

24. Vai projekta ietvaros tiek veikta risku analīze? Ja jā, kādi rīki tiek izmantoti risku analīzei, pārvaldībai?

Projekta sākumā tiek veikta risku analīze un pieņemti lēmumi par pretsoļiem, kas veicami, lai riskus mazinātu. Risku pārvaldībā netiek izmantoti rīki.

3. pielikums

ITIL drošības pārvaldības procesa datu modelis



Bakalaura darbs „Programmatūras ‘Ārsta Birojs 3’ drošības risinājumi” izstrādāts LU Datorikas fakultātē.

Ar savu parakstu apliecinu, ka pētījums veikts patstāvīgi, izmantoti tikai tajā norādītie informācijas avoti un iesniegtā darba elektroniskā kopija atbilst izdrukai.

Autors: Dārta Krampe _____

Rekomendēju darbu aizstāvēšanai
Vadītāja: Dr.dat. Darja Šmite _____

Recenzents: _____

Darbs iesniegts Datorikas fakultātē _____

Metodiķe: _____

Darbs aizstāvēts bakalaura gala pārbaudījuma komisijas sēdē
_____ prot. Nr. _____ , vērtējums _____

Komisijas sekretāre: _____