Tallinna Ülikool

Digitehnoloogiate instituut

Teadlikult ebaturvaliselt kirjutatud veebirakenduse valimine õppetöös kasutamiseks

Bakalaureusetöö

Autor Hans Metsoja

Juhendaja Inga Petuhhov

Autor:....., 2017

Juhendaja:..... ....., 2017

Instituudi direktor:....., 2017

Autori deklaratsioon

Deklareerin, et käesolev bakalaureusetöö on minu töö tulemus ja seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on viidatud.

……	……
(kuupäev)	(autor)

Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks

1. Mina Hans Metsoja (sünnikuupäev: 14. november 1990) annan Tallinna Ülikoolile tasuta loa (lihtlitsentsi) enda loodud teose „Teadlikult ebaturvaliselt kirjutatud veebirakenduse valimine õppetöös kasutamiseks“ mille juhendaja on Inga Petuhhov säilitamiseks ja üldsusele kättesaadavaks tegemiseks Tallinna Ülikooli Akadeemilise Raamatukogu repositooriumis.

2. olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile.

3. kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest tulenevaid õigusi.

Tallinnas ……..

Sisukord

Sissejuhatus

Uurimus on ajendatud tähelepanekust, et üha rohkem rakendusi ehitatakse veebitehnoloogiatele ning veebipõhistele raamistikele. Samuti on veebiarendus ka üks valdkondi, kuhu pärast ülikooli kõige tõenäolisemalt tööle siirdutakse.

Lisaks tasub mainida, et hiljuti ilmunud uurimustöö alusel (Unruh, et al., 2017) on populaarsemad internetist leitavad õppematerjalid vigased ning sisaldavad turvanõrkusi. Lisaks ei mainita juhendites ära, et antud näitekood ei sobi kasutamiseks tarbekeskkonnas ja ei sisalda turva-kontrolle.

Autor julgeb aga eeldada, et nii koolitöid tehes kui ka hiljem esimeses töökohas kasutavad noored algajad kindlasti interneti õpetuste juhiseid.

Tallinna Ülikooli informaatika õppekavas veebirakenduste arendamise kursus küll käsitleb elementaarseid veebiturvalisuse nõudeid, kuid valdkond on avar ja keeruline. Tallinna Ülikoolis loetav sissejuhatav infoturbekursus on väga ülevaatlik, ent ei jõua igal teemal pikemalt peatuda. Autor usub, et veebi turvalisusele oleks õppeprogrammis vajalik natuke põhjalikumalt tähelepanu pöörata ning täiendada kas veebirakenduste arendamise kursust, või luua eraldi valikaine mis keskendub vaid veebirakenduste turvalisusele.

Käesoleva bakalaureusetöö eesmärk on leida sobilik õpperakendus, mida saaks kasutada ülikoolis kas veebiprogrammeerimise kursuse laiendamiseks või eraldi valikaine koostamiseks. Erinevaid õpperakendusi mis vaatavad turvalisust eksisteerib palju, ning tasemed ja keerukus erineb suuresti.

Eksisteerib ohtralt veebilehti, kus saab harjutada elementaarseid turvanippe, näiteks www.dareyourmind.net ja www.hackthis.co.uk. Samuti eksisteerib õpperakendusi, mis on mõeldud just elukutseliste turvatestijate (ingl k *pen-tester*) või kogenud veebiarendajate õpetamiseks.

Eesmärgi saavutamiseks tutvutakse Tallinna Ülikooli informaatika bakalaureuse õppekavaga ja tarkvara arendaja kutsestandardiga. Samuti vaadatakse üle veebirakenduste suurimad turvapuudused.

Töö ühe sammuna luuakse hindamismudel mis võtab arvesse kõike kriteeriume ning peamisi veebirakenduste nõrkusi. Mudel annab võimaluse hinnata erinevaid teadlikult ebaturvaliselt kirjutatud veebirakendusi ühte moodi ning leida neist sobiv õppetöös kasutamiseks.

Töö esimene peatükk tutvustab teoreetilisi aluseid hindamismudeli loomiseks. Teises peatükis keskendutakse hindamismudelile endale ning kolmandas räägitakse rakendustest endist.

1. Töö teoreetilised alused

Kuna vaja on valida rakendus õppetööks, tuleks enne uurida täpsemalt mis sisaldub õppekavas ja kutsestandardis ning mida pakub olemasolev programm. Tallinna Ülikooli informaatika õppekava võib siin pidada heaks sisendiks, mis annab ülevaate mida õpetab ülikool ja mis tehnoloogiatele keskendutakse.

Tehnilisema poole pealt taskuks kindlasti uurida OWASP riskide edetabelit, mis koondab endasse kõige levinumad vead, mida veebirakendustes tehakse. Samuti on OWASPi veebileht hea ressurss ka soovitatud leevenduste ning õpperakenduste leidmiseks.

1.1. Kutsestandard

Esimesena on aluseks võetud Eesti kvalifikatsiooniraamistiku (EKR) tarkvaraarendaja, tase 6 kutsestandard, millele tuginetakse rakenduskõrghariduse ja bakalaureuseõppe õppekavade koostamisel. Antud dokument on kooskõlas Euroopa kvalifikatsiooniraamistikuga ning annab kriteeriumid, millele peaks vastama õppekava läbinud spetsialisti teadmised.

Antud standardis on välja toodud kolm turvalisuse seisukohast olulist punkti: Tarkvara arendaja tööosadeks on tarkvaralahenduste kavandamine koostöös kliendiga, arendusprotsessis osalemine ning tarkvarasüsteemi realiseerimine (Infotehnoloogia ja Telekommunikatsiooni Kutsenõukogu, 2014)

Võib väita, et nimetatud töid läbi viies on oluline mõelda turvalisuse peale. Juhul, kui tarkvara algusest peale ilma turvameetmeteta projekteerida, on neid hiljem väga raske lisada ning samuti viiakse reaalsed meetmed sisse arendusprotsessi käigus.

Tarkvaraarendaja, tase 6 kutsestandardis on andmeturbe, ehk infoturbe põhimõtete tundmise vajalikkust mainitud kahe kompetentsitaseme all, mis on ära toodud tabelis 1. Turvalisust mainitakse põgusalt kahes punktis, mis on ka tabelis välja toodud

Tabel 1. EKR Tarkvaraarendaja, tase6 arhitektuuri projekteerimine väljavõte

B.2.2 Lahenduse arhitektuuri analüüsimine ja valimine (Arhitektuuri projekteerimine (eCF- kompetents A.5.))		EKR tase6
Kirjeldus	**Teadmised**	**Oskused**
Tagab lahenduse töölehakkamise, arvestades standardeid, levinumaid tehnoloogiaid ja andmeturbe põhimõtteid.	a) teab tüüpilisi arhitektuuri mustreid, **b) teab andmeturbe põhimõtted, standardeid ja levinumaid tehnoloogiaid,** c) teab erinevaid jõudluse tagamise viise.	a) kavandab sobiva arhitektuuri, b) valib sobivad komponendid, c) arvestab olemasolevate süsteemide arhitektuuridega, seob olemasolevate süsteemide arhitektuurid ja tagab nende koostöövõime, **d) arvestab jõudluse ja turvalisuse nõuetega,** e) arvestab lahenduse kuluefektiivsusega, f) arvestab tootestamise mõjuga.
B.2.5 Testimine (Testimine (e-CF kompetents B.3.))		EKR tase6
Kirjeldus	**Teadmised**	**Oskused**
Tagab süsteemi ootuspärase töötamise/käitumise.	a) teab testimise metoodikaid, b) teab erinevaid testimise vahendeid, **c) teab andmeturbe nõuded.**	a) valib sobivad testimismeetodid, b) määratleb ja kavandab testjuhud ning viib need läbi, c) arvestab konkreetse funktsionaalsuse kriitilisust, d) automatiseerib testimisprotsessi

1.2. Tallinna Ülikooli informaatika õppekava

Aluseks on võetud Digitehnoloogiate instituudi 2016/2017 õppeaastal kehtiv õppekava.

Uuest õppekavast on välja nopitud Tallinna Ülikooli informaatika õppekava veebiarendust ja programmeerimist puudutavad erialaained (vt tabel 2) ning vaadatud lähemalt, mis teemasid need õppeained sisaldavad ja millistes programmeerimiskeeltes praktikumid läbi viiakse.

Tabel 2. TLÜ õppekava programmeerimisained

Ainekood	**Aine nimi**
IFI6074.DT	Programmeerimise alused
IFI6069.DT	Programmeerimise põhikursus
IFI6076.DT	Veebiprogrammeerimine
IFI6068.DT	Sissejuhatus infosüsteemidesse
IFI6107.DT	Sissejuhatus infoturbesse
IFI6211.DT	Eesrakenduste arendamine
IFI6091.DT	Objektorienteeritud veebirakendused
IFI6095.DT	Veebiraamistikud
IFI6059.DT	Rakenduste programmeerimine

Programmeerimise alused keskendub programmeerimis alastele põhimõtetele ja andmetüüpidele ning siinkohal on veel vara andmeturbest rääkida. Õppetöös kasutatakse programmeerimiskeelt Python (IFI6074.DT ainekaart, 2016).

Programmeerimise põhikursus annab sissejuhatava ülevaate erinevatest programmeerimismeetoditest ning vaatleb eelmisel kursusel mainitud teemasid sügavamalt. Luuakse suuremaid ja keerukamaid rakendusi. Kursus keskendub programmeerimise õpetamisele Java keeles. Veebirakendusi ning võrguprogramme antud kursuse raames ei tehta (IFI6069 ainekaart, 2016).

Veebiprogrammeerimine on PHPd ja MySQLi tutvustav kursus, kus keskendutakse just veebirakenduste loomisele (IFI6076.DT ainekaart, 2016).. Ainekaardil mainitakse ära ka turvalisus, kuigi ainekonspekt seda olulisel määral ei käsitle (Kippar, 2009).

Sissejuhatus infosüsteemidesse käsitleb üldist infosüsteemide projekteerimist ning turvalisust kui sellist eraldi teemaks ei võeta, vaid räägitakse sellest põgusalt ja üldiselt (IFI6068.DT ainekaart, 2016).

Sissejuhatus infoturbesse on üldisi infoturbe põhimõtteid tutvustav aine. Kuna aine raames tehakse sissejuhatus paljudesse teemadesse, võiks seda käsitleda eeldusena, enne kui hakata veebi turvalisust süviti vaatama. Mainitakse ära ohud, turvaeesmärgid, tehakse sissejuhatus krüptograafiasse, mis on omakorda vajalik, et veebirakenduste turvalisust käsitleda, ning räägitakse ka juurdepääsukontrollidest, võrguturbest jms. Kõige tähtsama punktina võib välja tuua praktikumi „Erinevate veebitarkvarade turvanõrkuste tuvastamine ja analüüsimine“, mis puudutab muuhulgas murdskriptimist (ingl k. *XSS*) ja SQL süstimist (ingl. k *SQL Injection*) ning haakub käesoleva uurimuse teemaga kõige rohkem (IFI6107.DT ainekaart, 2016).

Paraku ei saa sissejuhatava aine suure mahu juures ühelgi teemal kaua peatuda, kuid kursus annab vajalikud eeldused, et üliõpilasel oleks võimalik veebi turvalisuse teemadesse paremini süveneda ning neist ka hõlpsamini aru saada.

Eesrakenduste (*ingl. k Front end applications*) **arendamine** keskendub vaid rakenduste arendamisele. Ainekaardil on välja toodud erinevad põhimõtted, loogika ja tehnoloogiad, mida käsitletakse, kuid turvalisust ei mainita (IFI6211.DT ainekaart, 2016).

Objektorienteeritud veebirakendused tutvustab objektorienteeritud põhimõtteid. Peamiselt kasutatakse Javascripti ja PHPd (IFI6091.DT ainekaart, 2016). Aine konspektist ei õnnestunud leida viiteid infoturbele (Kippar, 2014).

Veebiraamistikud on veebiarendajale tõenäoliselt kõige tähtsam kursus. Aine raames antakse kiire sissejuhatus enamlevinud veebiraamistike teemal ning tutvustatakse nende pakutavaid võimalusi. Aine toimub peamiselt praktikumi vormis. Räägitakse erinevatest PHP ja JavasScripti teekidest, puudutakse ka pilveservereid ning NoSQL andmebaase. Ka sellel kursusel turvateemasid ei käsitleta (IFI6095.DT ainekaart, 2016).

Rakenduste programmeerimine kätkeb endas Java rakenduste loomist, nii mobiili-, eraldiseisvate kui ka klient-server ja veebirakenduste loomist. Samuti vaadeldakse Java krüptograafia teeki (IFI6059.DT ainekaart, 2016).

Tallinna Ülikooli informaatika õppekava erialaainete analüüsi kokkuvõtteks võib öelda, et antakse väga head alused veebirakenduste loomiseks ning piisavalt sissejuhatust andmeturbesse, et turbeteemaga natuke sügavamalt tegeleda, kuid puudub põhjalikum ülevaade sellest, millised on levinumad rakenduste ründed ja kuidas end nende eest kaitsta ning puuduvad seega ka asjakohased

praktikumid. See kinnitab autori eeldust, et veebirakenduste turvalisust käsitlev aine võiks õppekavas eksisteerida või peaks täiendama veebirakenduste arendamist puudutavaid kursuseid ning selle jaoks oleks vaja head õppevahendit.

Samuti võib arvata, et eelistatud keel õpperakenduse jaoks võiks olla PHP, äärmisel juhul Java. Antud keeltega on tudengite kokkupuude kõige põhjalikum, seega saaks keskenduda rünnete läbimängimisele ning kaitsepõhimõtete läbiproovimisele nii, et programmeerimiskeel või selle mittepiisav oskus ei saaks takistuseks.

1.3. OWASP riskide edetabel

Kõige tähtsamaks allikaks siinses töös võib pidada OWASPi projekti raames loodud turvanõrkuste teadlikkuse tõstmiseks loodud edetabelit (*Open Web Application Security Project*).

OWASP on infoturbe ekspertide eestvedamisel tegutsev tasuta infoturbe alaseid materjale loov rahvusvaheline kogukond, mis hõlmab laia skaalat veebi turvalisusega seotud teemasid. Muuhulgas loovad nad juhendeid, õpperakendusi, turvatestimise proksit, dokumentatsiooni ja arendaja juhendeid ning tööriistu.

OWASPi turvalisusprojektidest võtavad osa turbespetsialistid üle maailma, kogukond on avatud ja igaüks võib kaasa lüüa. Kuna turvalisusteemaga tegelevaid silmapaare on seetõttu palju, võib OWASPi kodulehel asuvat informatsiooni pidada ajakohaseks, korrektseks ja usaldusväärseks. Oluline on mainida, et valeinformatsiooni on sinna keeruline lisada, sest iga juhendi ja dokumendi kohta koostatakse kogukonna poolt retsensioon.

OWASP riskide edetabel (OWASP TOP10) on 2003. aastal alguse saanud projekt, mis käsitleb edetabeli vormis veebirakenduste tavalisemaid vigu ning jagab soovitusi nende riskide minimeerimiseks. Dokumendi viimane versioon pärineb aastast 2013, kuid veapõhimõtted ja ründed on suuresti samad ning dokumenti võib pidada endiselt kehtivaks (OWASP, 2013).

Kuigi töö kirjutamise ajal on valmimas Owasp Top10 2017 on see dokuement siiski läbi vaatamise ning retsenseerimise faasis, plaaniga avaldada viimistletud versioon augustis 2017. (OWASP top10, 2017)

Lisaks tutvus autor ka 2017a dokumendiga ning ei leidnud sealt olulisi muudatusi. Märkimisväärsena võib lugeda rakendusliideste (ingl. k *API*) sissetoomise. (Owasp top10 2017 RC, 2017). Edasises töös kasutab autor siiski 2013a ilmunud dokumenti.

Tasub mainida, et OWASP riskide edetabel on loodud teadlikkuse tõstmiseks ning ei ole kinnitatud standard (Wichers, 2013). Samuti on OWASP riskide edetabel väga üldine riske käsitlev dokument, näiteks punkt A6 käsitleb tundliku teabe lekitamise teemat väga pealiskaudselt, kuigi kodulehel on viited põhjalikumale dokumentatsioonile.

OWASP riskide edetabel sisaldab endas järgmisi riske:

A1 süstimine (ingl. k *injection*). Pahatahtliku koodi või käsu süstimine rakendusse. Ei piirdu ainult teadaoleva SQL-i süstimisega, vaid sisaldab endas ka käskude süstimist serveri operatsioonisüsteemi, koodi süstimist rakendusse, mis seda seejärel jooksutab, ning IMAP süstimist, mis võimaldab otsest ligipääsu e-posti serverile, samuti LDAP süstimist jpm..

A2 Nõrk autentimine ja sessioonihaldus (ingl. k *Weak authentication and session management*). HTTP(S) on olekuvaba protokoll, st autentimisinformatsioon peab iga päringuga kaasas olema, nagu ka sessiooni võti. Sisaldab endas kasutaja või sessiooni ülevõtmist.

A3 Murdskriptimine (ingl k *XSS*). Käsitleb ründeid, kus ründajal on võimalik lisada enda kood veebirakendusse, mis seejärel pannakse tööle ohvri arvutis või veebilehitsejas.

A4 Ebaturvalised otseviited objektidele (ingl. k *Insecure Direct Object References*). Võimaldab ründajal saada kätte fail, väärtus või parameeter, mis peaks olema lubatud ainult kindlate ligipääsuõigustega kasutajatele. Teisalt lubab ka ründeid, kus objekti viide on välisele objektile, mis võetakse üle ning vahetatakse ära.

A5 Serveri konfiguratsiooni vead (ingl k *Security Misconfiguration*). Probleemid serveri konfiguratsiooniga või selle puudumine. Vaikeseadetes lubavad veebiserverid teinekord liiga paljudele ressurssidele ligi saada. Mõistlik on osa operatsioone ära keelata.

A6 Tundlike andmete paljastamine (ingl. k *Sensitive Data Exposure*). Käsitleb endas kõike, mis puudutab informatsiooni hoidmist, käsitsemist ning väljalekitamist, sealhulgas turvasoklite kihi puudumist (SSL), nõrka krüptograafiat, brauserisse või kolmandale osapoolele kogemata valede andmete saatmist.

A7 Puuduv ligipääsu kontroll (ingl k *Missing Function Level Access Control*). Kätkeb endas ründeid ja riske, mis tulenevad näiteks sellest, kui PDF raport asub failide kaustas ja lõppkasutajatele viidatakse otse faili kujul, kuigi tegelikult peaks seda tegema läbi veebiserveri ja mitte otse viidates. Või ei kontrollita, kas antud kasutaja peaks saama üldse failile ligi.

A8 Päringu murdvõltsimine (ingl. k *Cross Site Request Forgery*). Rünne, mis sunnib lõppkasutaja poolt sisse logitud veebirakendusel täitma ründaja poolt ette antud käsku. Käskudeks võivad olla näiteks raha ülekandmine etteantud kontole ja kõigi e-kirjade kustutamine.

A9 Haavatavate kolmanda osapoolte raamistike kasutamine. (ingl.k *Using Components with Known Vulnerabilities*). Näiteks erinevad sisuhaldustarkvarad või graafikute joonistamiseks kasutatav Javascripti teek jms.

A10 Kontrollimata ümbersuunamised ja edastamised (ingl. k *Unvalidated Redirects and Forwards*). Rünne, kus ründajal õnnestub suunata kasutaja veebilehitseja usaldusväärsest rakendusest ümber enda kontrollitavale veebilehele, kasutades HTTP-REDIRECT meetodit.

Kuna iga ründe ja vea teemal on kirjutatud pikki uurimistöid ning juba kas või OWASP A2 riski, sessioonihalduse kohta on kodulehel üleval seitse erinevat dokumenti, ei vaatle antud bakalaureusetöö lähemalt erinevaid viise sessioonihaldust rünnata ning nende kaitsemehhanisme. Seda tuleks muidugi teha õppematerjali loomise lõppstaadiumis kui rakendus on juba valitud.

Samuti on kirjutatud hulgaliselt uurimistöid rünnete tõrjumise teemadel ning autor ei pea otstarbekaks siinses töös rünnetest lähemalt rääkida. Näiteks Tallinna Ülikoolis 2011. aastal kaitstud Sten Schwede bakalaureusetöö „Levinumad Rakendusloogilised rünnakud veebilehtede vastu“ käsitleb põhjalikult murdskriptimise ja päringu murdvõltsimise ründeid ning annab ka ülevaade SQL-tüüpi süstimisest (*injection*) (Schwede, 2011).

2. Hindamiskriteeriumid

Hindamismudel on vajalik, et valida välja õppetööks sobiv teadlikult ebaturvaliselt kirjutatud rakendus. Mudel aitab luua kriteeriumid, mida järgides on võimalik hinnata kõiki rakendusi ühtemoodi. Võimalikult suur osa mudeli kriteeriume peaksid olema selged ja ühesed ning subjektiivselt võetavaid kommentaare peaks olema vähe.

Lähtutakse ka TLÜ informaatika õppekavast, tuuakse sisse OWASP TOP10 ning lähtutakse ka autori isiklikust kogemusest infoturbe valdkonnas – autor on 2012. aastast töötanud infoturbe vallas.

Ideaalne oleks, kui antud rakendusega on kaasas ka soovituslikud kaitsemehhanismid, ehk ühest rakendusest on 2 versiooni – haavatav ja turvaline.

Võetakse arvesse ka rakenduse paigaldamise lihtsust ning dokumentatsiooni.

2.1. Õppekavast tulenevad eeldused

Kuna eesmärk oleks eelkõige keskenduda turvalisusele, sai tehtud teadlik valik programmeerimis keele asjus, et vältida keele õppimisele kuluvat lisa-aega.

Lähtudes TLÜ informaatika õppekavast tundub, et tudengid veedavad kõige rohkem aega tehes veebirakendusi just PHP keeles. Samuti puututakse kokku ka Java keelega, kuid mitte veebirakendusi tehes. Lisaks võib kaaluda ka Pythonit, kuna antud programmeerimiskeel on inimloetav ja lihtne aru saada.

OWASP riskide edetabelist võetakse sisse kõik ründed välja arvatud A5 – serveri konfiguratsiooni vead ning A9 – haavatavate kolmanda osapoole teekide kasutamine.

Serveri konfiguratsiooni vead jäetakse välja, kuna tegemist on väga spetsiifiliste vigadega mis on igal veebiserveril ka erinevad. *Apache* veebiserveri puhul toimivad ründed ja kaitsed ei kehti näiteks IIS veebiserveri korral. Jättes selle välja, keskendume vaid veebirakenduse loogikakihile ning rakenduse enda vigadele

Haavatavate kolmanda osapoolte teekide kasutamine jäetakse välja, kuna see ei anna midagi täiendavat juurde. Päris rakendustes toovad kolmanda osapoole teegid sisse turvavigu, mille üle puudub kontroll ning mida arendajal on raske märgata või korda teha. Siinkohal aitab CVE-de

(*Common Vulnerabilities and Exposures*), ehk avastatud ja avalikustatud turvanõrkuste jälgime Riske hinnates tooksid aga kolmanda osapoole teegid sisse täpselt samasuguse turvaaugud nagu arendaja enda rakendus võib seda teha, ning kõik eelnevad on juba muude OWASPi edetabeli punktidega kaetud.

2.2. Autori kogemus

Autor arvestab ka enda isiklikku kogemust mis on tekkinud aja jooksul antud valdkonnas tööd tehes.

Esiteks tahaks autor rõhutada, et kindlasti tuleks tähelepanu pöörata sessiooni haldusele, nagu mainitud ka OWASPi projektis. Lisaks OWASPi soovitustele võiks ka sessioonivõtme, või muude tundlike parameetrite saatmisel HTTP päringute sees või üle URL parameetrite sisse võtta. Samuti tuleb tähelepanu pöörata sessiooni kestvusele, ehk kui lehel mõnda aega midagi ei tehta, ei ole sessioon enam aktiivne. Ning kindlasti ei tohiks võimalik olla võtta URL koos sessiooni parameetritega, ning selle abil terve sessioon üle võtta.

Teise punktina peab autor väga oluliseks krüptograafiat ning selle õiget kasutamist. Nõrga algoritmiga või valesti teostatud krüptograafia on siinkohal hullem, kuna see võib luua illusiooni, et rakendus on turvaline ja andmed on kaitstud.

Kindlasti peaks siin vaatama krüptograafiliste algoritmide tugevust, ning rääkima mis ohud kaasnevad, kui kasutada aegunud algoritme. Täpsemat infot krüptograafiliste algoritmide kohta saab RIA krüptograafiliste algoritmide elutsükli uuringust[1]

Samuti peaks võtma vaatluse alla juhuarvude genereerimise. Elust enesest on olukord, kus algoritm on tugev ning krüptograafia oleks toimiv, kuid võtme arvutamiseks kasutatakse ette arvatava või tuletatava väljundiga juhuarvu generaatorit. Peaks tooma mõned näited, kuidas ei tohiks genereerida juhuarve, näiteks lineaarsed juhuarvude generaatorid, või sessiooni võtmed, kasutaja nimest vms parameetrist tuletatud krüptograafiline võti. Samuti peaks tooma näiteid kuidas tuleks genereerida juhuarve ning mida arvestada.

[1] https://www.ria.ee/public/RIA/Kruptograafiliste_algoritmide_uuring_2015.pdf

Siinkohal peaks mainima ka võtme haldust ning võtmete transporti, üle vaatama kuidas võtmeid rakenduses hoida ja käsitleda, milline rakenduse kiht peaks tegelema andmete krüpteerimisega, kas seda peaks tegema serveris või eesrakenduses jne.

2.3. Valminud hindamismudel

Valminud hindamismudel käsitleb järgmisi punkte:

Rakenduse loomiseks kasutatud programeerimiskeel ja tehnoloogia. Sisend on võetud Tallinna Ülikooli informaatika õppekavast. Rakendus peaks eelistatult olema kirjutatud PHP keeles, kuid kaaluda võib ka Java või Pythoni keeles kirjutatud programme. See välistaks vajaduse kulutada olulises mahus aega keele õppimiseks.

Lähtutakse põhimõttest, et rakenduse keel ja tehnoloogia ei tohi saada takistuseks. Rakendus peaks olema ka lihtsasti paigaldatav.

Rakenduse elukaare olek. Kas rakendust uuendatakse aktiivselt, millal on viimane versioon rakendusest ilmunud. Kuna interneti turvalisuse teemad on väga kiiresti aeguvad, ei ole mõtet võtta valikusse väga vana rakendust. Teinekord leitakse ka ammu kasutusel olevates ning turvaliseks peetud rakendustes vigu. Aegunud rakendust loetakse hindamismudelil üheks välistavaks teguriks

Rakenduse tehnilised nõuded. Info mõttes ja võrdluseks tasub kirja panna, et oleks näha millised rakendused lähevad lihtsalt tööle ning millistel on vaja väga spetsiifilist serverikeskkonda.

Dokumentatsiooni olemasolu. Väga oluliseks kriteeriumiks on dokumentatsiooni olemasolu. Dokumentasiooni kohapealt vaatame järgmisi punkte:

- Kas eksisteerib dokumentatsioon paigaldamise kohta?
- Kas eksisteerib dokumentatsioon rünnete kohta?
- Kas eksisteerib dokumentatsioon soovituslike kaitsemeetmete kohta?

Kas on realiseeritud ka kaitse mehhanismid?

Oluline kriteerium on see, et rakendusest võiks olemas olla kaks sama funktsionaalsusega versiooni, millest üks on turvaline ning teine mitte. Lisaks oleks suur abi täpsest lugemismaterjalist turvalises versioonis kasutatud kaitsemehhanismide kohta. Juhul, kui rakendus on ainult vigane ega sisalda rünnete eest kaitseid, tuleks see lugeda poolikuks, kuna keskenduda võiks siiski ka sellele, kuidas rakendusi hästi teha.

Rünnete ning funktsionaalsuste vastavus

Antud töö ulatuses otsustati välja valida järgmised ründed: A1 – süstimine, A2 – nõrk sessioonihaldus, A3 – murdskriptimine, A4 – ebaturvaline objektidele viitamine, A6 – tundlike andmete lekitamine, A7 – puuduv ligipääsuõiguste kontroll, A8 – Päringu murdvõltsimine ning A10 –kontrollimata ümbersuunamiste ründed.

Samuti võtaks autor omalt poolt sisse veel järgmised punktid:
Nõrk krüptograafia, Võtmehaldus ja krüpteeritud andmete salvestamine, nõrk juhuarvude genereerimine.

Kokku saadakse järgmised punktid funktsionaalsuse võrdlemiseks:

- Süstimis tüüpi ründed
- Sessiooni puudutavad ründed
- Murdskriptimine
- Ebaturvalised otseviited objektidele
- Tundlike andmete paljastamine
- Nõrk krüptograafia, aegunud algoritmid
- Võtmehaldus, andmete salvestamine
- Nõrk juhuarvude genereerimine
- Puudulik õiguste kontroll
- Päringu murdvõltsimine
- Kontrollimata ümbersuunamised

Antud punkte võiks hinnata järgmiselt (tabel)

Tabel 3Funktsionaalsuse hindamine

Jah	Rakenduses on olemas vastav funktsionaalsus / vastab nõudele
Ei	Rakenduses puudub vastav funktsionaalsus / ei vasta nõudele
PL (pole rakendatav)	Antud nõue selle rakenduse kohta ei kehti (Näiteks salvestatud andmete krüpteerimine, kui rakendus andmeid ei salvesta)
PI (puudub informatsioon)	Puudus võimalus veenduda, kas puuduva dokumentatsiooni, testimise keerukuse või muude probleemide tõttu

Hindamistulemuste vormistamiseks koostas töö autor järgmise vormi.

Tabel 4 Hindamistulemused

	A1. Rakenduse info	
Rakenduse nimi		
Rakenduse koduleht		
Rakenduse tehnilised nõuded		
Rakenduses kasutatavad keeled/tehnoloogiad		
	A2. Rakenduse paigaldamine	
Kommentaar		
	A3. Dokumentatsiooni olemasolu (Ei, Jah, Pole rakendatav, Puudub info)	
Dokument	Hinnang	Kommentaar
A3.1 Paigaldamist abistav dokumentatsioon		
A3.2 Rünnete/Vigade dokumentatsioon ja näited		
A3.3 Soovitatud paranduste dokumentatsioon		
A3.4 Soovitatud paranduste koodinäited		
Muu.		
	A4. Rakenduse vastavus eel-määratud turvalisuse nõuetele	
Funktsionaalne nõue	Hinnang	Kommentaar
A4.1 Süstimis-tüüpi ründed		

A4.2 Sessiooni/Autentimist puudutavad ründed		
A4.3 Murdskriptimine		
A4.4 Ebaturvalised otseviited objektidele		
A4.5 Tundlike andmete paljastamine		
A4.6 Nõrk krüptograafia		
A4.7 Võtmehaldus		
A4.8 Nõrk juhuarvude genereerimine(Krüptograafia, sessiooni võtmete jms puhul)		
A4.9 Puudulik õiguste kontroll		
A4.10 Päringu murdvõltsimine		
A4.11 Kontrollimata ümbersuunamised		
	Muud kommentaarid	

Autor mainib siinkohal ära, et nõuded on üldised. Vaadatakse, kas murdskriptimis ründed on rakenduses olemas, kuid ei hakata eraldi lahkama murdskriptimis rünnete tüüpi ning erinevaid kaitsemehhanisme. Kuna antud rakenduse kasutamine oleks siiski bakalaureuse õppes rakendatud, siis ei pea autor vajalikuks olla hindamismudelis nii täpne ning täielikult kõik ära katta, vaid anda ettekujutus rünnetest ja enam levinud kaitse mehhanismidest ning alus tudengile iseseisvalt ja konkreetsemalt teemadega edasi tegeleda, kui selleks tekib vajadus.

3. Rakenduste võrdlemine

Valitud rakendusi testitakse süstemaatiliselt samasuguse mudeli alusel. Pärast esialgset rakenduste otsingut vaadatakse esmalt dokumentatsiooni, projekti uuendusi, viimati muudetud kuupäeva jms, et mõningad rakendused välistada. Samuti välistatakse kohe alguses rakendused, mille probleemiks on tehnoloogia mittesobivus – ehk rakendus eeldab mõne programmeerimis keele tundmist mida koolis ei ole õpetatud või mõne spetsiifilise tehnoloogia tundmist mille õppimine võib võtta aega.

Testimiseks kasutatakse võimsuselt tagasihoidlikku sülearvutit. Muuhulgas jälgitakse mälu ja ressursi kasutust, juhul kui see osutub suureks. See peaks tagama, et antud rakendusi oleks kindlasti võimalik jooksutada arvutiklassides ja ka tudengitel vanemate arvutite peal.

Testimiseks kasutatud arvuti **Thinkpad T440** olulisemad parameetrid:

- Protsessor: Intel I5-4200 Ultra Low Voltage
- Operatiivmälu: 12GB
- HOST Operatsioonisüsteem: Windows 10
- Kasutatud virtualiseerimis platvorm: Oracle VirtualBox[2]

Kasutatud tarkvara:

- Fiddler [3]- veebiproksi
- Wireshark [4]– võrgupakettide salvestamise ja vaatamise tööriist
- Burp Suite [5]– veebiproksi koos sisse ehitatud rünnete moodulite ning skriptimisega

Rakenduste valikul kasutati nii OWASPi kodulehte, infoturbeteemalisi internetifoorumeid, Stackoverflow programmeerimis teemalist foorumit ning otsingumootoreid kui ka infoturbespetsialistide, konkreetsemalt küsiti eravestlusel Tšehhi OWASPi liikme ning Londoni Turvalise Interneti kogukonna liikme soovitusi. Lisaks kattusid arvamused artikliga, mis andis ülevaate 15 parimast katkisest veebirakendusest (Vonnegut, 2015).

[2] https://www.virtualbox.org
[3] http://www.telerik.com/fiddler
[4] https://www.wireshark.org
[5] https://portswigger.net/burp/

3.1. Võrdlemise metoodika

Võrdlemisel alustatakse rakenduse dokumentatsiooni kontrollist. Edasi ei ole mõtet vaadelda rakendust, millel puudub dokumentatsioon või see on aegunud. Kontrollitakse, mis tehnoloogiat antud rakendus kasutab ning millises programmeerimiskeeles see on realiseeritud. Samuti vaadatakse, millal rakendust viimati uuendati.

Valitud rakenduse puhul tehakse näidispaigaldus virtuaalmasinasse, tehakse märkmeid, hinnatakse dokumentatsiooni. Esialgne rünnete ja funktsionaalsuste tabel täidetakse dokumentatsiooni põhjal, kui võimalik.

Järgmise punktina proovitakse läbi ka mõningaid ründeid. Vaadatakse, kas on olemas ka kaitsemehhanismid või soovitused, kuidas paremini toimida.

Kõiki ründeid ei ole siinse töö raames vaja läbi vaadata. Esiteks ei annaks see sisulist väärtust, vaid oleks lihtsalt kontrolliks, et rakendus vastab dokumentatsioonile. Kuna ühte tüüpi ründeid on erinevaid, ei saa ka iga rakendust testida sama tööriistaga, selletõttu oleks see väga ajamahukas.

Samuti ei hakata hetkel hindama soovituste ja paranduste ajakohasust, kuna sama tüüpi ründeid võib olla mitmeid ning iga ründe korral võib olla ka mitu erinevat viisi see maandada, ei hakata vähemalt valiku käigus sellele tähelepanu pöörama. Ainuke asi, mis aitaks, oleks koodiülevaade (ingl k *code review*) koos kasutatud meetodi analüüsiga, mille maht kasvab 100 turvaveaga rakenduse puhul mitmesaja tunnini.

Samuti võib koheselt välistada nn Musta kasti tüüpi õpperakendused, millel dokumentatsioon rünnete kohta puudub. Elukutselise spetsialisti treenimiseks on need head, kuid ülikoolis kasutamiseks ja õppematerjali loomiseks liialt keerukad. Samuti tuleks selliste rakenduste puhul suure põhjalikkusega otsida üles kõik turvavead, lugeda läbi kogu rakenduse kood ning see kõigepealt ära dokumenteerida.

3.2. Leitud rakendused ja rakenduste valik

Esialgsesse valikusse võeti järgmised rakendused

- bWAPP
- Damn Vulnerable Web Application (DVWA)
- Google Gruyere
- InsecureWebApp
- McAfee Free Tools (mitu rakendust)
- OWASP Mutillidae II
- OWASP Security Shepherd
- The ButterFly - Security Project
- OWASP WebGoat
- OWASP Bricks

3.2.1. Eelvaliku põhjal välistatud rakendused

InsecureWebApp[6] – 2004. aastal loodud rakendus, mis mõeldud just õpetamiseks. See on kirjutatud Javas ning kasutusel on ka MySQL. Kahjuks pärineb viimane tarkvaraversioon aastast 2005. Seega saab rakenduse välistada.

McAfee Free Tools [7]– McAfee pakub mitut veebiturvalisusele keskenduvat rakendust, täpsemalt on olemas projektid Hacme Casino, Hacme Books, Hacme Travel ja Hacme Bank. Kahjuks puudub esmavaatlusel informatsioon selle kohta, millal neid viimati uuendati. Nõuded Windows XP-le ja .NET raamistiku versioon 1.1-le on ka põhjused, miks antud rakendusi hetkel põhjalikumalt edasi ei vaadelda. Samuti on rakendused kirjutatud .NET raamistikule, raamistikule Ruby on Rails või C++-is, millega kokkupuude ei ole tudengitel tõenäoliselt piisav, et hakata veebirakenduste turvalisust õpetama neid tehnoloogiaid kasutades.

OWASP Security Shepherd [8]– Security Shepherd on pigem raamistik kui rakendus. See on kirjutatud Javas ning pakub võimalusi luua enda harjutusi, test-ülesandeid ja lehekülgi. Hetkel vaatame valmisrakendusi. Kuigi Java on lubatud programmeerimiskeel ei saa antud rakendus kasutatud programmeerimiskeele tõttu kõrget prioriteeti, kui võtta lisaks fakt, et tegemist on siiski raamistikuga ning mitte valmis rakendusega.

[6] https://www.owasp.org/index.php/Category:OWASP_Insecure_Web_App_Project
[7] https://www.mcafee.com/sg/downloads/free-tools/index.aspx
[8] https://github.com/OWASP/SecurityShepherd/releases/tag/v3.0

The ButterFly - Security Project [9]– ButterFly on PHPs kirjutatud rakendus, mille viimane versioon pärineb aastast 2008. Plussina mainitakse küll ära, et kaasas on ka korrektne implementatsioon. Samuti on kaasas olev dokumentatsioon äärmiselt detailne ja põhjalik. Kahjuks on rakenduse paigaldamine keeruline ning see jookseb vanadel PHP ja SQLi versioonidel. Samuti võib üheksa aastat vanu kaitsemehhanisme pidada tänapäeval aegunuks.

OWASP WebGoat [10]– on üks kuulsamatest ebaturvalistest veebirakendustest. See katab peaaegu täielikult ära OWASPi Top10 ründed ning on kirjutatud Javas. Toetab samuti enda rünnete ja õppelehtede loomist ning on võimalik kasutada kui raamistikku. Rakendus aga pooldab nn musta kasti testimist, kus on küll vihjed, aga täpne vigade kirjeldus puudub.

Damn Vulnerable Web Application (DVWA) [11]**–** DVWA on PHPs kirjutatud veebirakendus aastast 2010. Kuna rakendusel puudub igasugune dokumentatsioon ning rakendus on ka seitse aastat vana, jäeti see kõrvale.

3.2.2. Eelvaliku põhjal võrdlemiseks sobivad rakendused

bWAPP [12]– bWAPP on PHPs kirjutatud ning MySQLi kasutav ebaturvaline veebirakendus, kus on üle 100 turvaaugu. Viimane versioon on pärit aastast 2014. See on küll kolm aastat vana, kuid loeme selle piisavalt uueks, et rakendust edasi vaadelda. Samuti tundub dokumentatsioon piisav olevat.

Google Gruyere [13]– Pythonis kirjutatud ja 2010. aastal uuendatud veebiturvalisuse rakendus. Erinevalt DVWAst on dokumentatsioon piisavalt täpne ning sisaldab näiteid, kuidas ründeid ära hoida. Seetõttu võeti rakendus võrdlusesse, kuigi keeleks on Python.

OWASP Mutillidae II [14]– PHP rakendus, mis katab ära OWASPi Top10 2007. ning 2010. aasta edetabelid. Olemas on dokumentatsioon, kuid mitte iga ründe kohta. Rakendus toetab LAMPi, XAMPPi. Rakendust uuendatakse aktiivselt, samuti on olemas näited, kuidas tuleks antud turva-augud ära parandada. Seepärast vaadeldi antud rakendust põhjalikumalt edasi.

[9] https://sourceforge.net/projects/thebutterflytmp/
[10] https://www.owasp.org/index.php/Category:OWASP_WebGoat_Project
[11] http://www.dvwa.co.uk
[12] http://www.itsecgames.com
[13] https://google-gruyere.appspot.com
[14] https://sourceforge.net/projects/mutillidae/files/

OWASP Bricks [15]– Bricks on PHP ja MySQL veebirakendus. Viimane versioon pärineb aastast 2013. Kuna dokumentatsioon ja näited on ülevaatlikud ning rakendus ei ole liiga vana, vaadeldi seda lähemalt.

3.3. Rakenduste võrdlus

Rakenduste võrdluse täpsemad andmed on esitatud lisades 1–4, kuid mõningad olulised tegurid võiks eraldi välja tuua. Nagu näha tabelist 5, siis täielikku katvust ei olnud ühelgi rakendusel. Samas tasub ära mainida, et kolm rakendust neljast sisaldasid ründeid, mida autor ei olnud oma mudelisse sisse võtnud. Samuti väärib märkimist, et ainult üks vaadeldud rakendus kätkeb näpunäiteid antud rünnete leevendamiseks.

Tabel 5 Rakenduste võrdlus

	Gruyere	bWAPP	Mutillidae 2	Bricks
Vigade arv rakenduses	24	100	~250	10
Vigade katvus nõuete suhtes	5/11	8/11	8/11	3/11
Kas eksisteerib ka näide, kuidas teha õigesti	Jah	Ei	Ei	Ei

Kõik vaadeldud rakendused olid lihtsalt paigaldatavad ning abistav dokumentatsioon oli hea.

Küll aga võib tuua välja, et turvavigu puudutav dokumentatsioon oli tõeliselt hea vaid rakendustel Gruyere ja Bricks. bWAPPi puhul on küll olemas vigade loetelu, kuid ei ole täpsemat dokumentatsiooni selle kohta, kus need asuvad ning kuidas neid rünnata.

Mutillidae vigade nimekirjas on üle 250 vea, kuid dokumentatsioon ja õppevideod on olemas vaid mõnekümne kohta. Enamik vigu tuleb ise üles leida. Samuti jäi autorile mulje, et Mutillidae rakendus on mõeldud juba elukutselistele kogemustega arendajatele või turvatestijatele, kes orienteeruvad teemas vabalt ja suudavad ilma erilise vaevata kasutada rünnete koostamiseks vajalikke tööriistu. Samuti on Mutillidae dokumenteerimata ründed vägagi keerulised ning nõuavad kogemust ja teadmisi, et neid turvaauke ära kasutada.

Kõige paremini realiseeritud oli Google Gruyere . Funktsionaalsuselt, kasutajamugavuselt ning dokumentatsioonilt oleks antud rakendus ideaalne, kui ta kataks ära rohkem ründeid . Tehniliselt

[15] http://sechow.com/bricks/download.html

oli rakendus kõvasti lihtsam kui Mutillidae või bWAPP, ka ründeid sooritada oli lihtne, ei olnud vaja kuigi palju spetsiifilisi tööriistu või taustateadmisi. Samuti olid rünnete kõrvaldamise meetmed väga hästi lahti seletatud. Kahjuks aga pole antud rakenduse vigade nimekiri piisav.

4. Tulemused

Uurimistöö tulemusena ei leitud õppetööks kasutamiseks sobivat teadlikult ebaturvaliselt kirjutatud veebirakendust. Suurem osa rakendusi oli realiseeritud ebasobivas programmeerimiskeeles või olid need lihtsalt liiga vanad. Samuti tulid olemasolevatel rakendustel ilmsiks olulised puudujäägid dokumentatsioonis - ei kirjeldatud piisavalt detailselt ründeid ja/või rünnete kaitsemehhanisme. Mõnel rakendusel puudus dokumentatsioon sootuks.

Leiti kaks väga hea ideega ja hästi realiseeritud rakendust – Google Gruyere ja OWASP Brick –', mis sobiksid dokumentatsiooni, kasutajamugavuse, lihtsuse ja realisatsiooni poolest väga hästi ülikoolis kasutamiseks, kuid antud rakendustel oli väga vähe ründeid realiseeritud ning katvus halb.

Mutillidae oli teisalt väga suure vigade arvu ja suure katvusega õpperakendus, kuid paraku eksisteeris dokumentatsioon vaid osaliselt. See osa dokumentatsioonist, mis oli olemas, oli vägagi põhjalik, ent paljud vead on seal katmata ja puudu on soovitused, kuidas end antud vigade eest kaitsta. Samuti tuleb selle rakendusega tööd tehes kasutada keerukaid tööriistu, nagu näiteks Burp Suite, mis algajatele suure tõenäosusega üle jõu käivad. Palju testimist toimub ka nn musta kasti meetodil. Vigade esindatuse poolest oli antud rakendus kõige parem, kuid osutus liiga keeruliseks.

Lisaks ei käsitlenud mitte ükski vaadeldud rakendus nõrka krüptograafiat rakenduse tasemel (vaadeldi küll vana SSL versiooni kuritarvitamist) ja juhuarvude genereerimist, millest tulenevate rünnete ja nende vältimise demonstreerimist autor aga õppetöös oluliseks peab.

Ühe olulise punktida võib mainida ka soovituste puudumist. Vigu oli igas rakenduses kuid puudus hea näide etalonteostusest kuidas peaks antud probleeme vältima või riske leevendama. Antud materjalid tuleks luua kas ise, koos näite koodiga, toetama juba olemas olevat rakendust või siis realiseerida ära enda kirjutatud õpperakenduses.

Autor leiab, et praegusel kujul ei sobi ükski vaadeldud rakendustest piisavalt hästi teema õpetamiseks ülikoolis. Tuleks luua rakenduse funktsionaalsete nõuete spetsifikatsioon ning realiseerida see rakendus koos kogu vajaliku dokumentatsiooniga. Kogu rakendust ei pea tingimata kirjutama täielikult iseseisvalt, tõenäoliselt saaks kasutada kas WebGoati või Security Shepherdi raamistikku. Lisaks kaalub Tallinna Ülikool PHP keele vahetamist Node.JS vastu veebitehnoloogiate õpetamisel, seega tasuks ette mõeldes kaaluda ka sobiva rakenduse tegemist hoopis Node.JS tehnoloogial.

Samuti tasuks üle vaadata ka OWASP 2017a edetabel ning võtta sisse rakendusliideseid puudutavad turbenõuded ning nõrkused.

Samuti on tekkinud eesti turul ka mõned ettevõtted, ühe näitena RangeForce (Serious training for serious attacks, 2015) kes just arendavad kommerts platvorme arendajate ja administraatorite treenimiseks. Antud tarkvarad ei ole veel valmis ning on beeta-testimise faasis, samuti puudub töö autoril informatsioon antud platvormi litsentsi hinna suhtes, ning autor ei tea kas antud ettevõtted oleks nõus koostööd ülikoolidega seda tasuta võimaldama, Kuid seda tasuks kindlasti uurida.

Kokkuvõte

Autor alustas antud tööd 2015/2016 õppeaastal sooviga leida rakendust, mida saaks kasutada õppe materjalina. Autor tutvus õppekava ning õppekava loomise alusdokumentidega. Samuti uuris, mis on kirjas kutsestandardis. Võttes juurde OWASP riskide edetabeli ja isikliku kogemuse sai koostatud hindamismudel mille vastu siis rakendusi võrreldud.

Töö kirjutamise käigus hakati ka uuendama OWASPi edetabelit, kuid OWASPI 2017a edetabel ilmub suve lõpus, ei sisaldanud erilisi muudatusi siis toimiti edasi vana edetabeli järgi.

Autor jättis teadlikult välja osa tarkvara mis on suunatud nö elukutselistele professionaalidele kes on juba omandanud mõned aastad reaalselt töö ning arhitektuuri kogemust, või tarkvara mis on mõeldud ründe-testijate koolitamiseks, pidades seda liiga keerukaks ülikooli esimeses astmes.

Autoril ei õnnestunud leida ühtegi valmisrakendust mis vastaks järgnevale kolmele üldkriteeriumile korraga:

1) ajakohane ja uuendatud;

2) sisaldaks erinevaid tüüpi turvanõrkusi korraga;

3) oleks olemas ka soovituste ja paranduste juhised.

Autor järeldas, et ükski hinnatud tarkvaradest ei sobi antud eesmärgiks. Edasiarenduse ja täiendusena tuleks kindlasti luua vastav tarkvara, kas siis ise või mõne raamistiku peale, koos dokumentatsiooni, vigade parandustega

Samuti tuleks tööd jätkates mõelda veel ühe korra läbi hindamismudel ning lisaks veebi- ja mobiilirakendustele võtta sisse ka asjade interneti ja rakendusliideste valdkond mida mainiti töö ajal uuendatud OWASPi edetabeli 2017a mustandis.

ˇ

Summary

Using Deliberately Insecure Web Applications in Teaching Process

The aim of this thesis was to find a suitable deliberately insecure web application to teach bachelor level computer science students application security. Idea was to find a web application full of vulnerabilities, but ideally it should also have good documentation and suggested mitigations for vulnerabilities.

To achieve the goal several steps were made. First, author read through Occupational standards to get familiar with recommendations and requirements of „Software developer “career path. In addition, curriculum of The Tallinn University School of Digital Technologies was digested to find how much does it touch security topics.

Secondly, assessment model was created based on personal work experience in the field of information security and OWASP 2013 Security awareness Document. Additional resources were gathered about the attacks from thesis done previously.

Author tried to focus on tooling what would be usable to second or third year bachelor’s students, leaving out professional software used for penetration testers training and qualification exams as well as paid, expensive commercial platforms. In addition, some of the software was left out because it was just out of date or used a technology what is not familiar to students of Tallinn University Informatics.

After pre-selected software was compared and tested, results were surprising. There is no suitable software for such goal as of today. Some of the applications were updated, had very good documentation but coverage on the security issues was poor. None of the tested applications had actual sample code for mitigations or suggested fixes.

As a result, if such software is needed we need to consider other possible options. If ready-made software is not a suitable option, one possible solution would be to create specification and code one from scratch, this would give us exactly what we need but is hard to keep updated and maintained.

But it would be also possible to use existing framework and build our vulnerable application on top of that, this would mean amount of development work is significantly reduced, but we still need to create scenarios, mitigations and documentation to support it.

Lastly there are several commercial platforms developed also in Estonia. One possibility would be to co-operate with them as well to integrate the usage in university curriculum.

In conclusion, author tried to find a good vulnerable web application, suitable for the level of second or third year students to practice their application security skills, but this goal was not achieved. It was found, that all existing software would need a lot of improvement or learning application should be specifically made just for this use case.

Kasutatud kirjandus

Balakrishnan, A. M. (kuupäev puudub). *Bricks Documentation*. Allikas: http://sechow.com/bricks/docs/

Denihan, M., & Duggan, S. (kuupäev puudub). *OWASP*. Allikas: OWASP Security Shepherd: https://github.com/OWASP/SecurityShepherd

Dewhurst Security. (kuupäev puudub). *Damn Vulnerable Web Application (DVWA)*. Allikas: http://www.dvwa.co.uk

Infotehnoloogia ja Telekommunikatsiooni Kutsenõukogu. (2014). *Kutsestandardid*. Allikas: Tarkvaraarendaja, tase 6: http://www.kutsekoda.ee/et/kutseregister/kutsestandardid/10546992/kirjeldus

Insecure Webapp Repository. (2005). Allikas: https://sourceforge.net/projects/insecurewebapp/files/

Kippar, J. (2009). *Veebirakenduste loomine PHP ja MySQLi abil.* Tallinn: Tallinna Ülikool, Informaatika Instituut.

Kippar, J. (2014). *Veebirakenduste jätkukursus.* Tallinn: Tallinna Ülikool.

Leban, B., Bruce , M., & Tabriz, P. (2010). *Google Gruyere*. Allikas: https://google-gruyere.appspot.com

McAfee. (kuupäev puudub). *McAfee Free Tools*. Allikas: http://www.mcafee.com/us/downloads/free-tools/index.aspx

Mesellem, M. (kuupäev puudub). *bWAPP an extremely buggy web app*. Allikas: bWAPP: http://www.itsecgames.com

OWASP. (2005). *InsecureWebApp*. Allikas: http://insecurewebapp.sourceforge.net/main/index.html

OWASP. (2013). *OWASP top 10 - 2013 The Ten Most Critical Web Application Security Risks.*

OWASP. (2016). *OWASP Mutillidae II*. Allikas: https://sourceforge.net/projects/mutillidae/

OWASP. (2016). *The Open Web Application Security Project*. Allikas: owasp.org

OWASP. (kuupäev puudub). *OWASP WebGoat*. Allikas: https://github.com/WebGoat/WebGoat

OWASP top10. (2017). Allikas: https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project

Owasp top10 2017 RC. (2017). Allikas: https://github.com/OWASP/Top10/raw/master/2017/OWASP%20Top%2010%20-%202017%20RC1-English.pdf

Rajs, R. (kuupäev puudub). *The ButterFly - Security Project*. Allikas: https://sourceforge.net/projects/thebutterflytmp/

Schwede, S. (2011). *Levinumad rakendustehnoloogilised rünnakud veebilehtede vastu (Bakalaureusetöö).* Allikas http://www.cs.tlu.ee/teemaderegister/.

Serious training for serious attacks. (November 2015. a.). Allikas: Barclays : https://www.home.barclays/news/2015/11/Rangeforce.html

TLÜ DTI. (2016). *IFI6059.DT ainekaart.* ois.tlu.ee.

TLÜ DTI. (2016). *IFI6068.DT ainekaart.* ois.tlu.ee.

TLÜ DTI. (2016). *IFI6069.DT ainekaart.* ois.tlu.ee.

TLÜ DTI. (2016). *IFI6074.DT ainekaart.* ois.tlu.ee.

TLÜ DTI. (2016). *IFI6076.DT ainekaart.* ois.tlu.ee.

TLÜ DTI. (2016). *IFI6091.DT ainekaart.* ois.tlu.ee.

TLÜ DTI. (2016). *IFI6095.DT ainekaart.* ois.tlu.ee.

TLÜ DTI. (2016). *IFI6107.DT ainekaart.* ois.tlu.ee.

TLÜ DTI. (2016). *IFI6211.DT ainekaart.* ois.tlu.ee.

Unruh, T., Skoruppa, M., Maggi, F., Rieck, K., Seifert , J.-P., & Yamaguchi , F. (2017). Leveraging Flawed Tutorials for Seeding.

Wichers, D. (2013). OWASP top10 presentation.

Vonnegut, S. (2015). *Checkmarx blog*. Allikas: 15 Vulnerable Sites To (Legally) Practice Your Hacking Skills: https://www.checkmarx.com/2015/04/16/15-vulnerable-sites-to-legally-practice-your-hacking-skills/

Lisad

Lisa1: bWAPP andmed

	A1. Rakenduse info	
Rakenduse nimi	bWAPP	
Rakenduse koduleht	http://www.itsecgames.com	
Rakenduse tehnilised nõuded	1) Oracle Virtualbox või 2) veebiserver, SQl server, WAMP või XAMPP	
Rakenduses kasutatavad keeled/tehnoloogiad	PHP ja MySQL, Javascript, HTML	
	A2. Rakenduse paigaldamine	
Kommentaar		
	A3. Dokumentatsiooni olemasolu(Ei, Jah, pole rakendatav, Puudub info)	
Dokument	Hinnang	Kommentaar
A3.1 Paigaldamist abistav dokumentatsioon	Jah	Olemas, lühike kuid piisav
A3.2 Rünnete/Vigade dokumentatsioon ja näited	Ei	Pakuvad raha eest koolitust, ainult üldine loetelu kättesaadav
A3.3 Soovitatud paranduste dokumentatsioon	Ei	Puuduv
A4.3 Soovitatud paranduste koodinäited	Ei	Puuduv
A4.4 Turvavigade loetelu	Jah	Olemas, annab hea ülevaate mida rakendus toetab kuid täpsemad kirjeldused puuduvad.
	A4. Rakenduse vastamine eel-määratud turvalisuse nõuetele	
Funktsionaalne nõue	Hinnang	Kommentaar
A4.1 Süstimis-tüüpi ründed	Jah	Nii HTML, SQL, Operatsioonisüsteemi käsud ja Path-Injection
A4.2 Sessiooni/Autentimist puudutavad ründed	Jah	Nii katkine autentimine, nõrgad paroolid kui ka sessiooni haldus
A4.3 Murdskriptimine	Jah	Mõlemad tüübid, nii peegeldatud kui salvestatud, Kokku 19 erinevat tehnilist viga
A4.4 Ebaturvalised otseviited objektidele	Jah	
A4.5 Tundlike andmete paljastamine	Jah	Nõrk SSL, BEAST/CRIME/BREACH, base64 encoding
A4.6 Nõrk krüptograafia	Puudub info	
A4.7 Võtmehaldus/salvestamine	Ei	

A4.8 Nõrk juhuarvude genereerimine(Krüptograafia, sessiooni võtmete jms puhul)	Puudub info	
A4.9 Puudulik õiguste kontroll	Jah	Kaustadele ja failidele otsene ligipääs ja viitamine
A4.10 Päringu murdvõltsimine	Jah	Konto üle võtmine võimalik
A4.11 Ebaturvalised ümbersuunamised	Jah	
	Muud kommentaarid	
	Rakendus tundub hea ja põhjalik, kuid puudub dokumentatsioon vigade kohta. Raha eest on võimalik tellida täpsem koolitus, aga hetkel tuleb seda ise uurida.	

Lisa2: Google Gruyere rakenduse hindamine

	A1. Rakenduse info	
Rakenduse nimi	Google Gruyere	
Rakenduse koduleht	https://google-gruyere.appspot.com	
Rakenduse tehnilised nõuded	1) Jookseb brauseris 2) Saab käivitada kliendi arvutis	
Rakenduses kasutatavad keeled/tehnoloogiad	Python, HTML, JS	
	A2. Rakenduse paigaldamine	
Kommentaar	Paigaldamine on äärmiselt lihtne, piisab vaid arhiivi alla laadimisest ning Pythoni faili käivitamisest. Vaja on ainult Pythonit.	
	A3. Dokumentatsiooni olemasolu(Ei, Jah, pole rakendatav, Puudub info)	
Dokument	Hinnang	Kommentaar
A3.1 Paigaldamist abistav dokumentatsioon	Jah	Olemas, lühike kuid piisav
A3.2 Rünnete/Vigade dokumentatsioon ja näited	Jah	Kogu dokumentatsioon rünnete kohta olemas
A3.3 Soovitatud paranduste dokumentatsioon	Jah	Paranduste dokumentatsioon olemas
A4.3 Soovitatud paranduste koodinäited	Jah/ei	Vähesel määral
A4.4 Turvavigade loetelu	Jah	Olemas, annab hea ülevaate mida rakendus toetab, koos kirjeldusega kuidas neid ära kasutada
	A4. Rakenduse vastamine eel-määratud turvalisuse nõuetele	
Funktsionaalne nõue	Hinnang	Kommentaar
A4.1 Süstimis-tüüpi ründed	Jah	Lubab süstida enda koodi veebiserverisse. Lubab süstida SQL koodi otse andmebaasi
A4.2 Sessiooni/Autentimist puudutavad ründed	Jah	Võimaldab tavakasutajal võtta üle administraatori õigused
A4.3 Murdskriptimine	Jah	Mitut tüüpi, nii salvestatud kui peegeldatud
A4.4 Ebaturvalised otseviited objektidele	Ei	
A4.5 Tundlike andmete paljastamine	Jah	*Path traversal* tüüpi rünne on võimalik
A4.6 Nõrk krüptograafia	Ei	
A4.7 Võtmehaldus/salvestamine	Ei	
A4.8 Nõrk juhuarvude genereerimine(Krüptograafia, sessiooni võtmete jms puhul)	Ei	

A4.9 Puudulik õiguste kontroll	Ei	
A4.10 Päringu murdvõltsimine	Jah	Konto üle võtmine võimalik
A4.11 Ebaturvalised ümbersuunamised	Ei	
	Muud kommentaarid	
	Lisaks on võimalik ka teenustõkke (DOS) rünne. Rakendus on lihtne, kiirelt paigaldatav, Äärmiselt hea dokumentatsiooniga. Ründeid on käsitletud põhjalikult ning pakutud on ka viisid kuidas neid ära hoida konkreetse rakenduse puhul. Samuti on toodud välja sammud ründe sooritamiseks. Kahjuks on aga katvus väga väike ning keskendutakse ainult neljale põhilisele ründele	

Lisa 3. OWASP Bricks hindamine.

	A1. Rakenduse info	
Rakenduse nimi	Owasp Bricks	
Rakenduse koduleht	http://sechow.com/bricks/	
Rakenduse tehnilised nõuded	1) PHP ja MySQL toega veebiserver	
Rakenduses kasutatavad keeled/tehnoloogiad	PHP, MySQL, JS, HTML	
	A2. Rakenduse paigaldamine	
Kommentaar	Rakenduse paigaldamine sujus, samme oli palju kuid manuaal oli piisav	
	A3. Dokumentatsiooni olemasolu(Ei, Jah, pole rakendatav, Puudub info)	
Dokument	Hinnang	Kommentaar
A3.1 Paigaldamist abistav dokumentatsioon	Jah	Olemas, põhjalik ning mitme platvormi jaoks
A3.2 Rünnete/Vigade dokumentatsioon ja näited	Jah	Kogu dokumentatsioon rünnete kohta olemas, samuti näited
A3.3 Soovitatud paranduste dokumentatsioon	Ei	
A4.3 Soovitatud paranduste koodinäited	Ei	
A4.4 Turvavigade loetelu	Jah	Olemas, annab hea ülevaate mida rakendus toetab, koos kirjeldusega kuidas neid ära kasutada
	A4. Rakenduse vastamine eel-määratud turvalisuse nõuetele	
Funktsionaalne nõue	Hinnang	Kommentaar
A4.1 Süstimis-tüüpi ründed	Jah	Lubab süstida SQL koodi otse andmebaasi Lubab süstida enda koodi veebirakendusse
A4.2 Sessiooni/Autentimist puudutavad ründed	Jah	Lubab ennast muuta teiseks kasutajaks
A4.3 Murdskriptimine	Ei	
A4.4 Ebaturvalised otseviited objektidele	Ei	
A4.5 Tundlike andmete paljastamine	Ei	
A4.6 Nõrk krüptograafia	Ei	
A4.7 Võtmehaldus/salvestamine	Ei	

A4.8 Nõrk juhuarvude genereerimine(Krüptograafia, sessiooni võtmete jms puhul)	Ei	
A4.9 Puudulik õiguste kontroll	Ei	
A4.10 Päringu murdvõltsimine	Ei	
A4.11 Ebaturvalised ümbersuunamised	Jah	
	Muud kommentaarid	
	Rakendus on hästi tehtud, dokumentatsioon sisaldab palju infot ja vihjeid. Paigaldamine sujus. Kahjuks on puudu soovitused, kuidas antud rakendust turvaliseks teha. Samuti keskendub rakendus väga põhjalikult erinevat tüüpi SQL süstimisrünnetele. Kümnest turvaveast on kaheksa erinevad süstimisründed SQL pihta.	

Lisa4 Owasp Mutillidae 2 hindamine

	A1. Rakenduse info	
Rakenduse nimi	OWASP Mutillidae 2	
Rakenduse koduleht	https://sourceforge.net/projects/mutillidae	
Rakenduse tehnilised nõuded	1) PHP ja MySQL toega veebiserver 2) Toetab kõiki enam-levinud operatsioonisüsteeme	
Rakenduses kasutatavad keeled/tehnoloogiad	PHP, MySQL, JS, HTML	
	A2. Rakenduse paigaldamine	
Kommentaar	Rakenduse paigaldamine sujus. Abiks oli video kujul juhend. Samas on rakendus päris keeruline, rünnete sooritamine eeldab ka spetsiaalseid tööriistu ning tundub, et rakendus on mõeldud kasutajale kes juba orienteerub pigem hästi veebi turvalisuse valdkonnas.	
	A3. Dokumentatsiooni olemasolu(Ei, Jah, pole rakendatav, Puudub info)	
Dokument	Hinnang	Kommentaar
A3.1 Paigaldamist abistav dokumentatsioon	Jah	Olemas, nii kirjalik kui video
A3.2 Rünnete/Vigade dokumentatsioon ja näited	Jah	Kogu dokumentatsioon rünnete kohta olemas, samuti näited
A3.3 Soovitatud paranduste dokumentatsioon	Ei	
A4.3 Soovitatud paranduste koodinäited	Ei	
A4.4 Turvavigade loetelu	Jah	Loetelu on täielik, juhendid osalised
	A4. Rakenduse vastamine eel-määratud turvalisuse nõuetele	
Funktsionaalne nõue	Hinnang	Kommentaar
A4.1 Süstimis-tüüpi ründed	Jah	Lubab süstida SQL koodi otse andmebaasi
A4.2 Sessiooni/Autentimist puudutavad ründed	Jah	
A4.3 Murdskriptimine	Jah	Toetab mitut erinevat tüüpi murdskriptimist ning dokumentatsioon on hea
A4.4 Ebaturvalised otseviited objektidele	Jah	
A4.5 Tundlike andmete paljastamine	Jah	
A4.6 Nõrk krüptograafia	Ei	
A4.7 Võtmehaldus/salvestamine	Ei	

A4.8 Nõrk juhuarvude genereerimine(Krüptograafia, sessiooni võtmete jms puhul)	Ei	
A4.9 Puudulik õiguste kontroll	Jah	
A4.10 Päringu murdvõltsimine	Jah	
A4.11 Ebaturvalised ümbersuunamised	Jah	
	Muud kommentaarid	
	Rakendus on hästi tehtud, dokumentatsioon sisaldab palju infot ja vihjeid. Peaaegu iga ründe kohta on olemas ka video. Ründed on uuendatud ning ajakohased ja projekt on aktiivselt arenduses. Samas puuduvad näited kuidas teha asju õigesti ning soovitused kuidas antud vead	