Tallinna Ülikool

Informaatika Instituut

SISUHALDUSSÜSTEEMIDE TURVALISUS HARJUMAA ÜLDHARIDUSKOOLIDE VEEBILEHTEDE NÄITEL

Bakalaureusetöö

Autor: Mikk Lilles

Juhendaja: Meelis Karp

Autor: ..... „ ..... „2015
Juhendaja: ..... „ ..... „2015
Instituudi direktor: ..... „ ..... „2015

Tallinn 2015

Autorideklaratsioon:

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)

TALLINNA ÜLIKOOL

Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks

Mina _____ (sünnikuupäev: _____)
(*autori nimi*)

1. annan Tallinna Ülikoolile tasuta loa (lihtlitsentsi) enda loodud teose
   _____
   _____
   _____
   (*lõputöö pealkiri*)

   mille juhendaja on _____,
   (*juhendaja nimi*)

   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, _____
(digitaalne) allkiri ja kuupäev

Sisukord

Sissejuhatus

Tänapäeval on peaaegu kõik veebilehestikud dünaamilised, kasutavad andmebaase ja sisuhaldussüsteeme (CMS ehk *Content Management System*). Sisuhaldustarkvara valitakse peamiselt pakutavate võimaluste järgi, tihtipeale saab määravaks hind. Tarkvarade kirjeldustes rõhutatakse enamasti erilisi lisavõimalusi ning kasutamise mugavust ja lihtsust. Samas ei käsitleta peaaegu kunagi nende süsteemide turvalisust, mis võiks olla valiku tegemisel üheks kõige olulisemaks aspektiks.

Käesoleva bakalaureusetöö teema on valitud autori huvist veebilahenduste ja sisuhaldustarkvarade vastu. Teemat valides tutvus autor varem kirjutatud töödega ning avastas, et sisuhaldustarkvarade turvalisuse temaatikat ei ole varem lõputöödes eriti käsitletud. Arvestades seda, et aina enam veebilehti ehitatakse üles sisuhaldustarkvaradele, on teema vägagi aktuaalne ning vajalik nii arendajatele, veebitoimetajatele kui ka teistele teema vastu huvi tundjatele. Koolide kodulehekülgede turvalisusele keskendutakse sellepärast, et nende rünnakud on viimasel ajal meedias palju kajastust leidnud. Töö tulemused on lisaks teistele huvilistele suunatud eelkõige koolide veebilehtedega tegelejatele – nii riiklikul kui eraldi koolide tasandil.

Bakalaureusetöö eesmärk on kaardistada Harjumaa üldhariduskoolide veebilehtede turvalisuse hetkeseis ning võimaluse korral pakkuda soovitusi selle parandamiseks.

Peatükkides 1, 2.1, 5, 5.1, 5.2, 5.3, 5.4 ja 7 kasutab autor materjale enda seminaritööst „Populaarsete veebi sisuhaldussüsteemide turvalisus“.

Töö koosneb seitsmest sisupeatükist, mis jagunevad alapeatükkideks:

- Esimeses peatükis tuuakse välja töös kasutatavad mõisted ja lühendid.
- Teises peatükis kaardistatakse Eesti veebide turvalisuse hetkeolukord, kasutades meediast läbi käinud juhtumeid, ning analüüsitakse rünnakustatistikat Zone-h andmebaasi toel.
- Kolmandas peatükis tehakse lühike ülevaade teemaga seotud riiklikest plaanidest ja dokumentidest, et selgitada välja, mida tehakse koolide kodulehekülgede turvalisuse tagamiseks riiklikul tasandil.

- Töö neljas peatükk keskendub Harjumaa koolide veebilehtede turvalisusele. Nende veebilehtedega tegelejate seas viiakse läbi küsitlus, et analüüsida turvalisuse hetkeseisu ja töötajate turvateadlikkust. Uuritakse, kas IKT programmi raames plaanitud standardimissoov on vastavuses koolitöötajate nägemusega.
- Viiendas peatükis tehakse lühike ülevaade Harjumaa koolide seas kõige populaarsemaks osutunud CMS-ide turvalisusest.
- Koolidega kokkuleppel uuritakse manuaal- ja automaattestidega kolme väljavalitud kooli veebilehe turvalisust, püüdes leida nende turvaauke ja nõrku kohti. Meetodit ja tulemust kirjeldatakse kuuendas peatükis.
- Seitsmes peatükk põhineb arutelul, mille käigus antakse üldisi soovitusi turvariskide vähendamiseks.

Sisupeatükkidele järgnevad kokkuvõte, ingliskeelne resümee, kasutatud kirjandus ja lisad. Lugemise lihtsustamiseks lisatakse joonised tekstilõikude vahele. Viited on lisatud Zoteroga.

1. Töös kasutatavad mõisted ja lühendid

CVE *(Common Vulnerabilities and Exposures)* – keskkond, kus jagatakse informatsiooni ja statistikat levinud turvaaukude kohta

Hajutatud teenusetõkestamise rünne *(Distributed Denial of Service ehk DDoS)* – kui arvuti või arvutivõrgu ülekoormamiseks kasutatakse mitmeid arvuteid ja nende kasutajaid

HTTP vastuse poolitamine *(HTTP responce splitting)* – veebirakendus või keskkond ei suuda korralikult kontrollida sisendväärtusi. Ründaja saab määrata suvalisi päiseid, kontrollida vastuse keha osa või moodustada mitu eraldi HTTP vastust

Informatsiooni kättesaamine *(gain information)* – privaatsetele andmetele ligipääsemine kolmandate isikute poolt

Kataloogitee läbimine *(directory traversal)* – ründaja käsib veebirakendusel avada faili, mis peaks olema juurdepääsmatu ning mille ligipääsuks läbitakse kataloogipuu

Kujundusteema *(theme)* **–** sisuhaldustarkvaradele mõeldud kujunduspakk

Lubamatu faili lisamine *(file inclusion)* – ründaja lisab veebiserveris käivitatud koodi abil faile, nt *include directive*'iga

Lubamatu kasutajaõiguste eskaleerimine *(gain privilege, privilege escalation)* – ründaja saab kasutajaga serveris samad õigused

Läbistustestimine *(penetration testing)* – süsteemi nõrkuste tuvastamine ning tõestamine heatahtlikul eesmärgil

Murdskriptimine *(cross-site scripting, XSS)* **–** dünaamiliselt genereeritavate veebilehtede kasutamine võõrastesse arvutitesse tungimiseks

Mööda pääsema *(bypass something)* – ründaja pääseb mööda kasutaja poolt määratud turvatõketest, näiteks autoriseeringu päringust

Nõrkus *(vulnerability)* – ohtude realiseerimist võimaldav infovarade nõrk koht

Näotustamine *(defacement)* – veebilehele soovimatu sisu ülesriputamine või veebilehe sisu asendamine võõra isiku poolt

Omavoliline koodi käivitamine *(execute code)* – ründaja võimalus käivitada meelevaldset koodi sihtarvutis või sihtprotsessis

OWASP *(The Open Web Application Security Project)* – avaliku veebi rakenduste turvalisuse projekt

Programmiviga *(bug)*

Ristpäringuvõltsing *(cross-site request forgery, CSRF)* – ründaja saab ära kasutada veebisaidile usaldatud õigusi veebilehitseja poolt

Sisuhaldustarkvara *(content management system, CMS)* **–** tarkvara, mis haldab dokumente veebilehtede tarvis

Sotsiaalne manipulatsioon *(social engineering)* – tegevus, mille eesmärk on saada ligipääs arvutisüsteemile, manipuleerides kasutajat avalikustama konfidentsiaalset teavet nagu paroolid või võtma vastu vastavat pahavara

Süstimine *(injection)* – valideerimata info sisestamine kolmandate isikute poolt kasutaja serveris asuvatele failidele

Tagauks *(backdoor)* **–** serverisse jäetud või eraldi tehtud süsteemi sissepääsu võimalus, mis jätab jätkuva ründe võimaluse. See on ohtlik ka selle pärast, et tarkvara uuendused ei eemalda neid faile ja neid on raske eristada. Tihti on see PHP-fail.

2. Ründed veebilehestike vastu

Selles peatükis antakse ülevaade tüüpilistest rünnakuviisidest veebilehestike vastu ja tuuakse Eesti ja välismaa veebilehtede põhjal näiteid, kuidas neid rünnakuviise kasutatud on. Analüüsitakse rünnakustatistikat Zone-h andmebaasi põhjal.

2.1. Levinud rünnakumeetodid

Igasuguse tarkvaraga võivad kaasneda turvariskid. Selles bakalaureusetöös uuritakse Harjumaa üldhariduskoolide näitel, mis on populaarsete sisuhaldussüsteemide kõige enam levinud turvariskid, missuguseid rünnakuid on nendele vabavaralistele sisuhaldustarkvaradele tehtud ja mis võimalusi on rünnakute vältimiseks. Järgnevalt annab autor ülevaate tüüpilistest turvariskidest, mis võivad eri CMS-idel esineda.

Veebilehtede vastu suunatud rünnakumeetodeid on väga erinevaid ja tulemus sõltub alati eesmärgist. Rünnaku eesmärgiks võib olla lihtsalt oma pingete maandamine kellegi veebilehte rikkudes või tuntuse kogumine sellega tegelevate inimeste ringis. Võib juhtuda, et sellised ründed ei ole väga tõsiste tagajärgedega ja piirdub ainult peidetud informatsiooni mingis mahus lugemisega. Väga tihti on aga tagajärjed tõsised ning ettearvamatud, mille üks näide on ründaja poolt sisestatud teksti väljakuvamine kasutaja veebisaidil. Väga ohtlike tagajärgedega võivad olla ka sellised rünnakud, mille eesmärgiks on isikuandmete kogumine – näiteks kontaktandmete müümine ettevõtetele, kes tegelevad massreklaamide saatmisega või telefonimüügiga. Kõige rängemad tagajärjed võivad olla aga sellisel juhul, kui näiteks pääsetakse ligi inimeste internetipanga paroolidele või väga isiklikele andmetele. Näiteks portaal www.eesti.ee peab olema eriti hästi turvatud, sest see sisaldab kõigi kodanike andmeid haigekassa, töösuhte, hariduse ja muu kohta. Samuti on väga ohtlikud sellised rünnakud, mille läbi halvatakse terve laialt kasutatava süsteemi töö, näiteks häiritakse internetipanga või kaarditehingute toimimist.

Veebirakenduste turvalisuse teemal püüab tõsta inimeste teadlikkust OWASP-i programm „Top Ten“ (OWASP 2013a). Sihtasutus OWASP (The Open Web Application Security Project – avaliku veebi rakenduste turvalisuse projekt) alustas tööd 2001. aastal. See on mittetulunduslik rahvusvaheline organisatsioon, mis tegutseb annetuste toel. Nad on määratlenud kõige kriitilisemad turvaprobleemid. (OWASP 2013a)

Sihtasutuse OWASP hinnangul on CMS-ide kõige kriitilisemad nõrkused ja nende ärakasutamise viisid järgmised:

- **Süstimine** (*Injection*)

  Kõige tavalisem süstimise vorm on SQL-süstimine. See on rünnak andmebaasikihi tasandil, mis seisneb SQL-päringusse omavolilise informatsiooni sisestamisel (Lehesalu 2007). Ründajad kasutavad seda väga tihti ja see on üks kõige ohtlikumaid ründemeetodeid, mida saab veebilehtede vastu sooritada.

- **Vigane autentimine ja sessiooni haldamine** (*Broken Authentication and Session Management*)

  Autentimise ja sessiooni haldamise alla käivad kõik autentimise ja aktiivsete sessioonidega seotud toimingud. Autentimine on väga oluline protsess ja kui selle toimingud – näiteks parooli vahetus, parooli unustamise korral uuesti saatmine, parooli meeldejätmine, konto uuendus ja teised sarnased tegevused – on vigased, siis saab seda väga tõsiste tagajärgedega ära kasutada (OWASP 2013b).

- **Murdskriptimine** (*Cross-Site Scripting* ehk XSS)

  Selle tehnika alus seisneb ühe serverarvuti mõjutamises, mille tulemusena server edastab oma kasutajale ründaja etteantud sõnumi. Sõnum võib olla nii kuuldelis-vaateline kui ka peidetud instruktsioon, mille käigus kasutaja arvuti teeb midagi ettearvamatut. (Wikipedia 2013)

- **Turvamata otseviide objektile** (*Insecure Direct Object References)*

  See on rünnakumeetod, mis tähendab seda, et viide mingile objektile on valideerimata ja ründaja saab sisestada endale meelepärase info. Näiteks väga levinud on URL-i muutmine, kus kasutaja andmete viide suunab hoopis teisele lehele. (OWASP 2013d)

- **Vigane turvakonfiguratsioon** (*Security Misconfiguration*)

  Selle nõrkuse väga hea näide on see, kui serveris installitakse automaatselt rakenduse administreerimise liides ja kasutaja ei kustuta seda installimise lõppedes ära. Ründaja avastab, et serveris on alles üldadministreerimise failid, millele saab ligi vaikimisi määratud paroolidega ning nüüd tekib ründajal võimalus serverit kontrollida (OWASP 2013f).

- **Privaatandmete kättesaadavus** (*Sensitive Data Exposure*)

Tavaliselt salvestatakse IT-süsteemides kasutajate andmed, näiteks salasõnad, krediitkaardi andmed, aadress jm, andmebaasidesse. Kui see süsteem ei ole piisavalt kaitstud, siis on suur tõenäosus, et keegi saaks seda pahatahtlikult ära kasutada ja selle informatsiooni varastada.

- **Puudulik kasutajaõiguste reguleerimine** (*Missing Function Level Access Control*)

  Tarkvarad ei kaitse alati oma funktsioone korralikult. Mõnikord määratakse funktsiooni kaitse konfiguratsioonis, mille seaded on aga valed. Arendajad peaksid lisama sobiva koodi, et kontrollida seadeid, aga unustavad tihti seda teha (OWASP 2013g).

- **Ristpäringuvõltsing** *(Cross-Site Request Forgery* ehk CSRF)

  See on rünne, kus lõppkasutaja sunnitakse käivitama soovimatuid programme, kui kasutaja on autenditud. Vähese abiga (saates lingi e-postile või vestlusesse) saab ründaja kasutaja ära petta, et ta avaks ja käivitaks programme, mida pahalane soovib. Kui lõppkasutajal on administraatori õigused, siis võib ohus olla terve veebitarkvara (OWASP 2013c).

- **Tuntud nõrkade tarkvarakomponentide ärakasutamine** (*Using Known Vulnerable Components*)

  Põhimõtteliselt on kõikidel rakendustel selline oht, sest enamik arendajad ei hoia oma komponente uuendatuna. Väga tihti arendajad ei teagi, mis komponente nad kasutavad ja kui need on omavahel sõltuvuses, siis see teeb asja hullemaks (OWASP 2013h).

- **Valideerimata ümber- ja edasisuunamine** (*Unvalidated Redirects and Forwards*)

  Väga tihti suunab tarkvara kasutaja teistele veebilehtedele või kasutavad siseviiteid sarnasel kujul. Mõnikord see veebileht, kuhu kasutaja suunatakse, on täpsustatud valideerimata parameetriga, mis lubab ründajal valida lehe, kuhu kasutaja suunatakse. Selliste ümbersuunamiste tulemusel võidakse proovida installida pahavara või üritatakse kasutajat panna sisestama salasõnu või muud privaatset informatsiooni.
  (OWASP 2013a)

Kõik OWASP-i poolt välja toodud kümme rünnakutüüpi on väga sagedased ja tihti, kui kasutajad leiavad viisi selliste rünnakute kaitseks, arendavad häkkerid välja uued meetodid, kuidas neid kaitsemehhanisme nõrgestada.

Samuti on väga levinud turvaohuks omavoliline koodi käivitamine, kuigi seda 2013. aasta OWASP-i aruandes ei olnud. See on väga ohtlik programmiviga, kuna see võimaldab ründajal üle võtta terve vigase protsessi. Sellest edasi saab ründaja üle võtta masina, mille peal see protsess töötab. Kuna see on üks kõige tulemuslikumatest rünnakumeetoditest pahatahtlikel eesmärkidel, siis peaks selle ennetamine olema väga tähtis ülesanne veebilehe administraatorite poolt.

Teenusetõkestamise rünnet kasutatakse väga tihti häkkerite ja teiste ründajate poolt, et päringutega ülekoormata võrke ja veebilehti. Tavalise teenustõkestamise rünnaku alla käib ka lihtne veebilehe korduv värskendamine. Näiteks on oht, et kui kasutaja tahab e-poes näha korraga 1000 toodet, siis pärast korduvat lehe värskendamist koormatakse andmebaas üle ning veebileht muutub mingiks ajaks kättesaamatuks. Hajutatud teenusetõkestamise ründe puhul kasutatakse tihti robotvõrku, mida teatakse kui botnette. See tuleb sõnadest *robot* ja *network* (Arvutikaitse 2015). Eesti keeles kasutatakse vahel ka sõna zombiarmee. See võrgustik koosneb suurest hulgast hõivatud arvutitest ehk zombidest, mida omaniku teadmata kasutatakse DDoS-rünnete korraldamiseks või rämpsposti levitamiseks. Selliste rünnakute taga liiguvad väga suured rahad, sest häkkerid, kes on saanud enda valdusesse arvuteid koos kasutajatega, müüvad nende arvutite nimekirjad edasi. Selliseid rünnakuid toimub iga hetk ning põhjalikku illustreeritud kaarti näeb aadressilt www.digitalattackmap.com.

1999. aastal loodi CVE (Common Vulnerabilities and Exposures), mille eesmärk on jagada informatsiooni turvalisuse nõrkustest ja privaatsete andmete pahatahtliku kättesaadavuse kohta. CVE kogub ja kategoriseerib info selle valdkonna andmebaasidest ja kuvab andmed süstematiseeritud kujul kasutajatele. Uus sissekanne algab potentsiaalse turvaaugu või süsteemi nõrkuse leidmisega. Omistatud informatsioonile antakse CVE identifikaator ja pannakse veebilehele üles. (CVE 2013b) CVE kodulehelt saab väga täpset ja aktuaalset infot sisuhaldustarkvarade turvaaukude ja nõrkuste kohta.

2.2. Haridussüsteemiga seotud veebilehtede rünnakud meedias

Eesti veebilehtede vastu toimub rünnakuid nagu kõikjal mujal maailmas ning ründemeetodid on üldjuhul just need, mida eelmises alapeatükis kirjeldati. Viimasel ajal on rünnakud ka meedias palju kõlapinda leidnud. Põhjus, miks just riigi- ja kohalike

omavalitsusasutuste veebilehed on levinud sihtmärgiks, võib olla selles, et rünnakute sõnumid on tihti poliitilised ja teatakse, et riigiasutuse lehele postitatud sõnum levib meedias paremini.

Riigi- ja kohalike omavalitusasutuste leheküljed võivad saada rünnete sihtmärgiks näiteks väheste finantsvõimaluste tõttu – puudub võimalus palgata kompetentset IT-spetsialisti, kes oskaks turvaprobleeme ennetada, ja kasutatakse vabavaralist tarkvara. Näiteks Ivari Horm kaitses 2012. aastal Tallinna Tehnikaülikoolis bakalaureusetöö „Vaba tarkvara kasutamine võrgu teenuste osutamisel Tallinna munitsipaalkoolides“. Tema uurimusest selgub, et enamik koole kasutavad koduleheküIgede haldamiseks tõesti just vabavaralist tarkvara. Hormi (2012, 41) sõnul „61% ehk 19 küsitluses osalenud kooli kinnitas, et nende koduleheküIg kasutab vabal tarkvaral põhinevat administreerimiskeskkonda (WordPress, Joomla!, Drupal vms)“.

2014. aastal leidis aset mitu küberrünnakut, mille sihtmärgiks oli e-kool (Himma 2014) (Roon 2014). 2010. aastal said tundmatud pahalased interneti kaudu ligipääsu Narva Kesklinna gümnaasiumi serveritele, mille kaudu suunati ringi ligi 3700 Miami (USA) ja Kuuba telefonikasutajate kõnet. Gümnaasiumile esitati 374 000 krooni suurune arve (Tokareva 2010). 2006. aastal ründasid häkkerid korraga paljude edu.ee võrgus olevate Eesti koolide ja haridusasutuste kodulehti, mistõttu paljud neist ei avanenud või nende esilehel seisis häkkerite poolt jäetud teade: „We Want The Peace. Stop The War“ (Postimees 2006). Koolidele suunatud küberrünnakute näiteid on palju ka välismaalt. 2015. aasta alguses (Matthew 2015) häkiti näiteks Ühendkuningriigis Yorkshire'is Sowerby kooli koduleheküljele, kuhu X-saadi nimeline rühmitus postitas poliitilise sõnumi Ameerika Ühendriikide ja Iisraeli kohta. 11. jaanuaril 2015 häkiti Floridas asuva Oakleafi gümnaasiumi kodulehele, kuhu ennast Zulu Squadiks nimetav rühmitus postitas Guy Fawkesi maski kujutise ning lehe looja programmeerimisoskust kritiseeriva teate (Actionnews 2015).

Kui võiks arvata, et enamasti tulevad rünnakud pigem väljastpoolt asutust, kas siis poliitilise või muu motivatsiooniga, siis on ka juhtumeid, kui koolide koduleheküIgi ründavad just sama kooli õpilased. Näiteks 2015. aasta 6. jaanuaril (Mersereau 2015) häkkis Ameerika Ühendriikides Virginia osariigis arvamuste kohaselt ärritunud õpilane sisse kooli kodulehele, olles pettunud kooli otsuses alustada tunde õigeaegselt vaatamata keerulistele teeoludele. Maakonna kooli koduleheküIje avalehele postitati ebatsensuurne

teade. Raskekujulisem näide on Ameerika Ühendriikide Illinois' osariigist, kus 2014. aasta lõpus häkkisid kaks õpilast ühe koolitöötaja e-posti ja õpikeskkonda U-46, kus nad muutsid enda kohta tundides osalemise arvestust. Muu hulgas oli neil võimalus näha kaasõpilaste isikuandmeid (Smith 2014). Hiinas (Groom 2014) häkkis eelmisel sügisel kooli õpikeskkonda kõigest 13-aastane õpilane.

Kuna Eestis püüeldakse selle poole, et programmeerimist õpetataks juba algkoolis (Roosaar 2012), siis peab arvestama, et õpilaste oskused selles vallas hakkavad arenema aina varasemast east. Nende oskustega peavad kaasas käima ka koolide infosüsteemid.

2.3. Rünnakustatistika analüüs

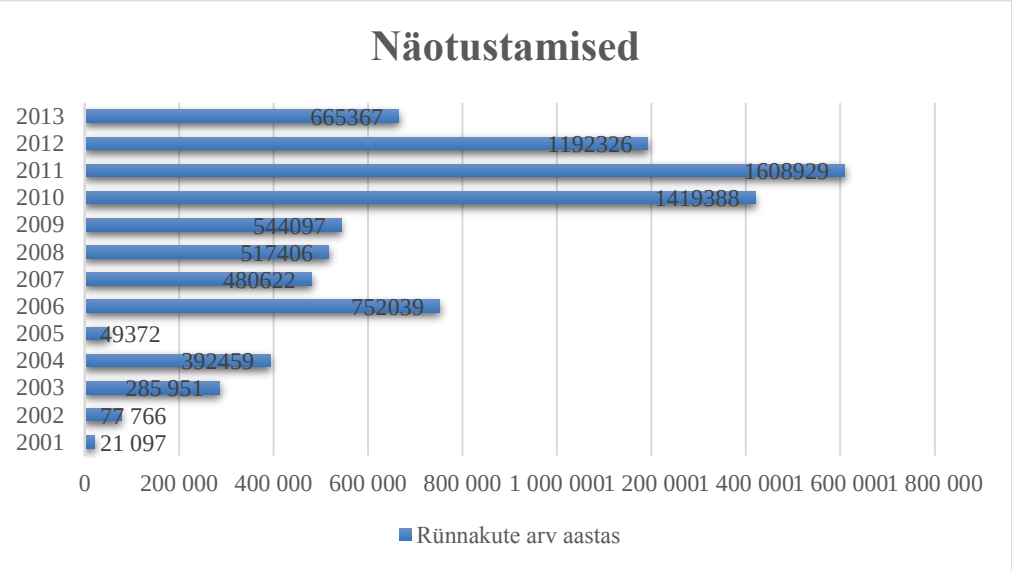
Üks väga tihti kasutatav rünnaku tüüp on näotustamine. Näotustamise eesmärk võib olla poliitiline, äriline või paljudel juhtudel lihtsalt oma võimete demonstreerimine teistele sarnastele ründajatele või näotustajate grupeeringutele. Sageli on ründajateks isikud, kelle kohta kasutatakse inglise keeles väljendit *script kiddie.* Tavaliselt nad kasutavad rünnakuks kellegi teise loodud koodi või programmi ning serverisse installitud tagaust. IT-turvalisuse eestvedaja Rootsis, David Jacoby (2010) räägib intervjuus, mis ta andis ajakirjale Securelist, tööriistadest ja meetoditest, mida näotustajad kasutavad rünnakuteks. Ta ütleb, et näotustamisi korraldavad tavaliselt inimesed, keda ei huvita, mis veebilehte nad näotustavad. Nende peamine eesmärk on leida serveris turvaauk ja siis seda ära kasutada, kas veebilehe sisu muutmise näol või faili üles laadimise teel, kus ründaja teeb ennast teatavaks. Ta toob ka välja, et kuigi meedia nimetab selliseid ründajaid häkkeriteks, siis tegelikult võiks neid kahte terminit ikkagi eristada. Häkker ei ründa üldjuhul suvalisi saite ja enamasti on häkkerite rünnakuobjektiks konkreetne veebisait. Samuti eelistab häkker tegutseda rünnatud süsteemis märkamatult ning võimalusel kustutada kõik jäljed seal käigust. Ründajad kasutavad sageli ka *google dorks*'i nimelist meetodit, et leida turvaaukudega serverid ja veebilehed. Selleks kasutatakse erilisi otsingusõnastusi, et leida turvaaukudega programme või pluginaid. Näotustamistest on kujunenud võistlus erinevate gruppide vahel, kes saavad tänu näotustamiste arhiividele näha rünnakustatistikat. Üks suurimaid selliseid arhiive on Zone-h.

Lehekülg http://zone-h.org/ koondab veebilehtede rahvusvahelist rünnakustatistikat ja hoiab endas arhiivi veebilehtede näotustamise kohta. Zone-h kuulub ettevõttele Security Lab ja see asutati sooviga koguda turvaintsidente mitteametlikest allikatest, et avalikkus

saaks paremat ülevaadet toimunud rünnakutest. Zone-h asutati aastal 2001. H-täht tähistab sõna häkkerid või häkkimine. Algselt oli plaanis peale intsidentide arhiivi luua ka foorumilaadne suhtluskanal, aga ajapikku kogunes sinna liiga palju muud kõrvalist vestlust ja asutajad otsustasid jätta saidi ainult statistilist informatsiooni kajastavaks kanaliks (Kovacs 2013).

Näotustamine jõuab Zone-h lehele postitajate kaudu, pärast mida nende oma automaatne robot salvestab näotustatud veebilehe Zone-h andmebaasi koos peegeldusega rünnakust. Zone-h töötajad käivad kõik salvestised manuaalselt läbi ja kinnitavad nende andmete õigsust. Security Labi omaniku Alberto Redi sõnul on kaks kolmandikku näotustamiste postituskatsetest võltsid. Rünnaku ohvriks langenud ettevõtted võtavad väga tihti nendega ühendust ja paluvad rünnatud kodulehe raporti Zone-h-st maha võtta, aga arhiveerijad omakorda selgitavad, et selle avalikustamise eesmärk ongi veebilehtede omanikele teada anda, et nad peavad tegelema oma saitide turvalisusega. (Kovacs 2013)

Redi toob artiklis välja näotustamiste koguarvu aastate lõikes (Joonis 1):



Joonis 1. Näotustamiste arv aastate lõikes

Kuna artikkel kirjutati 8. juunil 2013, siis ei ole viimase aasta info täielik.

Zone-h jagab näotustamised kahte kategooriasse – kas toimub avalehe või paljude lehtede näotustamine samas serveris. Üks näide näotustamisest on läbi SQL-süstimise. Häkker saab serveris administraatori õigused ja ligipääsu failidele. Järgmiseks saab paari koodireaga vahetada `index.html`-i häkkeri poolt valitud failiga. Sageli kasutavad näotustajad rünnakuteks teadaolevaid sisuhaldustarkvarade ja nende pluginate turvaauke.

Nagu ka ühes näites peatükis 6.4, kasutati tuntud WordPressi plugina turvaauku, millega saadi kätte andmebaasi infot sisaldav konfiguratsioonifail. Sellega oleks häkkeril täielik ligipääs kõikidele veebilehe failidele. Häkkerid saavad *google dorks*'i meetodiga leida veebilehed, kus see konkreetne moodul kasutusel on ja vastava skriptiga teha need päringud ja serveris asuvate failide muutmine automaatseks.

Tuntumate rünnatud .ee domeenide seast võib Zone-h lehelt leida näiteks `www.kaspersky.ee`, `www.mcdonalds.ee`, `www.eestipoksiliit.ee`, `www.peugeot.ee`, `www.hyundai.ee` ja `www.panasonic.ee`. 24. märtsi seisuga on sellel aastal näotustatud 118 .ee domeenil asuvat lehte.

Alates aastast 2006 on rünnatud .edu.ee domeene 150 korral. `www.varbla.edu.ee` veebilehte on Zone-h järgi näotustatud kahel korral. Esimene kord 2006. ja teine kord 2014. aastal, millest viimasel juhul lisati failisüsteemi PHP-fail, mida kuvati avalehel. Kahel korral on näotustatud ka `koeru.edu.ee` veebilehte.

Sõltumata domeenist, leiti 7 Eesti haridussüsteemiga seotud veebilehte, mida on 2015. aasta jooksul 24. märtsi seisuga näotustatud. Need lehed on järgmised:

- 24. märtsil näotustati Tallinna Tõnismäe Reaalkoolile kuuluv õpilasteesinduste liidu koduleht. Veebilehele sisestati teade: „Hacked by NG689Skw". Ründe kanaliks kasutati Slider Revolutioni nimelist pluginat, mille vanemates versioonides oli ohtlik turvaauk sees.
- Sama mooduli turvaauguga näotustati ka Are Põhikooli koduleht. 18. märtsil 2015 rünnati hariduse virtuaalkeskkonna HAVIKEse demolehte `yexaka7.havike.eenet.ee/joomla/`. See veebileht on 29. märtsi seisuga maha võetud.
- 2. märtsil rünnati Tallinna Koolitervishoid SA veebilehe aadressi `www.kth.ee/?page=news&id=162`.
- 28. veebruaril rünnati munitsipaalhuvikooli Narva Noorte Meremeeste Klubi veebilehte, kus piltide kausta lisati häkkerite nimeline GIF-fail aadressile `http://www.nnmk.ee/images/jdownloads/screenshots/zxcvbnm.gif`, ning veebilehe algse index-faili asemel käivitati ründaja poolt lisatud fail, mis kuvas veebilehe külastajatele eelnimetatud GIF-faili.

- 26. jaanuaril rünnati Tallinna Tondi Põhikooli veebilehte ja näotustamiseks kasutatud pildi fail on veebilehe failisüsteemis veel alles 29. märtsi seisuga. See näitab ka seda, et kui veebilehtede haldajad avastavad rünnakud ja eemaldav failisüsteemist failid, mis kuvavad võõrast sisu, siis ülejäänud failid, mille häkker on failisüsteemi lisanud võivad jääda märkamatuks. Kui üks selline fail on näiteks mõni PHP-skript, mis on lisatud toimimaks tagauksena, siis on suur turvaauk veebilehel veel alles.
- 18. jaanuaril rünnati õppeprogrammi „Kaks keelt – üks meel“ veebilehte aadressil `www.2k1m.ee/images/nyet.gif`.
- 12. jaanuaril rünnati Tallinna lasteaia Mooniõied veebilehte `http://moonioied.ee/`.

Selle statistika järgi oli .ee domeenil asuvatest näotustatud veebilehtedest iga 16. haridussüsteemiga seotud leht. Kuna kõiki sel aastal rünnatud lehti pole veel korda saadud ja puudub nende avamise võimalus, et teha kindlaks, kas tegemist on koolidega seotud veebilehega, siis võib rünnatud koole olla rohkemgi.

Arvestades meediakajastust ja rünnakute statistikat, siis on ilmne, et haridusasutuste veebilehtede turvalisus on vägagi aktuaalne teema. Võib arvata, et näotustamine on kõige sagedasem koolide veebilehtede ründamise eesmärk, sest rahalist kasu on väga ebatõenäoline saada (vrd e-poe ründamisel) ja ka tundlike andmete otsimine ei oleks koolide puhul põhjendatud. Näotustamise teeb väga lihtsaks see, et koolide väheste võimaluste tõttu, on nende veebilehtedel palju turvaauke.

3. Riiklik tegevus haridusasutuste IKT küsimustes

Eesti põhikooli- ja gümnaasiumiseadus (§ 55, § 67, § 69) näeb ette, et koolid peavad vastuvõtutingimused, arengukava, kodukorra jm info avaldama veebilehel (Riigi Teataja 2010). Samas, sellest dokumendist ei selgu veebilehe nõudeid. Küll aga Tallinna Haridusametis (2010b) kinnitatud „Tallinna munitsipaalharidusasutuste infotehnoloogilise keskkonna programmis aastateks 2011–2015“ tuuakse välja, et kõik seda soovivad haridusasutused võiksid saada lähitulevikus kasutada „tsentraalselt administreeritud ja majutatud, kehtestatud nõuetele ja kokkulepitud funktsionaalsusele vastavalt eelkonfigureeritud ning lihtsalt käsitletava sisuhaldussüsteemiga varustatud koduleheküljе, mille sisu haldamine oleks jõukohane kõigile haridusasutuste töötajatele“. Selle eesmärgi saavutamiseks nähakse ette, et haridusamet koostab haridusasutuste koduleheküljе standardinõuded ning tagab ja finantseerib haridusasutustele tsentraalselt administreeritava ja majutatava sisuhaldustarkvaraga varustatud koduleheküljе kasutamise võimaluse (Tallinna Haridusamet 2010b, 7). IKT programmi tegevuskava (Tallinna Haridusamet 2010b, 11–12) järgi oli koduleheküljе standardi koostamise tähtaeg 2011. aastal. Standardile vastava koduleheküljе kasutamise ja majutamise võimalust peaks tegevuskava järgi 2014. aastaks kasutama 125 haridusasutust.

Töö autor saatis programmi esimehele päringu selle kohta, kuidas on arenenud tegevuskavas esitatud plaanid koostada koduleheküljeе standard ja tagada standardile vastava koduleheküljе kasutamise ja majutamise võimalus. Viivi Lokk vastas, et nad on korraldanud veebileheküljeе ülevaatusi, kuid lisaks sellele on ka kehtestatud nõuded ameti juhtaja käskkirjaga. See Tallinna koolide koduleheküljeе vormistamise juhend (Tallinna Haridusamet 2012) kehtestab nõuded vaid info kajastamisele, kuid mitte tehnilisele poolele. Vormistamisjuhend sätestab, et koolid peavad veebilehel esitama nädala sööklamenüü ja tugiteenuste loetelu jne. Programmis ära märgitud standardi koostamise ja kodulehtede majutamise kohta infot saada ei õnnestunud.

IKT nõukogu koosoleku 2010. aasta 15. detsembri protokollist nr 1.-4/138 (Tallinna Haridusamet 2010a), võib lugeda, et külalisena esinenud Elar Lang avaldas muret avaliku sektori veebilehtede turvalisuse pärast: „Selleks, et teada, kuidas oma süsteeme kaitsta, tuleb teada, kuidas rünnatakse. Kui turvalised on süsteemid, kus oma andmeid hoiame?

Kui turvaline on eKool? Andmekogud? Asutuse veebileht?“. Väino Olev tegi koosolekul ettepaneku tellida koolide, lasteaedade, huvikoolide kodulehtede ründekindluse hindamiseks ametliku auditi. Viivi Lokilt ei õnnestunud saada vastust selle kohta, kas audit tehti või on plaanis teha.

Hariduse Infotehnoloogia Sihtasutuse (HITSA) eesmärk on see, et iga haridustaseme lõpetajatele tagatakse tänapäevased digipädevused ning IKT tark kasutamine õpetamises ja õppimises ning õppetöö korraldamises tõstab õppe kvaliteeti. (HITSA n.d.) Sihtasutusel on viis põhilist tegevussuunda:

- Digipädevused kõigil haridustasemetel
- Erialaspetsiifilised digipädevused kutse- ja kõrghariduses
- IKT spetsialistide ettevalmistus kutse- ja kõrghariduses
- Õppimine ja õpetamine digiajastul
- Hariduse infosüsteem (HITSA n.d.)

HITSA on Eesti IT Kolledži asutaja ja haldaja. HITSA alla kuulub (HITSA 2015) IT Hariduse Arenduskeskus (eesmärk on IT hariduse edendamine kõikidel haridustasemetel ning IKT haridust toetavate programmide IT Akadeemia, Tiigriülikool, Progetiiger jne elluviimine). Hariduse Infosüsteemide Arenduskeskus (tegeleb IKT rakendamist toetavate infosüsteemide SAIS, ÕIS, repositooriumid ja õpikeskkonna Moodle haldamise ja arendamisega).

Eesti Hariduse ja Teaduse Andmesidevõrk (EENet 2014) on HITSA osakond, mille eesmärk on tagada teaduse, hariduse ja kultuuri jaoks vajalik infotehnoloogilise taristu areng ja stabiilne toimimine. EENeti 2015–2020 strateegias on toodud välja, et riikliku IT-taristu seire on olnud puudulik ja ei ole selget ülevaadet taristu arendusplaanidest. Samuti juhitakse tähelepanu sellele, et infosüsteemide ja taristu arendamine on olnud hajutatud ja koostöö keskselt koordineerimata.

TAAT ehk Eesti haridus- ja teadusasutustevahelise autentimise ja autoriseerimise taristu on EENeti tasuta teenus, mis võimaldab laiendada ühes asutuses kasutavate elektrooniliste identiteetide kehtivust mitmete haridus- ja teadusalaste teenuste kasutamiseks. (TAAT n.d.)

HAVIKE on EENeti loodud virtuaalne serverikeskkond, mis pakub kasutajatele lihtsalt paigaldatavat veebitarkvara. Olulisemad eesmärgid HAVIKEse arendamisel on pakkuda veebirakendusi, mis toetavad õpitegevust; muuta tarkvara kasutamise esimene samm – paigaldamine – niipalju lihtsaks, et sellega saab hakkama iga arvutialaste põhiteadmistega inimene; propageerida Eestis valminud ja eesti keele toega tarkvara. (HAVIKE n.d.) HAVIKE pakub haridusasutustele tasuta WordPressi versiooni 4.0.1 ja Joomla! versiooni 3.3.1 sisuhaldussüsteemide kodulehti. Oluline oleks ära märkida, et 2. aprilli seisuga on kõige uuem WordPressi versioon 4.1.1 ja Joomla! 3.4. Joomla! juhenditest võib leida õpetusi lausa 2011. aastast pärit versioonile 1.5.25 (Joomla! 2011), kus administraatorile on toodud rasvases kirjas välja hoiatus „Joomla uuendamisel versioonilt 1.5.25 versioonile 2.5.6 ei säili veebi kujundused (templates) ega laiendused (plugins)“. Hoiatuse võib leida ka HAVIKESe uudisterubriigist: „Enne veebirakenduse uuendamise kinnitamist palun veenduge, kas uus versioon toimib ootuspäraselt“. (HAVIKE n.d.) Uute versioonide teated on uudisterubriigis pealkirjaga „HAVIKEses veebirakendused uuendatud“, mis ei jäta selget muljet, et administraatorid peaksid seda ise tegema, kuigi võib olla, et uuendamist siiski julgustatakse mõne teise kommunikatsioonikanali kaudu.

Riigi Infosüsteemi Amet (RIA n.d.) koordineerib riigi infosüsteemi arendamist ja haldamist, et riik saaks rahvast teenindada parimal võimalikul moel. RIA-l on palju allüksuseid ja tegevusvaldkondi, näiteks kriitilise informatsiooni infrastruktuuri kaitse (KIIK), infoturbeintsidentide käsitlemise osakond (CERT), infosüsteemide turvameetmete süsteem (ISKE).

ISKE (RIA 2012) väljatöötamisel ja arendamisel on aluseks võetud Saksamaa BSI (saksa k. *Bundesamt für Sicherheit in der Informationstechnik*, inglise k. *Federal Office for Information Security*) avaldatav infoturbe standard – IT Baseline Protection Manual (saksa k. *IT-Grundschutz*). RIA on andnud välja infoturbe soovituste juhendi (RIA 2009), mis on kirjutatud kokkuvõtlikult ning väga selges keeles ning millega tutvumine võiks olla kohustuslik igale haridusasutuses administreerimisrolli omavale töötajale, kes tutvustaks dokumendi põhimõtteid ka teistele arvutikasutajatele. RIA peaks olema esimene koht, kuhu haridusasutus annab teada infoturbeintsidendist või selle kahtlusest.

Haridusasutuste IKT-ga on seotud mitmed Eesti asutused ja allüksused, millel on veel eraldi palju teenuseid. Samas Tallinna Haridusameti IKT programm ei ole seotud HITSA ja EENeti tegevustega ja EENeti HAVIKEse teenus on uuendamise eest hoiatamisega vastuolus RIA infosüsteemide turvameetmete süsteemi ISKE-ga. EENeti teenuste osakonna juhataja sõnul tehti HAVIKEse loomisel koostööd Tiigrihüppe Sihtasutuse ja EITSA-ga (Eesti Infotehnoloogia Sihtasutus), kuid mitte näiteks RIA-ga. Nagu ka EENeti strateegias oli välja toodud, siis eri asutuste koostöö ei ole koordineeritud. Arvestades Eesti riigi suurust, ei oleks ilmselt mõtet koostada Tallinna koolidele eraldi veebilehekülgede standardit ning veebilehekülje kasutamise ja majutamise võimalust, nagu 2011–2015 aasta Tallinna Haridusameti IKT programmis kinnitati, eriti juhul kui EENet pakub sellist tasuta teenust juba 2008. aastast. Samas kui EENeti enda strateegiast ilmneb, et „IT-taristu seire on olnud puudulik ja ei ole selget ülevaadet taristu arendusplaanidest“, siis jääb küsitavaks, kas koolidele peaks selle teenuse kasutamist soovitama. Standardi koostamisel ei tohiks unustada ka turvalisust ja selle tagamise eest vastutajate määratlemist ning seda võiks luua koostöös RIA-ga.

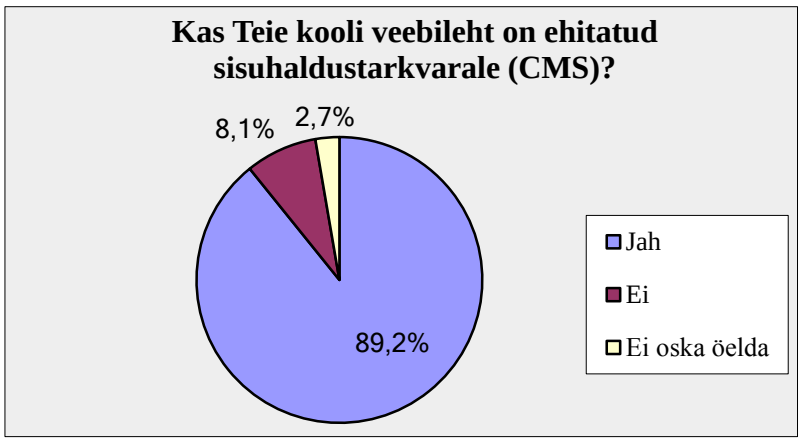
4. Turbeteemaline küsitlus Harjumaa koolitöötajate seas

4.1. Küsitluse meetod ja ettevalmistus

Küsitluse eesmärk oli uurida koolides vastava valdkonnaga tegelejate suhtumist veebilehtede turbesse. Sihtrühmaks valiti Harjumaa üldhariduskoolid Eesti Hariduse Infosüsteemi lehelt. Koolide veebilehtedelt otsiti välja info- või IT-juhi kontakt. Juhul, kui sellise töötaja kontakti ei leitud, siis võeti välja üldine e-posti aadress. Kaaskirjas märgiti ära küsimustiku eesmärk, rõhutati vastuste anonüümsust ja pakuti välja võimalus juunikuus tutvuda küsimustiku tulemustega. Küsimustik saadeti 136-le koolile ja vastused saadi 37-st. Koos kommentaarilahtritega koosnes küsimustik 16 küsimusest, millest paljud eeldasid põhjalikke vastuseid (vt Lisa 1. Koolide veebilehtede turvalisuse küsitlus). Küsimustiku koostamiseks kasutati SurveyMonkey lahendust.

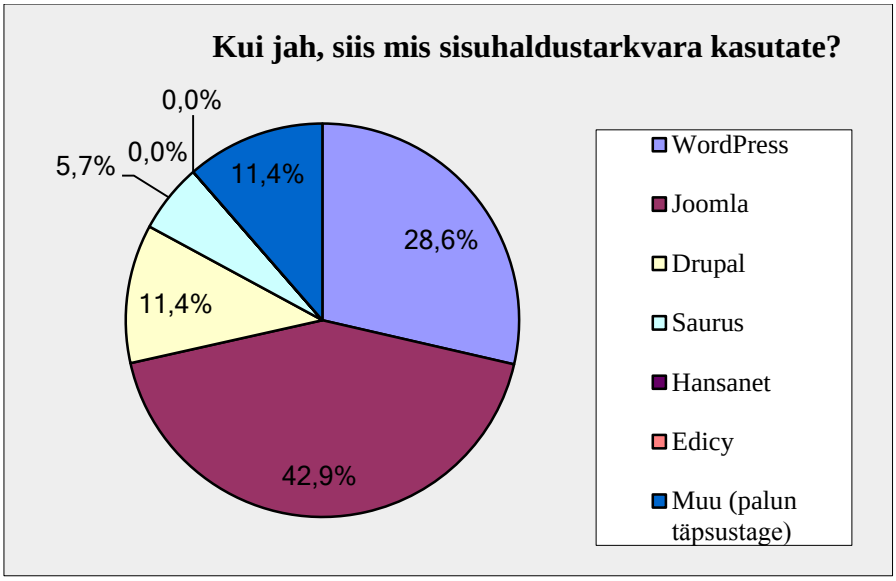
4.2. Küsitluse tulemused ja analüüs

Esimese küsimusega uuriti, kas kooli veebileht on ehitatud sisuhaldussüsteemile. Vastused kinnitavad Hormi (2012, 41) tulemust, millest selgus, et enamik koole kasutab vabal tarkvaral põhinevat sisuhaldussüsteemi. Selle töö kirjutaja küsimustikule vastanutest kasutab sisuhaldussüsteemi (sõltumata maksumusest) 89,2% ehk 33 vastanud koolidest (vt Joonis 2).



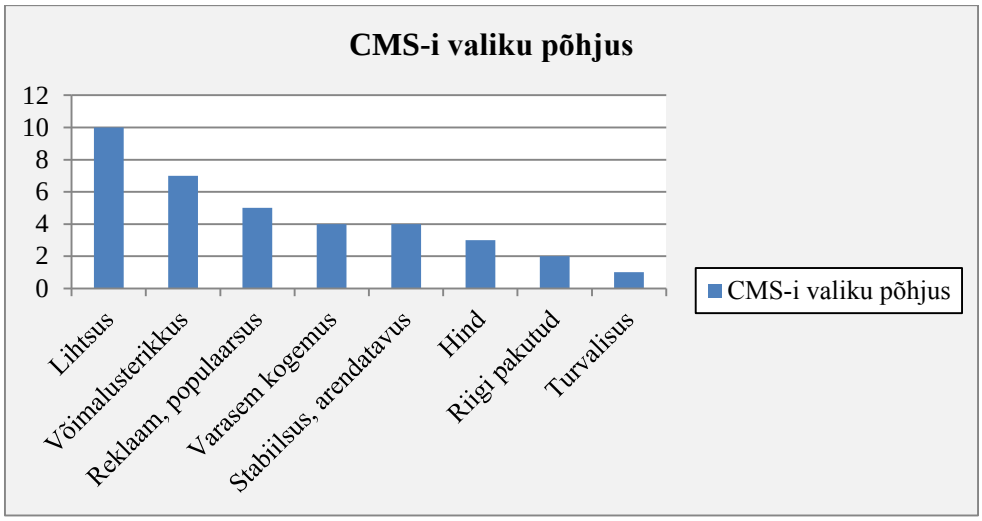
Joonis 2. Harjumaa koolide sisuhaldussüsteemide kasutamine.

Sisuhaldussüsteemidest kõige populaarsemad olid tasuta tarkvarad. Kõige populaarsem oli Joomla! (42,9%), seejärel WordPress (28,6%), Drupal (11,4%) ja Saurus (5,7%) (vt Joonis 3). Vastusevariandi „muu“ alla oli märgitud enda tehtud CMS, WebCreator Pro, Etomite ning ühel korral tõdeti, et seda vastust ei teata (kõiki ühel korral). Vastuste seast kõige populaarsemate sisuhaldussüsteemide turvalisuse kohta saab täpsemalt lugeda peatükis 5.



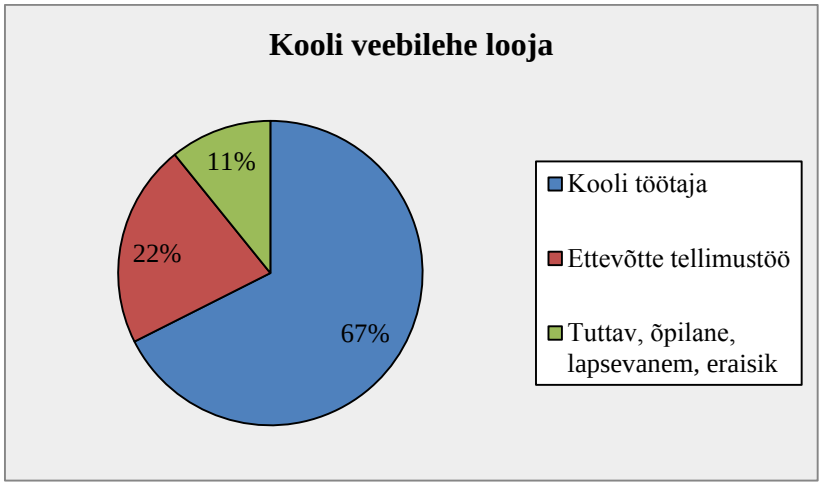
Joonis 3. Populaarseimad sisuhaldussüsteemid Harjumaa koolides

Vastajatelt küsiti, mille põhjal nende kool just selle CMS-i valis. Et näha, mis vastajad ise pakuvad, vastusevariante ette ei antud. Vastused kategoriseeriti küsitluse lõppedes. Mõnes vastuses selget põhjendust ei olnud – nt „oli ainuke mõistlik valik“, „ei olnud otsustamise juures“, „oli juba nii valitud“. Selgelt põhjendati valikut kõige enam lihtsusega, mida mainiti 10 korral (vt Joonis 4). Järgmisel kohal oli võimalusterikkus, mida nimetati 7 korral ning reklaam ja populaarsus, mida nimetati 5 korral. Valiku põhjendamisel nimetati turvalisust vaid ühel korral. Stabiilsust ja arendatavust (kas süsteemi loojad arendavad seda edasi) on nimetatud 4 korral.



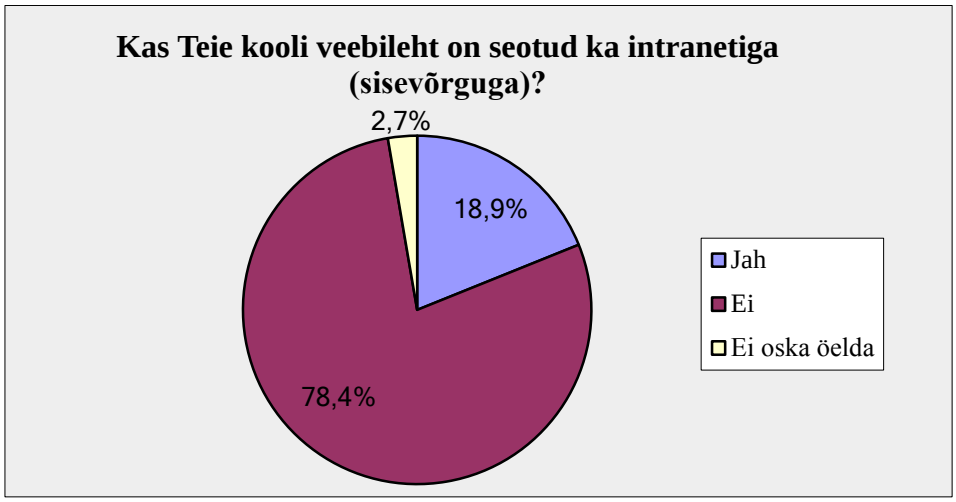
Joonis 4. CMS-i valiku põhjendus Harjumaa koolides

Vastajatelt küsiti ka seda, kes nende kooli veebilehe tegi. 67% vastas, et kooli töötaja ja vaid 22%, et veebileht tehti vastava alaga tegelevas ettevõttes tellimustööna (vt Joonis 5). Mitu vastajat märkisid ära, et lehe loonud töötaja neil enam ei tööta. Samuti tõdeti, et leht tehti koostöös õpilasega. Turvalisuse seisukohalt võib siin näha ohtu, sest sellisel juhul võib endistel töötajatel ja õpilastel olla lihtsam juurdepääs lehele ning samuti ka võimalik rünnaku motivatsioon. 11% vastasid, et lehe lõi otsast lõpuni lapsevanem, õpilane, tuttav või muu eraisik. Seda ei ole võimalik määratleda, kui head valdkonna teadmised ja oskused lehti loonud lapsevanematel, õpilastel, kooli töötajatel ja teistel on, kuid turvalisuse mõttes oleks kindlam töö tellida professionaalidelt, kuid otsuse määrab ilmselt hind. Samas, EENeti HAVIKEse programmi kaudu pakutakse koolidele 2008. aastast tasuta Joomla! ja WordPressi põhjal kodulehti.



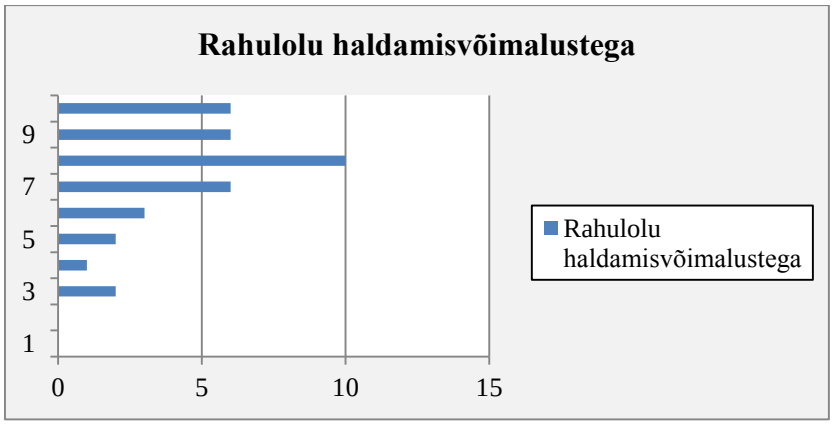
Joonis 5. Harjumaa koolide veebilehtede loojad

Vastajatelt küsiti ka seda, kas nende kooli veebileht on seotud ka intranetiga. 18,9% kinnitasid, et on (vt Joonis 6). Kuna intraneti ehk sisevõrgu kasutamine on mõeldud organisatsioonisiseseks kommunikatsiooniks, siis tihti ununeb, et see ei muuda võrku automaatselt turvaliseks (NovaInfosec 2009). Rünnakuid võib lisaks organisatsiooni seest tulla ka väljastpoolt. Ohtlik võib olla siseneda intranetti distantsilt (Microsoft n.d.). Nagu interneti puhulgi, on ohtlikud pahavara, nõrgad paroolid, krüpteerimatus jms (Brecht 2013).



Joonis 6. Intraneti kasutus

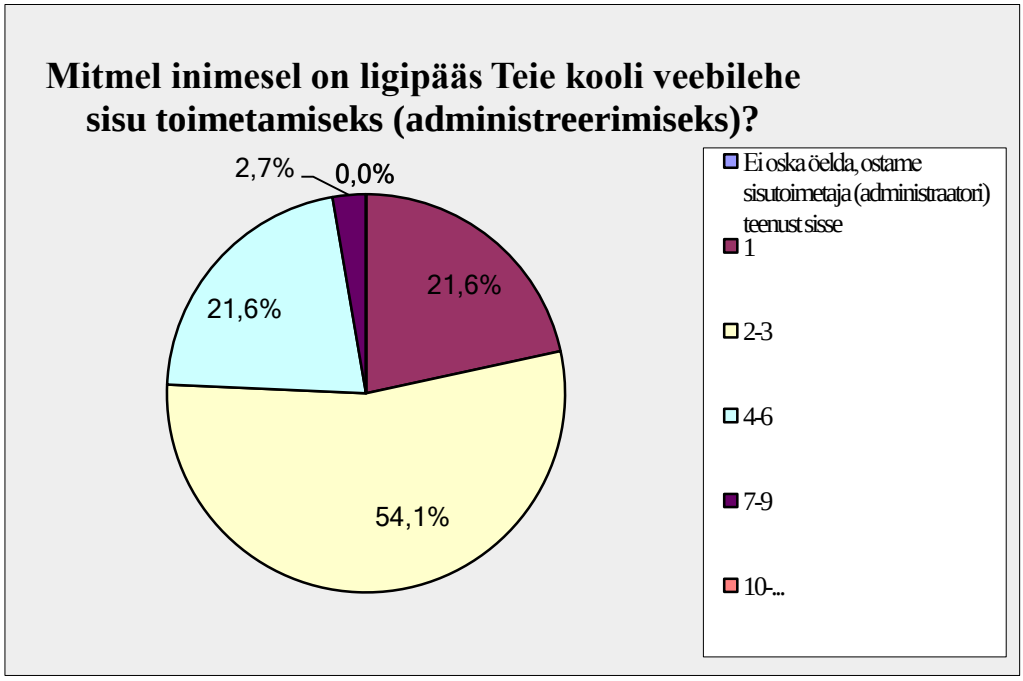
Oma kooli veebilehe haldamisvõimalusega oldi 10 punkti süsteemis rahul keskmiselt 7,61 puntki väärtuses (vt Joonis 7). Joonisel on horisontaalselt märgitud vastajate arv ja vertikaalselt hinnangu punktid.



Joonis 7. Rahulolu veebilehe haldamisvõimalustega

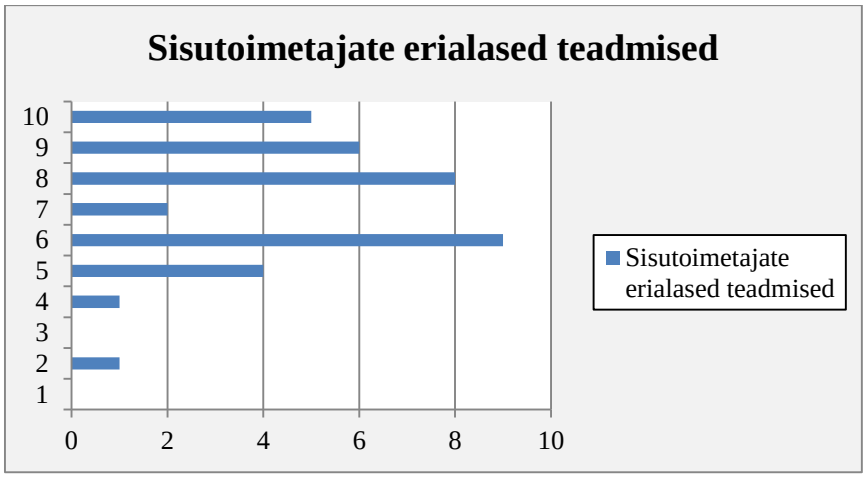
Hinnangut põhjendades tõdeti üldist rahulolu, kuid negatiivsena toodi kõige enam välja, et pole piisavalt lihtne ja mugav, mis on huvitav, sest CMS-i valikut põhjendati kõige enam just lihtsuse ja võimalusterikkusega.

Paroolide ja ligipääsuga seonduv on veebilehtede turvalisuse üks olulisemaid teemasid. Vastajatelt küsiti, kui mitmel inimesel on ligipääs nende veebilehe toimetamiseks (vt Joonis 8). 54,1% vastas, et 2–3 inimesel; 21,6% vastas, et 4–6 inimesel ning 21,6% vaid ühel inimesel. 7–9 inimesel oli ligipääs 2,7% vastanutest. Üldjuhul on kindlam anda ligipääsud väiksemale arvule inimestele. 10. küsimusega uuriti, mitu inimest on kahe viimase aasta jooksul veebilehel muudatusi teinud. Mitte keegi ei vastanud, et nende koolis oleks ligipääs veebilehele enam kui kümnel inimesel, aga samas tunnistasid kahe kooli esindajad, et kahe aasta jooksul on üle kümne inimese veebilehel muudatusi teinud. See võib tähendada nii seda, et ligipääsu omajad on vahetunud, kui ka seda, et paroole jagatakse – see oleks muidugi suur turvaoht. 8 kooli vastasid, et ligipääs on vaid ühel töötajal. Samas lausa 13 kooli vastasid, et viimase kahe aasta jooksul on muudatusi teinud vaid üks töötaja. Järelikult on koole, kus ligipääs on töötajatel, kes seda tegelikult ei vaja. Turvalisuse seisukohalt oleks mõistlik sellised ebavajalikud ligipääsud sulgeda.



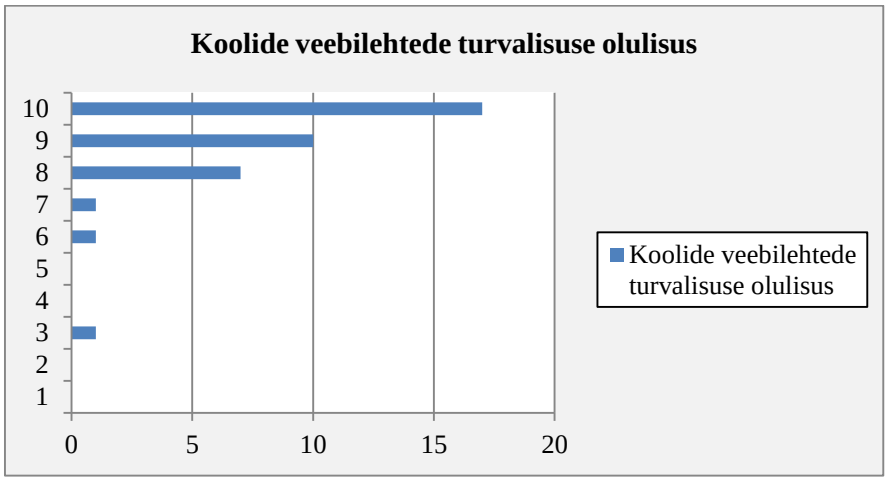
Joonis 8. Ligipääsevate töötajate arv

Veebilehe sisutoimetajate erialaseid teadmisi hinnati üsna kõrgelt – 10st keskmiselt 7,28 punkti väärtuses (vt Joonis 9). Joonisel on horisontaalselt märgitud vastajate arv ja vertikaalselt hinnangu punktid. Hinnanguid oli aga väga madalast väga kõrgeni ja kõige enam valiti vastuseks 6 punkti.



Joonis 9. Sisutoimetajate erialased teadmised

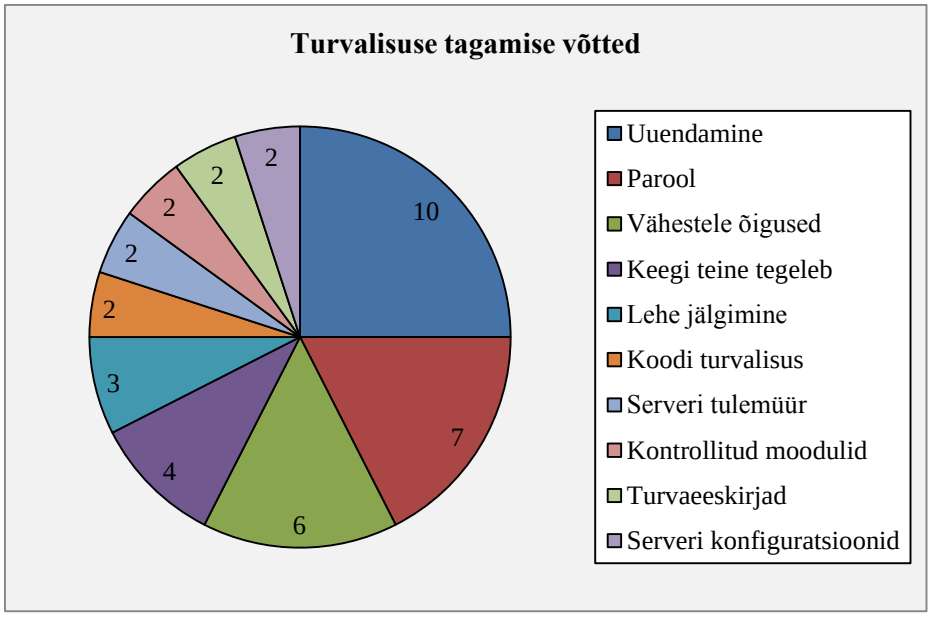
Turvalisuse teemadele kitsenedes uuriti vastajatelt, kui oluliseks peavad nad koolide veebilehtede turvalisust (vt Joonis 10). Lausa 17 kooli hindasid olulisuse kõige kõrgemate punktidega. Keskmiselt oli hinnang 8,97 puntki. Kõige vähem oluliseks hindas turvalisuse kool, kes andis turvalisuse olulisusele vaid 3 puntki. Huvitav on see, et kui keskmiselt hinnatakse turvalisus ülimalt oluliseks, siis vaid ühel korral nimetati turvalisus enda kooli CMS-i valimise põhjenduste seas.



Joonis 10. Koolide veebilehtede turvalisuse olulisus

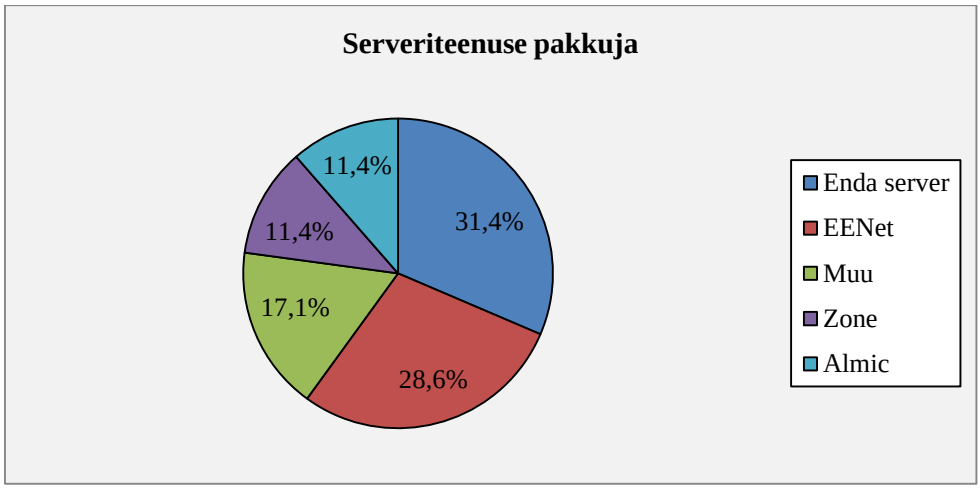
Kuigi veebilehe turvalisust peetakse ülimalt oluliseks, siis kahjuks osati üsna vähe nimetada konkreetseid võtteid, mida vastaja koolis veebilehe turvalisuse tagamiseks tehakse. Vastusevariante ette ei antud ja vastused kategoriseeriti küsitluse lõppedes. Joonis 11 näitab vähemalt kahel korral nimetatud võtteid. Kõige enam nimetati tarkvara uuendamist (10 korral). Sellele küsimusele vastas 34 kooli, seega tegelikult on 10 korda kindlasti liiga vähe. 7 korral nimetati paroolide keerulisust või nende aeg ajalt vahetamist. 6 korral nimetati, et ligipääsu ja administreerimisõigused on võimalikult vähestel. 4 korral

kirjutati, et turvalisuse tagamisega tegeleb serveriteenuse pakkuja, kuigi tegelikult see ei tähenda, et veebilehe haldajal ei tuleks ise turvalisuse tagamisega tegeleda. 3 korral märgiti ära, et veebilehte jälgitakse ja ka siis, kui midagi sinna lisada ei ole vaja. 2 korral nimetati koodi turvalisena hoidmist, serveri tulemüüri kasutamist, CMS-i vaid kontrollitud moodulite kasutamist, turvaeeskirjade järgimist ja serveri turvalist konfiguratsiooni. Vaid ühel korral märgiti veel turvalisuse teemalisi koolitusi, viirusetõrje kasutamist, CMS-i turvalisuse tagamist pakkuvaid pluginaid, administreerimislehe peitmist ja varunduse tegemist. See, et varunduse tegemist mainiti vaid ühel korral ei olnud ootuspärane. Serveriteenuse pakkujad teevad küll varundust, kui üldjuhul on selle kasutamine tasuline. Kõige lihtsam viis veebilehe varundamiseks on FTP-klientprogrammiga kõik veebilehe failid alla laadida ja salvestada, kas kõvakettale või pilveteenusesse. Kolm vastajat ei osanud nimetada ühtegi konkreetset turvalisuse tagamise tegevust ja kolm vastajat jätsid selle küsimuse vahele. Arvestades, et lahknevus veebilehe turvalisuse ülimalt oluliseks hindamise ja sellealaste võtete teadlikkuse vahel on väga suur, võib öelda, et koolitused ja seminarid võiksid sellises olukorras vägagi kasuks olla.



Joonis 11. Veebilehe turvalisuse tagamise võtted Harjumaa koolides

Kõige enam ehk 31,4% koole kasutab enda serverit (vt Joonis 12). Teenusepakkujatest on kõige enam kasutusel EENet (28,9%), Zone (11,4%) ja Almic (11,4%). Muu alla koondati kõik vastavalt ühel korral nimetatud: Atea, Andmevara, Virtuaal.com, Elion, Veebimajutus ja Koduleht.net.

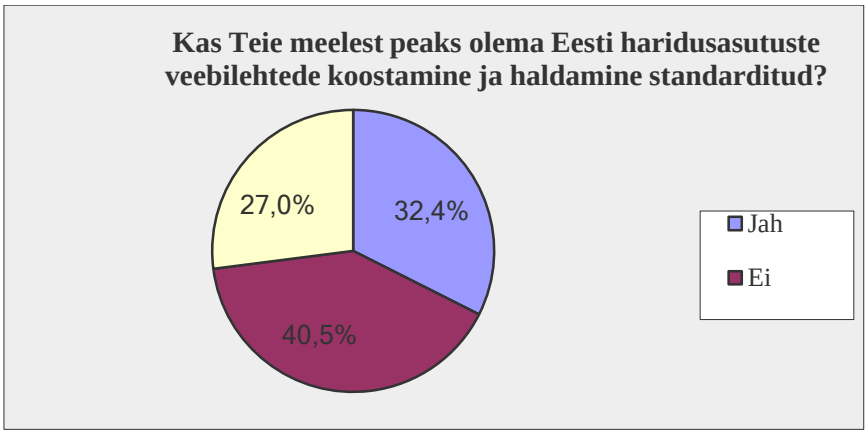


Joonis 12. Harjumaa koolide serveriteenuse pakkujad

Küsitlusele vastajatelt uuriti, kas ja milliseid turvaprobleeme on nende kooli veebilehel esinenud. Arvestuslikult 40,5% vastanutest tõi konkreetseid intsidentide näiteid (vastused on esitatud muutmata kujul). Üks vastaja tõdes, et „paar korda on WP kommentaaridesse tulnud võõraid teateid 4 aasta jooksul“. Ühel teisel vastajal on sarnane näide „phpbb foorumisse on roboti abil spami postitatud“. Keerulisemat olukorda kinnitab vastus „Eenetile tehtud rünnak. Meie kodulehte ei olnud võimalik taastada. Taastamise kulud maksis eenet kinni.“ Üks vastaja kinnitab, et Joomlas pole neil probleeme tekkinud, kuid „kunagisel Wordpressi kodulehel sai viirus mingi mooduli kaudu sisse“. Kolm vastajat kirjutasid kodulehe ülekoormamisest. Üks neist sõnab, et „koolis valikainetele registreerumise avanedes koormasid õpilased veebilehe üle. Kiireks lahenduseks sai tookord tehtud lihtsalt staatiline avaleht kahe valikuga, millest üks viis kodulehele ja teine valikainete registreerumisele.“ Teine selgitab, et „seda oskavad teha juba 6 klassi õpilased toksides F5 klahvi - DDOS juhendid on samuti googles olemas. Näiteks kui on moodle testid ja moodle jookseb kooli serveris siis tekib mõnel rüblikul idee server üle koormata, et ei peaks teadmiste kontrolli läbima. Siin aitab ikka serveri konfiguratsioon.“ Sama vastaja kirjutab ka, et „serveri avatud portide ja serveris jooksva tarkvara info skaneerimist on väga tihti - üle päeva. SSH logimise katseid samuti iga nädal. Aegajalt kui SSH ründed lähevad väga sagedaseks on mõistlik SSH teenus mõneks ajaks sulgeda või hakata filtreerima SSH juurdepääsu host.allow serveri scriptiga. Ühel korral jäi tööõnnetusena avatuks ehk õigustega 777 veebilehe üks kataloog. Kaks nädalat oli avatud ja selle ajaga paigaldati sellesse kataloogi pahavara, mis saatis laiali spämmi. ;)) Õnneks saime jaole.“ Ära on märgitud näiteks „kooli aadressilt pay-pali spammi saatmine“. Kirjutatud on ka näiteks, et „eelmise veebilehe versioonis oli kasutajatel võimalik teha omale konto. Sinna

tekkisid erinevad uued kasutajad kes polnud kooliga seotud. Nüüdsest on see võimalus ära võetud ja sisselogimise aken on peidetud.“ Samuti vastati, et „kunagi väga ammu, eks ikka on maha võetud esileht (uuendamata on olnud). Teenusepakkujat rünnatakse (mitte meie pärast, vaid üleüldiselt).“ Näiteid on ka soovimatu info sisestamisest. „Aastaid tagasi suutis keegi galeriisse riputada kahtlase väärtusega pildi“, „just hiljuti veebruari algul. CERT-EE saatis teate, et meie kodulehele on meie teadmata pilt riputatud (Hacked by the warrior)“ ning „on olnud korra juhus, kus lipsas sisse skript, mis pani erinevaid pilte veebilehele, aga saime sellele kiirelt jälile.“ Veel üks näide: „Korra on kodulehe headeri üle kirjutatud ja google search näitas kinga reklaami, kui kooli otsida.“

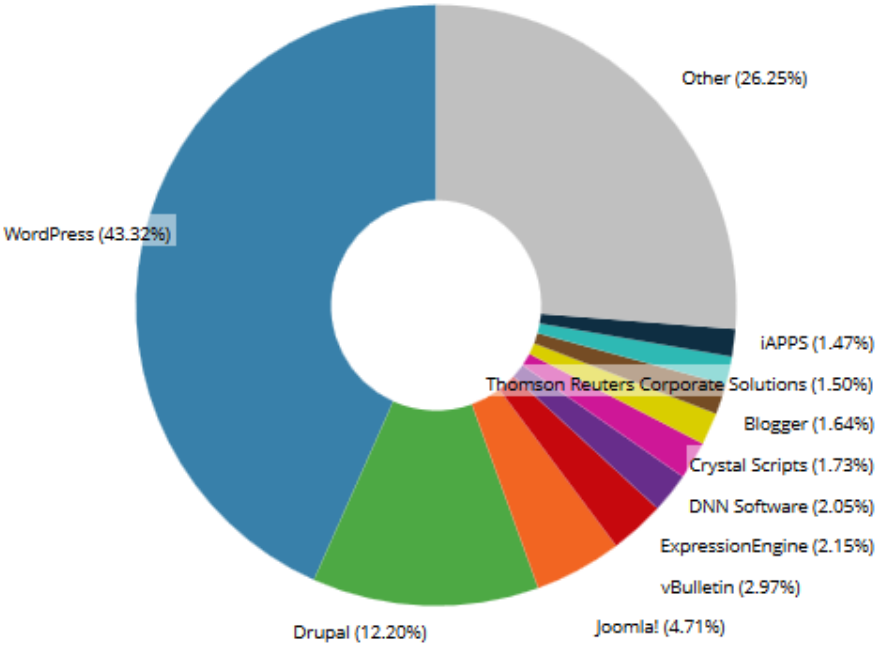
Vaatamata sellele, et Harjumaa koolide nii CMS-ide valik, serveriteenuse pakkujad kui ka toimunud turvaintsidendid pakuvad üsna kirju spektri, siis veebilehtedega seonduva standardimist selgelt ei soovita. Küsimusele, kas Eesti haridusasutuste veebilehtede koostamine ja haldamine peaks olema standarditud, on vastanud „jah“ 32,4% kooli esindajatest. 40,5% protsenti on vastanud „ei“ ja 27% „ei oska öelda“ (vt Joonis 13) – küsimuses üksmeelt ei leita. Selle põhjal võiks öelda, et juhul kui IKT programmi raames (vt peatükk 3) plaanitakse standardi koostamist jätkata, tuleks kindlasti olla suhtluses koolide vastava valdkonnaga tegelevate töötajatega. Standardimise vastased tõid näiteks välja, et „siis peaks jälle mingi uue asja ära õppima“, „liigne standardiseerimine viib totalitaarsuseni“, „kool kaotab sellest oma näo ja isiksuse“. Pooldajad tõid aga välja, et „koolidel võiks olla võimalus kasutada keskset kodulehe teenust. Nagu kohalikel omavalistustel on see võimalus“, „kõik kodulehed võiksid olla standarditud vähemalt niipalju, et näiteks kontakt andmed, inimeste nimed ja kõik muu kiiresti leitav info oleks kättesaadav ühtemoodi loogiliselt kõikidel kodulehtedel“, „peaks olema teatud info kergesti kättesaadav. Näiteks sisukord on samasugune iga kooli lehel.“ Kahevahele jäävate vastuste seast võiks tuua välja ühe teemat kokkuvõtva vastuse: „Standardiseerimine võiks olla (ja osalt juba ongi) sisuline. On määratud, mis kooli veebilehel olema peab. Tsentraalse veebihalduse puhul kaob aga ära kooli isikupära. Keskse halduse ja standardse kujunduse võimalus võiks siiski olemas olla, et kõik koolid ei peaks ise veebilehe loomise, kujundamise ja turvauuendustega tegelema. Koolidel võiks siiski säilida õigus ise otsustada, kuidas oma veebiväljund tekitada.“



Joonis 13. Standardimise soov

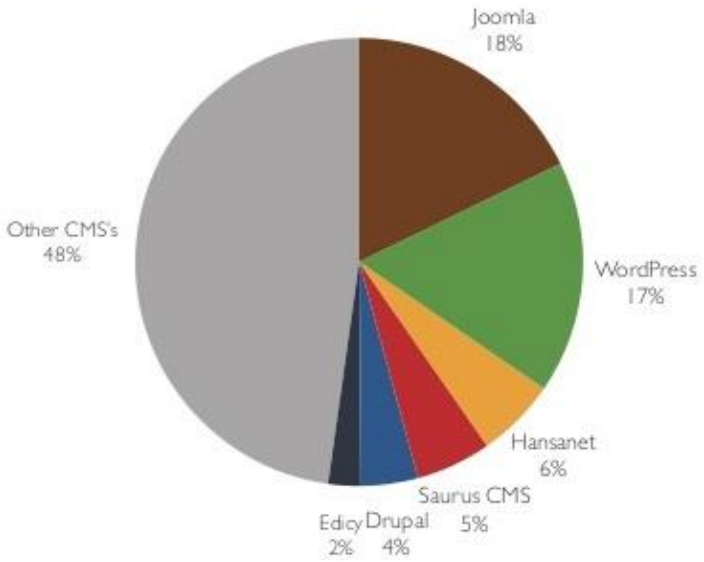
5. Populaarseimad CMS-id ja nende turvalisus

Harjumaa koolide küsitlusest selgus, et Joomla! (42,9%), WordPress (28,6%) ja Drupal (11,4%) on Harjumaa koolide seas kõige populaarsemad CMS-id. Sama esikolmikuni võib jõuda autoriteetse statistikakoguja Web Technology Surveysi (W3Techs n.d.) andmetest. Need kolm on ka Builtwithi statistika (2013) järgi kõige populaarsemad. Joonis 14 näitab, mis CMS-e miljoni kõige vaadatuma lehe seas kasutatakse (juhul kui kasutatakse CMS-i).



Joonis 14. Miljoni kõige enam vaadatava veebilehe CMS-i kasutus (Builtwith 2013)

2013. aasta märtsis andis Eesti kõige suurem Drupali arendaja Mekaia meie riigi CMS-i turuosast ülevaate (vt Joonis 15). Nad uurisid 23 000 .ee domeeniga kodulehte. CMS-i äratundmiseks kasutati selleks valmistatud skripti, mis arvab ära kasutatud tehnoloogia uurides HTML-i eri mustreid. 2013. aasta seisuga tunti ära 44% CMS-idest. 2009. aastal oli vabavaraliste CMS-ide osakaal võrreldes tasulistega 25% ning aastaks 2013 oli see kasvanud 50%-ni. (Mekaia 2013) Nende uuringus selgus, et kolm maailma populaarseimat CMS-i kuuluvad kõige populaarsemate hulka ka Eestis. Maailma populaarsuselt kolmas on Eestis küll viies, aga Mekaia tehtud uuringust selgub ka (Mekaia 2013), et Drupali osakaal on viimase aastaga jõudsalt tõusnud.



Joonis 15. Mekaia uuring eri CMS-ide Eesti turuosadest (Mekaia 2013)

5.1. Joomla! turvalisus

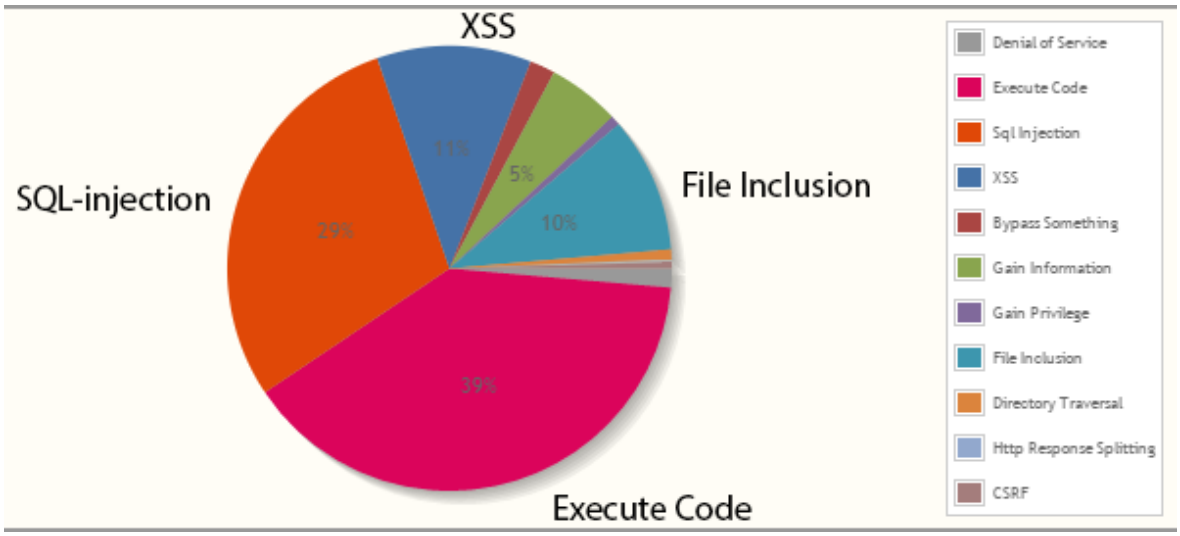
Dean Marshall, üks juhtivaid Joomla! (edaspidi Joomla) eksperte ja arendajaid on oma ettevõtte lehel toonud välja väga levinud Joomla kasutamisega tekkinud turvalisusega seotud probleemid.

Nendeks on näiteks:

- Nõrk administraatori parool
- Paroolide teistega jagamine ja kättesaadavasse kohta jätmine
- Vananenud tuumfailid (*core files*)
- Odava majutusteenuse pakkujad jagavad oma serverit paljude klientide vahel, seega ei pruugi nad suuta tagada selle piisavalt head kiiruse ja turvalisuse konfiguratsiooni.
- Joomla laseb faile üles laadida ilma igasuguste piiranguteta. Kui tegemist on foorumi laadse portaaliga, mis lubab faile üles laadida, siis see võib tekitada turvaprobleeme. Näiteks kuritahtlik kasutaja võib üles laadida liiga suuri faile, mis nõrgestab kogu lehe tööd või mistahes tüüpi faile, mille avamise tagajärgi ei oska ette näha.
- Puudub korralik kontroll Joomla laienduste arenduse üle. (DM Consultancy n.d.)

Nendest esimesed kolm on inimfaktorid, mis tähendab, et suur osa probleemidest tuleb kasutajate oma tegemata jätmisest, teadmatusest või hoolimatusest. Samad probleemid on ka teiste sisuhaldustarkvaradega. Järgnevalt tuuakse paar näidet tehnilistest turvaaukudest, mis on tuvastatud konkreetsetel Joomla versioonidel.

CVE kodulehel asuvalt graafikult (vt Joonis 16) on näha, et Joomlal on arvuliselt kõige rohkem raporteeritud turvanõrkusi seotud omavolilise koodi käivitamisega. See moodustab 39% kõikidest rünnakutest aastate jooksul. Teisel kohal on SQL-süstimine, mille väga sagedane põhjus on see, et laienduspaki on teinud kolmandad osapooled, kes ei järgi nii rangeid turvareegleid kui Joomla enda arendajad. Üks hullemaid olukordi saab olla näiteks see, kui keegi süstib sellise koodi URL-i reale: `http://www.minujoomlaveebileht.ee?userid=5'; DROP DATABASE minujoomlaandmebaasinimi-` . See kustutab terve andmebaasi koos selle sisuga. Kuna URL-i lõpus on MySQL keele kommenteerimismärk, siis andmebaas ei anna veatõrget ja arvab, et kõik läks nii nagu pidi. Mõnedel varasematel versioonidel kui 1.5.25 on SQL-süstimise võimalused olemas. Kui aga kasutada kogu aeg uusimat Joomla versiooni ning ei installi oma veebilehele kolmandate osapoolte tehtud mooduleid, siis on taoliste rünnakute oht väike. (Itoctopus 2012)



Joonis 16. Joomla nõrkuste statistika CVE andmetel (CVE Details 2014a)

5.2. WordPressi turvalisus

WordPress uuendab oma sisuhaldustarkvara keskmiselt kaks korda kuus. Uutes versioonides on tähtsal kohal turvaaukude parandamine ja kõrvaldamine. Kuna häkkerid

arenevad kogu aeg ja õpivad järjest rohkem rünnatavaid süsteeme tundma, siis peavad sisuhaldustarkvarade arendusmeeskonnad süsteemi pidevalt täiustama ja proovima rünnakuid ennetada. Sellest tulenevalt on olemas palju versioone, käesoleva bakalaureusetöö kirjutamise hetkel on uusim 4.1.2, mis tehti avalikuks 21. aprillil 2015 (WordPress.org 2015).

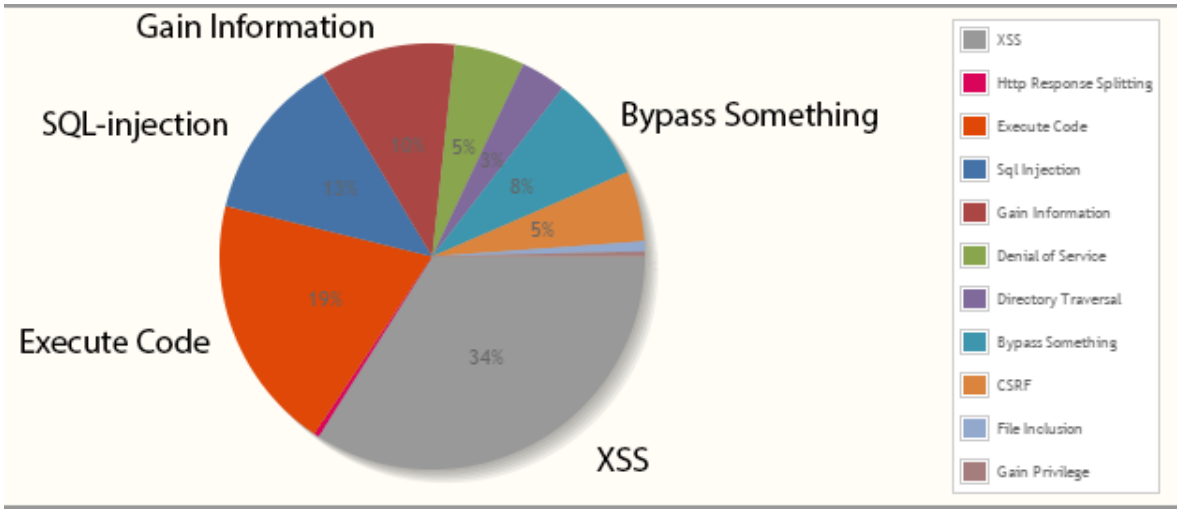
Siit aga tulenebki WordPressi platvormile loodud veebilehestike suurim turvaprobleem – WordPressi tarkvara vajab serverites pidevat turvalisematele versioonidele uuendamist. Nii WordPressi kui ka teiste CMS-ide suurimaks turvariskiks on see, et häkkerid on loonud automatiseeritud vahendid, mis aitavad väga kiiresti leida erinevate veebilehtede turvaauke ja nõrkusi. Turvaaukude leidmist aitab väga oluliselt vähendada sisuhaldustarkvarade õigeaegne uuendamine.

2007. aasta mais tehtud uuringust selgus, et 98% WordPressi sisuhaldust kasutavates veebilehtedes on privaatne informatsioon, kas vähemal või rohkemal määral kättesaadav mitte selleks volitatud isikutele (Kierznowski 2013). Põhjuseks oli uuendamata või mitte enam toetatud WordPressi sisuhalduse versioon. 2008. aasta detsembris väljastati versioon 2.7, kuhu oli sisse ehitatud ühe-kliki uuendamise protsess, mis tegi kodulehe korraliku haldamise palju lihtsamaks ja võimaldas uuendada oma tarkvara palju mugavamalt. (WordPress n.d.) Kui aga serveriteenuse pakkuja serveris ei ole failid uuenduseks vajalike õigustega, siis ei lubata WordPressi ühe-kliki uuendamist ning kasutajad peavad selleks ikkagi FTP-klientprogrammi kasutama ja paljude jaoks on see väga tülikas ning võib jääda tegemata.

WordPressi turvalisuse probleemid on välja toodud CVE kodulehel aastast 2004. Andmetest tuleb välja, et kõikide aastate ja versioonide peale kokku on kõige suurem probleem XSS-i ehk murdskriptimisega (vt Joonis 17), mis moodustab 34% kõikidest tuvastatud turvalisust puudutavatest probleemidest. Teisel kohal on omavoliline koodi käivitamine ja kolmandal kohal informatsiooni kättesaamine.

Üks näide ohtlikust XSSi nõrkusest esineb versioonides 3.0 kuni 3.9.2. Nimelt on nendes versioonides võimalik postituslahtritest käivitada JavaScripti. Need lahtrid on eelseadistatud selliselt, et need ei nõua autoriseerimist. JavaScript käivitatakse kui see kuvatakse kasutajale, kas blogi postitustes, eraldi lehel või administreerimiskeskkonnas saabunud postituste all. Administraatorini jõudnud postituses ei kuvata JavaScripti ja

häkker saab märkamatult tegutseda. Kõige tavalisemal juhul postitab häkker vastavasse lahtrisse JavaScripti koos mõnede linkidega, et postitus jõuaks administraatorile läbivaatamiseks. Kui nüüd administraator läheb saabunud postituste lehele, siis käivitatakse postitatud JavaScript. Käivitatud kood saab nüüd kasutada selle veebisaidi administraatori õigusi. Näiteks alustuseks kustutatakse jäljed käivitatud koodist ning järgmiseks vahetatakse aktiivse kasutaja parool ära või lisatakse uus kasutaja administraatori õigustega. Häkker saab soovi korral lisada uue PHP-faili serverisse. Kui postitus tehakse avalikuks, siis iga kasutaja, kellele seda posti näidatakse, on järjekordne ohver, kelle kasutajaõigusi saab ära kasutada. Näiteks saab häkker kasutajaõigusi ära kasutada andmepüügiks või rämpostitamiseks. (Pynnonen 2014)

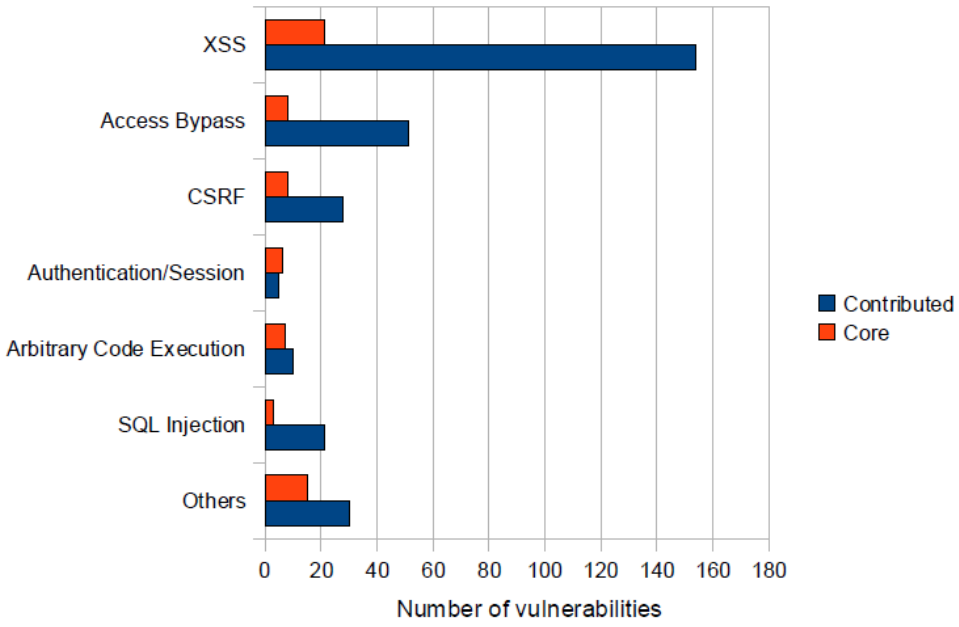


Joonis 17. WordPressi nõrkuste statistika CVE järgi (CVE Details 2014b)

Mathew J. Schwartz (Schwartz 2013) toob artiklis „WordPress Site Hacks Continue“ välja ühe väga olulise turvalisusega seotud aspekti, milleks on see, et selliseid sisuhaldustarkvarasid on väga lihtne kasutada, aga sellega kaasneb tihti olukord, et mitte ükski IT-alane professionaal ei vaata tarkvara installimist ega selle kasutamist üle. Probleeme võivad tekitada ka juba kõige lihtsamad valikud, nagu näiteks tugeva salasõna valimine. Administraatori kasutajanimeks näiteks „admin“ valimine muudab lehe kohe väga heaks sihtmärgiks. Väga suur turvarisk seisnebki selles, et CMS-ide installimine on tehtud lihtsaks, mugavaks ning igaühele jõukohaseks.

5.3. Drupali turvalisus

Drupal ei uuenda oma versioone sama tihti kui WordPress. Viimane versioon on 7.36 ja avaldati 2. aprillil 2015 (Rothstein 2015). Drupali meeskond võtab OWASPi programmi turvariskide hinnanguid tõsiselt arvesse. Ka näiteks 2010. aasta turvaraportis on see välja toodud (Jeavons ja Knaddison 2010, 2). Raportis hinnatakse eraldi Drupali turvaprobleeme ning leitakse, et kõige enam levinud nõrkus on XSS ehk murdskriptimine (vt Joonis 18). Punasega on joonisel tähistatud probleemid Drupali tuuma koodis ja sinisega vabatahtlike arendused.



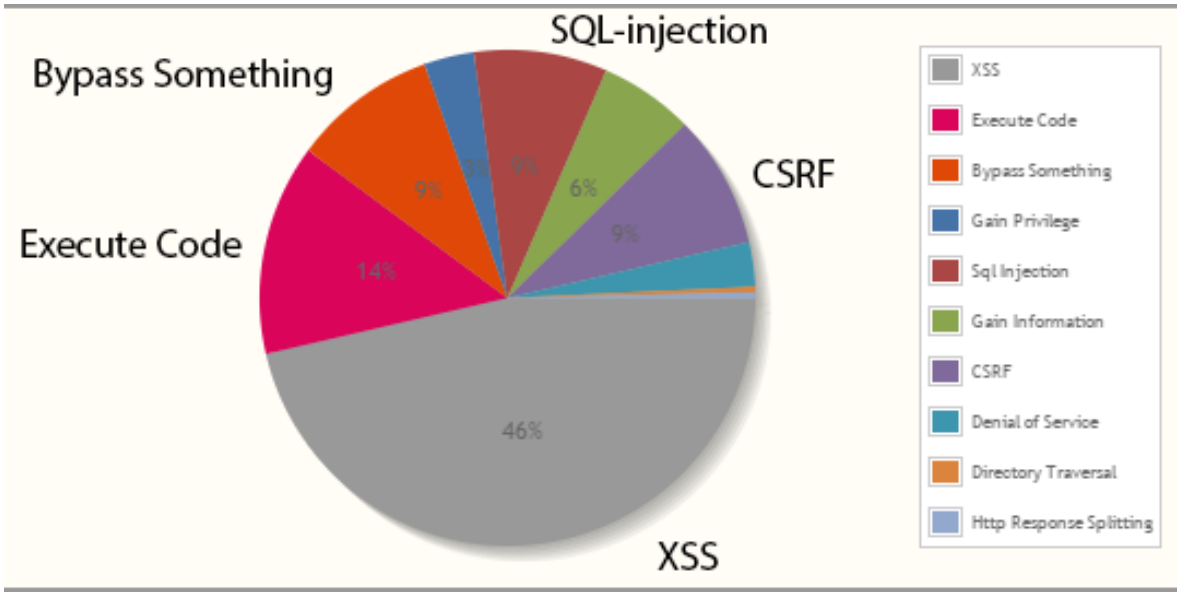
Joonis 18. Drupali turvaprobleemid (Jeavons ja Knaddison 2010, 4)

Raportis väidetakse, et Drupali tuum vaadatakse koodijupi haaval üle ja kõik peab vastama koodimis- ja turvastandarditele. Näiteks Drupal 7 puhul osaleb selles töös umbes 700 inimest, kellest vaid kahel on koodi üles laadimise luba (*commit access*).

Suuremad probleemid võivad esineda individuaalsete lehtede kohandatud koodiga. Drupali WhiteHat Security raporti kohaselt (Jeavons ja Knaddison 2010, 5) oli 90% 120st leitud nõrkusest seotud just kohandatud kujundusteemadega.

CVE andmebaasis oleva info põhjal (vt Joonis 19) on näha, et peaaegu pooled kõikidest turvanõrkustest on seotud murdskriptimisega. Selle peamine põhjus on Drupali tuuma mooduli ebapiisav kontroll enne kuvamist erinevates rakendustes. Moodulite nimed ja kirjeldused, mis asuvad metaandmete failides (tuvastatakse nende .info laiendustega) on

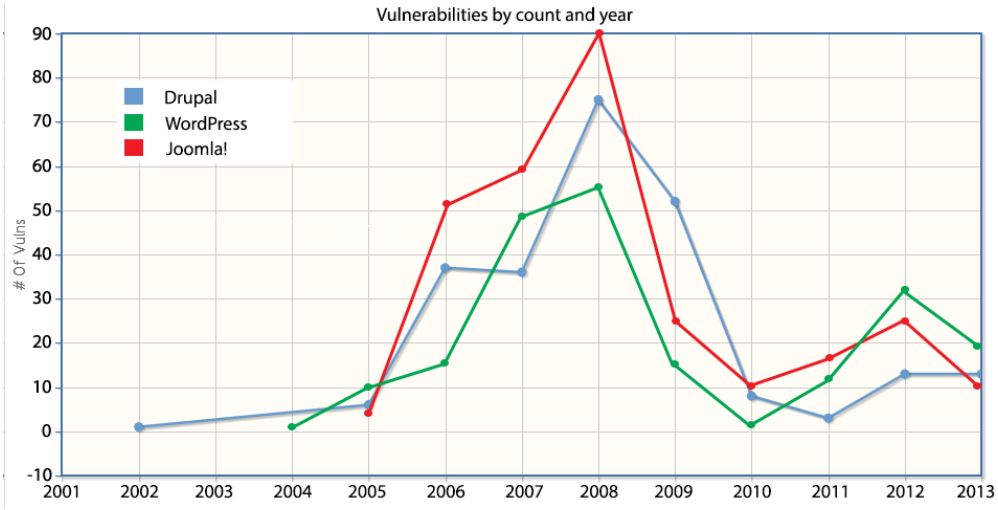
väga vähesel määral läbi vaadatud. Üks selline moodul on kontekstimoodul, mis võimaldab kasutajal jagada oma veebilehe erilaadseteks osadeks. Ründajad saavad sisestada meelevaldset koodi, et rünnata veebilehe administreerimise poolt. See võib muuta kasutaja konto turvalisust ja omakorda viia kolmandate isikute poolt käivitatud PHP-koodini või kasutatakse administraatori õigusi, et rünnata teisi süsteeme pahavaraga. (Klein Keane 2013)



Joonis 19. Drupali nõrkuste statistika CVE järgi (CVE 2014a)

5.4. Kokkuvõte populaarseimate CMS-ide turvalisusest

Kahjuks on kõikidel sisuhaldussüsteemidel turvaauke ja nõrkusi, millele ei ole veel lahendusi leitud. Sisuhaldussüsteemidel on suureks probleemitekitajaks allalaaditavad pluginad ja kohandatavad kujundusteemad. Kui CMS-ide tuuma koodi on võimalik üsna hästi kontrolli all hoida, siis igasuguseid kujundusteemasid ja muud saavad kasutajad ise luua ja teistele jagada. Harjumaa üldhariduskoolide näitel võib aga öelda, et võimalusterikkus on üks olulisemaid põhjendusi CMS-i valikus, teistpidi ka kõige ohtlikum näitaja.



Joonis 20. CMS-ide nõrkused aastate lõikes (CVE Details 2013)

Nagu graafikult näha, tõusis CMS-ide raporteeritud nõrkuste arv 2008. aastal kõige enam Joomlal ja kõige vähem oli neid WordPressil, samas 2013. aastaks on WordPressi nõrkuste arv siiski suurem kui Drupali ja Joomla oma. Nagu ka eelpool mainitud sai, tõestab Drupali turvalisust selle kasutamine suurtel olulistel veebisaitidel, kus päringute arv võib olla väga suur. WordPressi kasuks ei räägi ka pidev uute versioonide väljalaskmine, sest nendega on raske sammu pidada ja versioonid jäävad uuendamata. WordPress ei paranda vigu vanadel versioonidel ja ametlikult toetatakse ainult viimast versiooni. Uue versiooniga tuuakse aga vana versiooni vead välja WordPressi kodulehel ja nende teadmine muudab ründamise pahatahtlikele eriti lihtsaks.

Kui vaadata graafikut (vt Joonis 20) kõikide aastate peale kokku tuvastatud turvaprobleemidest, siis on märgata, et 2010. aastal oli väga vähe raporteeritud turvaauke. Arvatavasti näiteks WordPressi puhul on selle põhjuseks see, et nimetatud aastal ilmus ainult üks WordPressi versioon – 3.0. Selle versiooniga tuli WordPressi palju muudatusi nii sisu osas kui ka uue kujundusteema näol. Suurimaid muudatusi oli arvatavasti see, et enam ei olnud kahte erinevat sisuhaldussüsteemi – WordPress ja WordPress MU (*multiuser*), mis tähendas, et MU tarkvaraga sai mitu eraldiseisvat veebilehte luua, aga andmebaas on originaalsaidiga sama. Selle versiooniga sai WordPressist ühtne süsteem ehk *multisite*.

CVE statistika järgi (vt Joonis 20), kus on vaadeldud kõikide raporteeritud turvanõrkuste arvu, võib näha, et 2008. aastal oli kõikide CMS-ide nõrkuste arv tõusnud tippu. Seda võib põhjendada näiteks üldine majandusseis. Rahaliste raskuste tõttu võidi hakata veebilehti rohkem vabavaralistele CMS-idele ehitama. Samuti võis sellest tuleneda tahe omal käel

hakkama saada ja mitte palgata asjatundjat. Mida rohkem veebilehti neile süsteemidele loodi, seda enam võisid selguda nõrkused.

2013. aasta statistikat vaadates (vt Joonis 20) on näha, et kõige enam turvaprobleeme on avastatud WordPressil. Teisel kohal on Drupal ning kolmandal Joomla. Siin võib märgata selget korrelatsiooni nende sisuhaldussüsteemide populaarsusega samal aastal. Nimelt Builtwithi statistika järgi (vt Joonis 14) oli just WordPress (43,32 %) kõige enam kasutatud CMS, sellele järgnes Drupal (12,20%) ning kolmas oli Joomla (4,71%).

6. Turvalisuse kontrollimine kolme kooli veebilehe näitel

6.1. Meetod

Sihtmärgiks valitakse kolme Harjumaa kooli veebilehed, mis on võimalikult erinevad – tellimusveeb (*custom web*), Joomla ja WordPressi CMS-il. Koolidega võetakse ühendust, et paluda luba nende veebilehtede turvalisuse kontrollimiseks. Loa saamiseks lubatakse, et koolid jäävad anonüümseks. Eesmärk on hinnata nende veebilehtede turvalisust, leida nõrkusi ja neid tõestada. Terminit läbistustestimine (*penetration testing*) kasutatakse tihti tähistades ekslikult igat tüüpi turvalisuse kontrollimist. Turvalisuse kontrollimine (Engebretson 2013, 1–2) on aga laiem mõiste, mis hõlmab lisaks läbistustestimisele ka nõrkuste skaneerimist (*vulnerability scan*), nõrkuste ja turvalisuse hindamist ning turvalisuse ülevaateid ja auditeid.

Selles töös kontrollitakse nende veebilehtede turvalisust, kasutades läbistustestimise manuaal-teste, mis võimaldavad lähenemist jooksvalt muuta ja saada parema üldpildi. Lisaks kasutatakse kahte automaatskannerit, mis on abiks leidmaks nõrkuseid veebilehel. Alustuseks otsitakse infot veebilehtede ülesehituse kohta: kas on kasutatud CMS-i; kui jah, siis millist, kus asub sisselogimisleht jne. Seejärel tehakse erinevaid manuaal-teste, et leida nõrkusi. Leitud nõrkusi püütakse tõestada. Läbistustestimise lõpus koostatakse kokkuvõte, kus tuuakse välja leitud nõrkused ja pakutakse neile lahendusi.

Suureks ohuks on kasutajate turvakäitumine. Selleks, et näha kui lihtne oleks koolitöötajate kaudu veebilehte rünnata, tehakse kahe kooli töötajate seas sotsiaalse manipulatsiooni katse. Töötajatele saadetakse e-kiri ühel juhul ühe töötaja e-posti aadressilt ja teisel juhul kooli domeeni alt väljamõeldud töötaja nimega. Kirjas kutsutakse töötajaid üles lingi kaudu enda andmeid kontrollima. Lingi avamiskorrad loetakse kokku. Eesmärk on tõestada, kui lihtne on võõra aadressi alt lingiga kirju saata, et näiteks andmeid koguda või kirja avaja arvutisse pahavara paigaldada, ja seeläbi veebilehele juurde pääseda.

6.2. Kool 1

Esialgse info kogumiseks kasutatakse veebilehte http://builtwith.com. Sealt on näha, et veebileht ei ole ehitatud sisuhaldustarkvaraga. See tähendab, et veebilehe nõrkuste otsimiseks ei saa kasutada varem raporteeritud turvaauke.

Võib öelda, et testitud veebileht on üsnagi turvaline, kuna see ei ole kirjutatud avatud lähtekoodiga. Võimalus leida turvaauke on palju väiksem. Näiteks kasutatakse koodis palju eestikeelseid muutujate nimesid. See teeb häkkeritele, kellel on väljakujunenud töövõtted, turvaaukude otsimise raskemaks. Siseveebi sisselogimisaknas on POST-parameetrid eesti keeles. Turvaliseks teeb ka see, et veebilehel ei ole kasutatud tekstisisestamise välju. Ainsad väljad on siseveebi sisselogimise lahtrid, kus prooviti kõige lihtsamat SQL-süstimist, paigutades muutujaid ülakomade vahele. Veel üks turvalisuse näitaja on see, et veateateid ei kuvatud. See välistab selle, et ründaja saaks päringuid veateadetes sisalduva info järgi kohandada.

Lehel jooksutati Vega-nimeline automaatskanner turvaaukude leidmiseks. Vega on vaba lähtekoodiga turvalisuse skanner ja testimise platvorm. See leidis testitaval saidil sessiooni küpsiste kasutamises turvaaugu. Nimelt kui arvutid on samas turvamata wifi-võrgus, siis saab häkker kasutaja internetis tegutsemist jälgida. Selleks peab ründaja teada saama, kuidas suhtleb kasutaja arvuti wifiga, mis on krüpteerimata või krüpteeritud ebapiisavalt, näiteks WEP-ühendus. Lisaks on lehe siseveeb HTTP protokolli peal, mitte aga turvalise HTTPS-i peal.

Lehel leiti ka, et veebileht ei kasuta HTTP päises ainult HTTP-l kasutatavat küpsist (*HttpOnly flag*). Kui see oleks lisatud, siis lehel jooksutatud skriptid ei saa seda küpsist kasutada, isegi kui veebilehel on XSS-turvaaugud. Selle kooli veebilehe puhul saaks näiteks häkker lehel pahatahtliku skripti käivitamisel endale administraatoriga sama sessiooni küpsise.

Lehel oli ka intranetti sisselogimise parooli väljal *autocomplete* lubatud. Ehk veebilehitsejal lubatakse parool meelde jätta sellisel kujul nagu jäetakse tavalise vormi väljad meelde. Kuna sisse logimise andmed salvestatakse arvuti kõvakettale ja sellesse konkreetsesse veebilehitsejasse, siis ei ole see väga turvaline lahendus arvestades koolides kasutatavaid jagatud arvuteid.

6.3. Kool 2

Kodulehe esmase info saamiseks vaadati veebilehe lähtekoodi veebilehitsejast, kus `<meta>` *tag*'i all asuv „generaator" andis teada, et selle dokumendi genereeris Joomla tarkvara. Lähtekoodist oli ka näha, et veebilehel kasutatakse vana versiooni 1.5, mida enam ametlikult ei toetata. Kasutusel olev CMS kinnitati ka builtwith.com lehe järgi.

Järgmiseks otsiti Joomla ametlikest versiooniraportitest selle konkreetse versiooni parandatud turvaauke ning katsetati, kas sellel kooli veebilehel õnnestub neid ära kasutada.

Automaatseks veebilehe skannimiseks kasutati kahte tarkvara – Skipfish ning Vega. Mõlemad on avatud lähtekoodiga platvormid turvaaukude leidmiseks ja nende testimiseks.

Skipfishi tulemustelehel oli näha, et veebileht kasutab andmete struktureerimiseks SimplePie nimelist koodi-teeki, mis on kirjutatud PHP-s. Kui nüüd avada programmi poolt leitud SimplePie .php link, siis on sealt näha, milliseid teeke see veebileht kasutab ning häkker saab otsida eraldi nende turvaauke.

Vega skanner leidis, et veebileht ei kasuta HTTP päises ainult HTTP-l kasutatavat küpsist (*HttpOnly flag*), mis teeb võimaluse XSS-rünnakuteks.

Kuna veebilehe lähtekood ega ka Chrome'i moodul, ei andud täpsemat informatsiooni, mis Joomla verisooni veebileht kasutab, siis pidi autor selle teada saamiseks uurima lehel asuvate Joomla tuuma-moodulite versioone. Selle järgi leiti, et veebileht kasutab Joomla 1.5.16 versiooni. Proovides ära kasutada selle konkreetse versiooni raporteeritud nõrkusi, ei leidnud autor turvaauke, mida oleks saanud kõige sagedasemate rünnaku tüüpide jaoks ära kasutada.

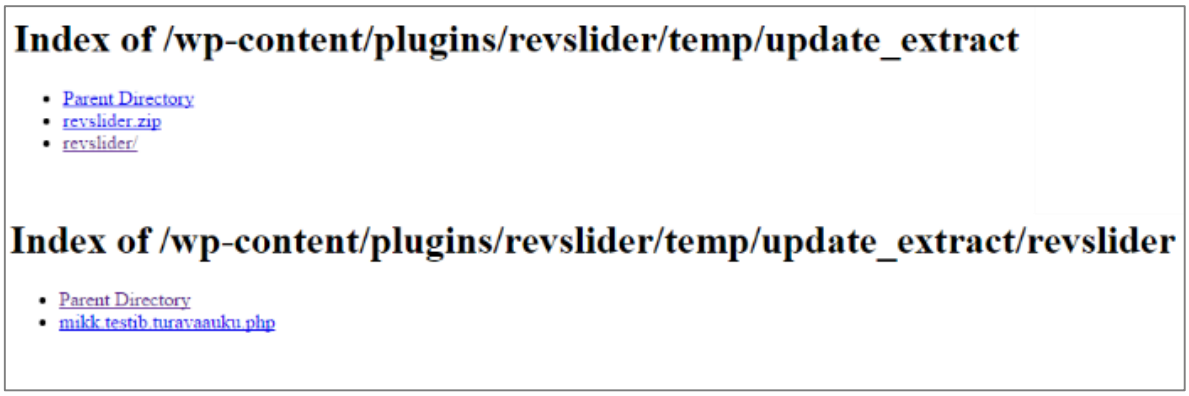
6.4. Kool 3

Autor kasutas sisuhaldustarkvara ja selle täpse versiooni teada saamiseks Chrome'i veebilehitseja moodulit WordPress Version Check. Tuvastati, et veebilehel on kasutusel WordPress 3.9. Veebilehe lähtekoodist oli näha, et lehel on kasutusel moodul RevSlider 3.9. Google'i otsing andis tulemuse, kuidas saaks selle mooduli turvaauke ära kasutada. Esimene katse andis kohe autorile veebilehe andmebaasi seadistamisefaili, kus asus andmebaasi info. Sisestatud URL oli `wp-admin/admin-ajax.php?action=revslider_show_image&img=../wp-config.php`. Veebilehe serveril

ei olnud tulemüüri, aga MySQL'i enda seadistus lubab andmebaasi siseneda ainult *localhost*'st.

Antud veebilehel oli ka jälgi eelnevast rünnakust. Nimelt leiti kaustast `/wp-content/plugins/revslider/temp/update_extract/revslider/` PHP-fail, mis tundus olevat häkkerite poolt jäetud tagauks.

Prooviti ka ise seda turvaauku testida, laadides üles mõni fail. Vähese vaevaga leiti internetist kellegi kolmanda isiku poolt loodud skript, millega saab käsurealt fail üles laadida. Kuna see konkreetne moodul on tehtud selliselt, et `revslider.zip` fail pakitakse automaatselt lahti serverisse lisamisel, siis saab sellesse pakki ära peita mistahes faile. Käsureale sisestati käsk, mis laadis üles ja asendas selle ZIP-faili ning pakkis automaatselt lahti. Käsk oli järgmine: `my $exploit = $ua->post("http://.../wp-admin/admin-ajax.php", Cookie => "", Content_Type => "form-data", Content => [action => "$action", client_action => "update_plugin", update_file => ["$update_file"]]);` (Cid 2014) Joonis 21 näitab serverisse lisatud faili asukohta. Lisatud failidega saaks ründaja teha näiteks XSS-rünnakuid.



Joonis 21. Testitud veebilehe serverisse lisatud fail

Edukast rünnakust kooli veebilehele on teavitatud kooli infojuhti.

6.5. Sotsiaalse manipulatsiooni katse

Arvutikasutajad on üsnagi altid avama võõra päritoluga e-kirju, nende manuseid allalaadima ja nendesse lisatud linke avama. Seda tõestas ka Aavik (2013) Tallinna Kaubamaja kontserni töötajate näitel. Selle töö autor teeb kahe turvalisuse kontrollimises osaleva kooli töötajatega katse, kas võlts-kiri tuntakse ära ja kui valmis nad on tuttavana tunduva e-posti aadressi alt saadetud linki avama.

Kirjutati lihtne programm kasutades Phytonit, mis koosnes algoritmist, kus määrati kirja vormistus ning programmi sisendiks olid kirja saatja, saajad ja sisu. `<href>` atribuudiga määrati ära, mis aadress kuvatakse avajale ning kuhu see link tegelikult viib. Siit võib kohe välja tuua, et üks indikaatoreid, miks see link võiks tunduda kahtlane, on vaadates veebilehitseja all vasakul nurgas HTML-i olekuriba, kui minna lingile hiirega peale. Sealt on näha, et sihtkoha aadress on täiesti erinev sellest, mis on kirjas endas.

Testi eesmärk on tõestada, kui lihtne on võõral inimesel saata kirju koos avatava lingiga ükskõik, kelle nime ja e-posti aadressi alt. Testis saadetud kirja sisuks on teade, et töötajad kontrolliksid oma isikuandmeid. Kirja lisatakse link, mis avab uue lehe, kus on teade, miks selline test tehti. Lingi küljes on loendur, mis saadab lingil käijate arvu Mixpaneli nimelisele analüüsiplatvormile.

Üks sellise võlts-kirja saatmise eesmärk võib olla õngitsemine (*phishing*). Sellise kirjaga oleks näiteks saanud lingis asuvale veebilehele sisestada tõetruu vormi, millega saaks koguda andmeid ja miks mitte kasutajanimesid ning paroole. Selle toimimist on tõestanud nii Aavik (2013) kui näited meediast, millest ühe rängema puhul jagasid 23 eestlast petisele enda pangaparoolid (Kuus 2009).

Samuti saaks lingi kaudu paigaldada kasutaja arvutisse viiruse, mis jälgib tema tegevust veebis ning ka seda, mis klahve klaviatuuril vajutatakse, seega oleks võimalik näha kõiki arvutikasutaja sisestatud paroole, sh veebilehe administreerimiskeskkonna kasutajatunnust ja parooli. Veebilehe administraatorile lingiga vastava viiruse saatmisega, oleks võimalik saada ligipääs tema arvutile, mis tähendab tervele kooli infosüsteemile väga suurt ohtu. Tänapäeval ei pea olema kogemustega häkker, et saaks vajaliku viiruse enda valdusesse. Internetist võib osta toimivaid viiruseid ning õpetuse, kuidas seda kasutada.

1. kooli puhul saadeti võlts-kiri kooli infojuhi nime ja e-posti aadressi alt 69 töötajale. Selle lingi unikaalseid avamisi oli 48. 2. kooli puhul saadeti võlts-kiri aadressilt andmed@“kooli domeeninimi“ ja väljamõeldud töötaja nime alt võlts-kiri 60 töötajale ja selle lingi unikaalseid avamisi oli 31. Kuna koolitöötajad jagavad tihti klassiruumi arvuteid, siis võib eeldada, et lingi avamisi oli veelgi rohkem. Katsest võib siiski järeldada, et kolleegi nime ja töökoha domeeninime alt saadud kiri mõjub usaldusväärsemalt ja seda ollakse eriti altid avama. Väljamõeldud nimega, kuid siiski töökoha domeeninimega e-posti aadress tekitab enam kahtlust. Kooli infojuhid võiksid töötajaid koolitada suhtuma e-kirjades asuvatesse

linkidesse ja manustesse ettevaatusega, võlts-kirju ära tundma ning seda teadmist ka õpilastele jagama.

6.6. Katsete järeldusi

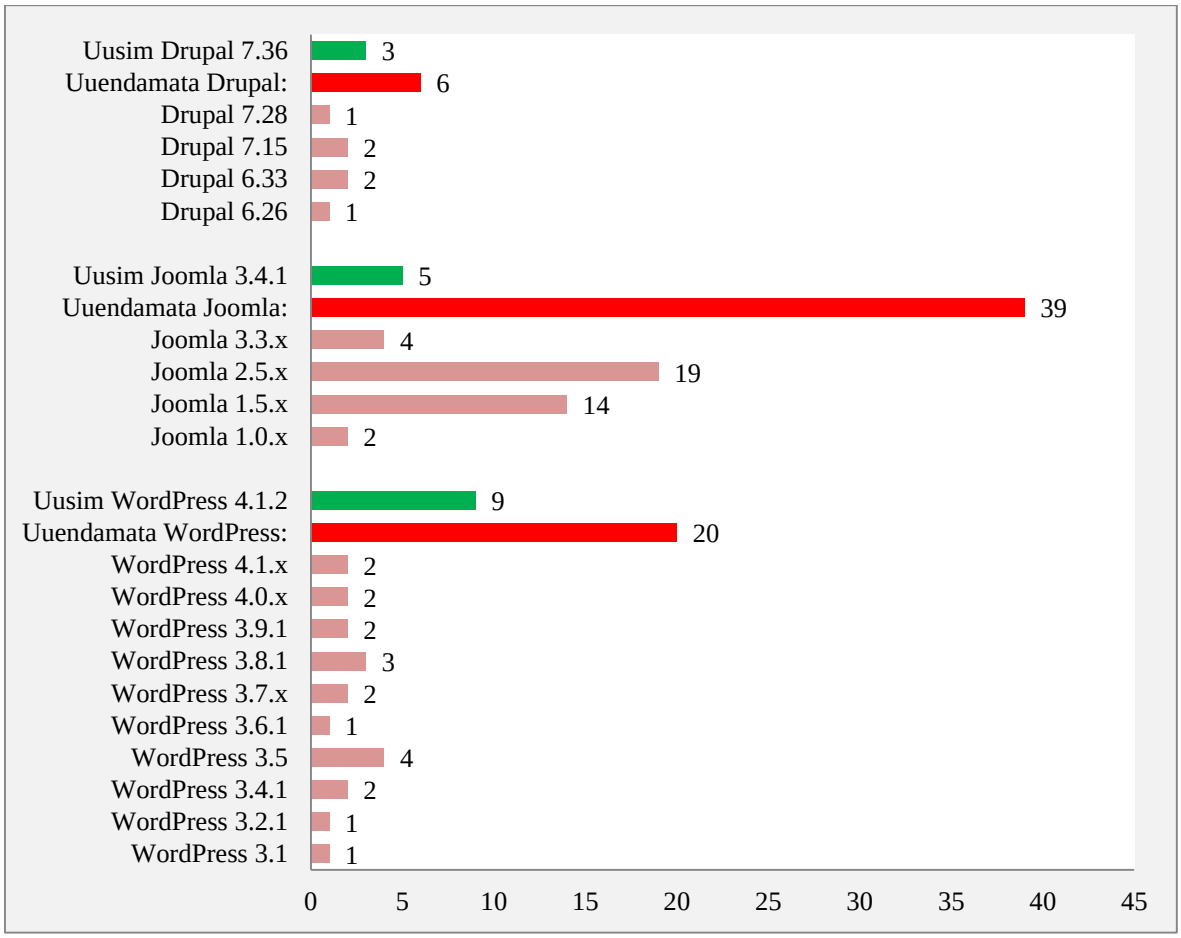
Otsides ja proovides ära kasutada koolide veebilehtedel asuvaid turvaauke, peab autor tõdema, et see ei ole lihtne ülesanne, kui ei ole enne millegi sellisega kokku puutunud. Kasutades otsingumootoreid, tulevad küll enamik raporteeritud turvaaugud välja, aga selleks, et osata neid ära kasutada, peab tundma õigeid tööriistu ja kindlasti aitab eelnev kogemus. Kui tahetakse teha võimalikult palju kahju nii, et konkreetne sihtmärk ei ole oluline, siis on ründajal mõistlikum skaneerida suurt hulka, et nende seast leida otsitud nõrkustega veebilehti.

Autori üllatuseks, ei õnnestunud leida kooli kodulehelt, millel kasutati Joomla 1.5 sisuhaldustarkvara, mitte ühtegi ohtlikku turvaauku. Sellele aitab kaasa see, et seal ei ole kasutusel mitte ühtegi moodulit, mille kohta oleks teada raporteeritud turvaauke. Arvatavasti kõik Joomla tuuma-moodulid, mida lehel kasutatakse, on uuendatud parandustega versiooni peale. Kooli leht, millel ei olnud kasutatud CMS-i osutus üpriski turvaliseks ning turvaaukude otsimine oli palju keerulisem, kui teisel kahel kodulehel. Kuna kolmandal veebilehel olid uuendamata nii WordPress kui ka lehel kasutatavad moodulid, siis õnnestus nende turvaauke ära kasutada.

Kuna uuendamata tarkvara kasutamine on väga sagedane ning sellega võivad kaasneda väga suured ohud, siis otsustati vaadata, mis versioonid sisuhaldustarkvaradest on kasutusel koolidel 24. aprilli seisuga. Uurimise alla võeti 135 Harjumaa kooli ja skannimiseks kasutati Chrome'i veebilehitseja mooduleid, builtwith.com veebilehte ning kodulehe lähtekoodi vaatlust. Nende veebilehtede seast vaadeldi 82 veebilehte, mis olid loodud Drupalile, Joomlale ja WordPressile. Tulemus on nähtav joonisel (Joonis 22), kus rohelisega on märgitud hetkel kõige uuem versioon, punasega uuendamata versioonid kokku ning helepunasega vanemad versioonid eraldi. Lehtedest ainult 20% olid uuendatud viimase CMS-i versiooni peale. 44% kasutusel olevatest versioonidest ei ole enam ametlikult toetatud.

Kokkuvõttes on tulemus üpriski halb ning inimesed ei ole ikka veel teadlikud kui oluline on hoida oma tarkvara koguaeg uusimana. Sisuhaldustarkvarade tootjad rõhutavad seda oma kodulehtedel ja väljaannetes pidevalt. WordPress ja Joomla on selleks eraldi

arendanud ühe-klikiga uuendamise võimaluse, et kasutajad saaksid mugavalt ja kiirelt uuendada oma tarkvara.



Joonis 22. Harjumaa üldhariduskoolide CMS-ide versioonid

Kahele valitud koolile tehti ka sotsiaalse manipulatsiooni katse, mille käigus saadeti töötajatele linki sisaldav võlts-kiri õigena tunduva nime alt. Lingi avajad loeti kokku. Töötajad olid üsna valmis e-postiga saadetud linki avama. See näitab, et turvalisuse puhul ei ole oluline vaid tehniline pool, vaid ka töötajate ja kõigi teiste arvutikasutajate turvateadlikkus, mida oleks võimalik tõsta vastavate koolitustega.

7. Üldisi soovitusi turvariskide vähendamiseks

Arvestades seda kui palju on viimaste aastate jooksul Eesti veebilehti rünnatud, on sisuhaldussüsteemide turvalisuse teema kindlasti aktuaalne. Paljusid rünnakuid saab tihtipeale ennetada. Järgnevalt antakse mõningaid soovitusi, kuidas riske vähendada. Soovitused põhinevad nii isiklikul kogemusel kui kirjandusel.

Mitmed õnnestunud rünnakud on tingitud inimfaktoritest. Veebilehe haldaja teadlik ja aktiivne tegutsemine aitaks riske vähendada. Üheks ohumärgiks võiks pidada näiteks seda, et vabavaralised CMS-id on kõigile lihtsasti kättesaadavad. Sellest võib tuleneda see, et installimist ja kasutamist ei kontrolli mitte ükski IT-alane professionaal, vaid lihtsalt asjast huvitatu. Tasuta kättesaadava CMS-i puhul võib kasutajale tunduda, et ega kaotada pole midagi ja proovib omal käel hakkama saada. IT-teenuseid peetakse üldjuhul väga kulukateks.

Inimfaktorite alla kuuluvad ka näiteks nõrga parooli valimine, samuti parooli jagamine ja teistele kättesaadavasse kohta jätmine. Samuti võiks inimfaktorite alla liigitada vead seadistamisel, CMS-i turvalisemale versioonile uuendamise ja varunduse tegemata jätmise. Varunduse tegemata jätmist ei saa iseenesest pidada nõrkuseks, kuid see oluline faktor muudab lehe rünnakujärgse taastamise palju keerulisemaks. Nagu selgus koolide veebilehtede CMS-ide versioonide uurimisest, on tarkvarade uuendamata jätmine kahjuks väga levinud. Enamasti tulevad sellised hooletusest tingitud vead teadmatusest. Samas võib ka olla põhjuseks see, et kasutajad on teinud ise muudatusi kujundusteemades või kasutusel olevates moodulites ja kardavad, et tarkvara uuendamine rikub nende muudatused ära.

Siinkohal võikski juhtida tähelepanu sellele, kui oluline on tõsta selles valdkonnas inimeste teadlikkust. Enam ei kuulu veebilehtede haldamine ja turvalisuse tagamine vaid arendajate pädevusse, sellega tegelevad ka täiesti teiste valdkondade inimesed.

Teiselt poolt tulenevad rünnakud tehnilistest faktoritest. Siia võiks lugeda seda, kui CMS-id on nõrkuste kaudu lihtsasti rünnatavad. WordPressi ja Drupali puhul on olnud kõige levinumaks ründetüübiks XSS ehk murdskriptimine (vastavalt 34% ja 46%). Siinkohal kordaks veelkord, et esimene samm kõikide nõrkuste ja ohtude minimaliseerimiseks oleks oma sisuhaldustarkvara hoidmine uuendatuna ja teiseks käia tihti oma veebisaidil, et hoida

silma peal kui peaks midagi korrast ära olema. Rääkides aga konkreetsemalt ründetüübist XSS, siis kõige efektiivsem, aga mitte lõplikult kindel moodus ennetada XSS rünnet, on lubada ainult JavaScripti, mis tuleb kasutaja poolt usaldatud saidilt. Soovitav on kasutada oma veebilehitsejate võimalusi ja lisasid, mis kaitsevad selliste rünnakute eest. Näiteks Firefoxil on olemas plugin nimega NoScript, mis lubab läbi ainult valideeritud koodi. Teistel populaarsetel veebilehitsejatel on ka sarnased lisad olemas. (NSA 2011)

Üks enim kasutatavaid ja ohtlikumaid ründemeetodeid on SQL-süstimine ja peamine eksimus, mida tehakse, et ründajad saaksid süstida SQL-koodi on see, et veebilehel lubatakse kasutada kontrollimata ja valideerimata sisendeid. Selleks, et sellist rünnakut ära hoida, peaks rakendus näiteks kõik erilise tähendusega tähemärgid – nagu ühekordsed ja topelt jutumärgid, semikoolon, kommenteerimismärgid – ükshaaval valideerima.

Teine võimalus SQL-süstimise ärahoidmiseks on mitte kunagi liita kasutajate sisendit rakenduse SQL-lausega, mis saadetakse andmebaasi päringuks. Lihtne moodus selle saavutamiseks on kasutada parameetritega päringuid. Need päringud on sellised, kus SQL-i muutujate osad asendatakse määrajatega (tavaliselt „?“). Näiteks selle asemel, et päringut sooritada selliselt: `SELECT email FROM users WHERE email = '<user_input>'`, võiks sooritada nii: `SELECT email FROM users WHERE email = ?`. (OWASP 2013e)

Joomla puhul on omavoliline koodi käivitamine (40%) aga kõige levinum (Kerner 2013). See võib tuleneda sellest, et enne 2011. aastat oli Joomla suurteks nõrkusteks pluginad. Kuna Joomla puhul ei olnud nende arendamine piisavate turvanõuetega, siis oli pluginate kaudu ründeid väga palju ning just nende abil on mugav kasutada koodi käivitamist ja SQL-süstimist. Siinkohal võibki tuua kõige paremaks lubamatu koodi käivitamise ründe ära hoidmiseks selle, et pluginate allalaadimisel ja enne CMS-iga liitmist, tuleks veenduda plugina päritolus ja sobivuses.

Kuid ükskõik, mis CMS-i poolt kasutaja otsustab, oleks põhiline, et ta vähendaks võimalikult palju inimfaktoritest tulenevaid turvariske ja oleks teadlik valitud sisuhaldustarkvara turvaaukudest ning muudest ohtudest.

Kokkuvõtlikult on kõige olulisemad soovitused CMS-i turvariskide vähendamiseks järgnevad:

ASUTUSELE:

- Infoturbe eeskiri. Paljudes asutustes on kodukorra eeskiri, millega tutvumine on kõigile töötajatele kohustuslik. Selle kõrval võiks olla ka infoturbe eeskiri.
- Koolitused. Asutused võiksid hoolitseda töötajate turvateadlikkuse eest. Koolide puhul takistab seda ilmselt vähene rahastus, seega võiks riigi poolt olla korraldatud kasvõi iga paari aasta tagant veebikoolitus koos testiga. Ühekordne koolitus ei oleks piisav, sest tegu on valdkonnaga, mille kiire areng on märkimisväärne. Ruth Randoja (2012) juhtis enda diplomitöös tähelepanu õpetajatele koolitusvajadusele selles valdkonnas ja töötas selleks välja ka mudeli.
- Turvalise veebilehe loomine. Nagu koolide infojuhtide seas tehtud küsitlusest selgus, siis 67% infojuhte lõid veebilehe ise (või endine töötaja) ja 11% koolidest lasid veebilehe luua lapsevanemal, õpilasel või lihtsalt tuttaval. See on täiesti arusaadav, et koolile on väga kulukas tellida veebileht vastava valdkonna tunnustatud ettevõttelt. Siinkohal võiks jällegi loota riigi abile, panustades näiteks HAVIKESe teenusesse.

SISUTOIMETAJALE:

- Kasuta tugevaid paroole, vaheta neid regulaarselt ning ära jaga neid teistega
- Suhtu e-kirjaga saadetud andmepäringutesse, linkidesse ja manustesse ettevaatusega
- Käi probleemide märkamiseks tihti oma veebilehel
- Ära kliki kahtlastel hüpikakendel

ADMINISTRAATORILE JA ARENDAJALE (lisaks sellele, mida soovitati sisutoimetajale):

- Veendu, et kasutad alati kõige uuemat CMS-i versiooni
- Konfigureeri veebisaidi seadeid korrektselt
- Võimaluse korral muuda eelseadistatud administreerimislehe URL
- Veendu pluginate päritolus võimalikult palju taustainfot otsides ja teiste kogemusi uurides

- Veendu, et HTML-kood on korrektne
- Luba ainult JavaScripti, mis tuleb kasutaja poolt usaldatud saidilt
- Kasuta veebilehitsejate lisasid, mis lubavad läbi ainult valideeritud koodi
- Kasuta parameetritega määratletud SQL-päringuid
- Kasuta ainult valideeritud ja kontrollitud SQL-sisendeid
- Varunda regulaarselt serveris olevaid faile
- Jaga õiguseid võimalikult piiratult.

Kokkuvõte

Eesti meedias on leidnud aina enam kõlapinda rünnakud riigi- ja kohalike omavalitsuste asutuste vastu. Levinud on rünnakud Eesti haridussüsteemiga seotud lehtedele, nii koolide veebilehtede, e-kooli, munitsipaalhuvikoolide, õppeprogrammide kui ka muuda lehtede vastu. Zone-h statistika järgi oli .ee domeenil asuvatest sel aastal rünnatud veebilehtedest iga 16. haridussüsteemiga seotud leht. Rahvusvaheliste näidete ja ka koolide infojuhtide vastuste põhjal võib öelda, et ründajad on aasta-aastalt aina nooremad.

Eesti riigi tegevus haridusasutuste IKT küsimustes on väga lai, kuid koordineerimata. Eesti põhikooli- ja gümnaasiumiseadus sätestab kooli veebilehe kohustuslikkuse, kuid seaduses pole selle loomine, haldamine, tehniline teostus ega muu fikseeritud. Tallinna Haridusamet on veebilehti puudutava standardi koostamise ja kõigile kättesaadava majutusvõimaluse pakkumise märkinud enda 2011.–2015. aasta IKT programmi tegevuskavasse, kuid neid võimalusi veel ei ole. Samas EENet juba pakub teenust HAVIKE, mis annab haridusasutustele võimaluse tasuta veebimajutuseks ning Joomlat või WordPressi kasutavale veebilehele, kuid kuna EENeti strateegiast ilmneb, et „ei ole selget ülevaadet taristu arendusplaanidest“, siis jääb küsitavaks, kas koolidele peaks selle teenuse kasutamist soovitama.

Selleks, et saada üldpilt Harjumaa üldhariduskoolide veebilehtede turvalisuse hetkeseisust ja töötajate turvateadlikkusest, tehti nende veebilehtedega tegelejate seas küsitlus. Samuti uuriti, kas plaanitud standardimissoov on vastavuses koolitöötajate nägemusega.

Kahjuks on kõikidel sisuhaldussüsteemidel turvaauke. Paljud neist lahendatakse uute versioonidega, seega uuendamata jätmine on eriti ohtlik, sest info vana versiooni turvaaukude kohta on avalik. Suureks probleemitekitajaks on allalaaditavad pluginad ja kohandatavad kujundusteemad, mille päritolu ja sobivust tuleks kontrollida, näiteks uurides teiste kasutajate kogemust. CMS-ide rünnatavus on selges korrelatsioonis populaarsusega. Levinud CMS-i kasutades on lihtne tekkivatele küsimustele vastuseid leida, kuid samas on neid ka avaliku info tõttu lihtsam rünnata.

Selle bakalaureusetöö raames prooviti eelneval kokkuleppel saidi omanikuga rünnata kolme Harjumaa kooli veebilehte. Autori valitud veebilehtede seast leiti kõige enam turvaauke WordPressile loodud lehelt, mis ise ja ka selle moodulid olid uuendamata. See

rünnaku katse osutus edukaks. Sellel lehel oli jälgi ka eelnevast rünnakust. Autor saatis kolmele katses osalenud koolile kokkuvõtte nende veebilehtede turvalisusest.

Kahele valitud koolile tehti ka sotsiaalse manipulatsiooni katse, mille käigus saadeti töötajatele linki sisaldav võlts-kiri õigena tunduva nime alt ja lingi avajad loeti kokku. Töötajad olid üsna valmis e-postiga saadetud linki avama. See näitab, et turvalisuse puhul ei ole oluline vaid tehniline pool, vaid ka töötajate ja kõigi teiste arvutikasutajate turvateadlikkus, mida oleks võimalik tõsta vastavate koolitustega.

Koostatud statistikast selgus, et 80% Harjumaa üldhariduskoolide veebilehtedest, mis on tehtud populaarseimate CMS-ide peale, on uuendamata.

Veebilehtede turvariskid on seotud haldajate hoolimatusest või teadmatusest tingitud inimfaktoritega (näiteks nõrkade paroolide valimine, tarkvara uuendama jätmine) ning tehniliste vigadega (valideerimata ümbersuunamised, kasutaja poolt antud sisendite vähene kontrollimine). Riske saab maandada järgides bakalaureusetöö viimases peatükis välja toodud soovitusi.

Ingliskeelne resümee

Security of Content Management Systems:

The Case of General Education Schools' Websites in Harjumaa

More and more websites are built on content management systems (CMS), so their security is becoming a vital question to developers, content managers and other interested parties. This thesis is focused on Estonian schools' websites, because attacks against them have been widely reflected on the media. The result of the thesis is directed to people managing school websites – either on a national or a school level.

Cyber-attacks against Estonian national and local authority websites have been brought forward increasingly; also against websites associated with Estonian school system: school, e-school, activity school, learning program websites etc. The Zone-h statistics show that every 16th defaced website on the .ee domain was connected to the school system. In Estonia, programming is being taught already in elementary school, also international examples and the judgement of school employees show that the attackers are younger each year. This means that school information systems must improve alongside with the students' knowledge.

Estonia's national activity in the field is quite wide. The law enacts the school's obligation to obtain a website but does not fixate its technical requirements. There are many institutions and organisations occupied with schools' information and communication systems, but they are not well coordinated, the funding is not certain and the strategies are not met. E. g. Tallinn Education Department had set a goal to implement a standard and provide all Tallinn's schools a united web platform and web page hosting, but improvement is not visual. Their goal is peculiar, because actually the Estonian Education and Research Network EENet already provides schools with free of cost web services like hosting and platform set-up. Unfortunately, security is not their main objective, and collaboration with Estonian Information System Authority RIA is weaker than should.

To get an overview of the security of general education schools' websites in Harjumaa and the knowledge of the matter at hand, the author conducted a survey among school

employees dealing with websites. The objective was also to research whether the wish for standardisation was in accordance with the vision of the responders.

The survey was sent to 136 employees of general education schools that were dealing with websites (information manager). Responses came from 37 schools. 89.2% of the responders assured, that their websites were built on a CMS. The most popular were the free of charge systems: Joomla! (42.9%), WordPress (28.6%) and Drupal (11.4%). The importance of school websites' security was assessed with 8.97 out of 10. But only one school mentioned security as the reason of choosing the CMS. The choice was mostly explained by simplicity and being full of possibilities. In spite of ranking the importance of security very high, the knowledge of its certain measures was low. The most named measure was updating (10 times). Difficult passwords and their renewing were mentioned 7 times. Keeping the number of people with access and administrative rights low was mentioned 6 times. For instance, back-up was mentioned only once. Almost every second responder knew of attacks made at their website. 67% of responders assured that their website was created by an employee. Outright 11% responded that their site was constructed by a student, parent or an acquaintance, but still the desire for standardisation was consentaneous. Taking into account that the importance of school websites' security was ranked very high but the knowledge of certain measures was quite low, training would be beneficial.

Unfortunately, all content management systems carry vulnerabilities. Many of them are patched with new versions, so leaving the system not updated is especially dangerous, because the information about the old version's vulnerabilities is public. Dangerous exploits may occur with plugins and themes, which origin and functionality should always be checked for example by researching the experience of other users. There is a definite correlation between statistics of attacking and the popularity of the CMS. It is easy to get support, if you using a popular CMS, but they are more convenient to attack due to all the information available.

Within the framework of the thesis, attempts were made to attack three general education schools' websites in Harjumaa. The chosen sites differed from each other – a custom web, a site running on Joomla! and a site on WordPress. The search for vulnerabilities was not easy, because the author had not been introduced to hacking beforehand. The knowledge of the right tools is vital. The most damage could be done when the target is not certain, so a large amount of websites can be scanned for vulnerabilities and to attack the ones with

found threats. Out of three targets, the author found the one built on WordPress the most vulnerable. The site itself and its modules were not updated. There were even signs of a former unrelated attack. Author sent a summary of the website security to the schools that participated in the penetration testing.

With two of the chosen schools a social engineering experiment was carried out. The employees were sent a URL-containing fake e-mail 1) under a name of another employee and 2) under a familiar seeming alias. The clicks on the URL were counted. The employees were quite willing to open the URL in the letter that could have possibly contained malware etc. It shows that technical side of security is not the only important aspect – the security knowledge of each computer user is also important and could be improved by training.

The created statistics show that 80% of the General Education Schools' Websites in Harjumaa that are built with popular content management systems, are not updated to latest version.

The security risks of websites are connected to the negligence and ignorance of website developers and also to human factors of its managers (choosing weak passwords, not updating) and technical mistakes (unvalidated redirects and forwards, poor SQL inputs control etc.). The risks could be reduced following recommendations made in the last chapter of the thesis.

Kasutatud kirjandus

Aavik, Kädi 2013. Kasutajate turvakäitumise uuring ja koolituskava tallinna kaubamaja kontserni näitel. https://itcollege.ois.ee/diploma (Accessed April 17, 2015).

Actionnews 2015. Oakleaf High School website hacked. http://www.actionnewsjax.com/news/news/local/oakleaf-high-school-website-hacked/njmCk (Accessed April 7, 2015).

Arvutikaitse 2015. Botnet. http://www.arvutikaitse.ee./?page_id=826 (Accessed May 2, 2015).

Brecht, Daniel 2013. 10 Common Intranet Security Issues: Is Your Business At Risk? *Bright Hub*. http://www.brighthub.com/computing/enterprise-security/articles/104796.aspx (Accessed April 12, 2015).

Builtwith 2013. CMS Usage Statistics. http://trends.builtwith.com/cms# (Accessed December 19, 2013).

Cid, Daniel 2014. RevSlider Vulnerability Leads To Massive WordPress SoakSoak Compromise. https://blog.sucuri.net/2014/12/revslider-vulnerability-leads-to-massive-wordpress-soaksoak-compromise.html (Accessed April 25, 2015).

CVE 2013b. Frequently Asked Questions. http://cve.mitre.org/about/faqs.html (Accessed December 16, 2013).

CVE 2014a. Drupal: Vulnerability Statistics. http://www.cvedetails.com/vendor/1367/Drupal.html (Accessed April 25, 2015).

CVE Details 2013. Vulnerability Distribution. http://www.cvedetails.com/ (Accessed December 10, 2013).

CVE Details 2014a. Joomla: Vulnerability Statistics. http://www.cvedetails.com/vendor/1367/Drupal.html (Accessed April 22, 2015).

CVE Details 2014b. Wordpress: Vulnerability Statistics. http://www.cvedetails.com/vendor/2337/Wordpress.html (Accessed April 23, 2015).

DM Consultancy The Most Commonly Known Joomla Security Issues. http://www.webdevelopmentconsultancy.com/joomla-security/common-security-issues.html?start=1 (Accessed November 27, 2013).

EENet 2014. EENeti strateegia 2015-2010. http://www.eenet.ee/EENet/assets/docs/EENeti_strateegia_2015-2020.pdf (Accessed April 12, 2015).

Engebretson, Patrick 2013. *The Basics of Hacking and Penetration Testing: Ethical Hacking and Penetration Testing Made Easy*. Elsevier.

Groom, Nelson 2014. Meet the 13-year-old boy who's been dubbed China's 'hacking prodigy.' *Mail Online*. http://www.dailymail.co.uk/news/article-2784488/Meet-13-year-old-boy-hacked-school-computer-online-store-insists-hes-using-powers-good-just-trying-fix-websites.html (Accessed April 7, 2015).

HAVIKE HAVIKEse kirjeldus. http://havike.eenet.ee/HAVIKE/index.php/kirjeldus (Accessed April 12, 2015a).

HAVIKE Juhend administraatorile. http://havike.eenet.ee/HAVIKE/index.php/tarkvara/joomla/joomlamigratsioon (Accessed April 12, 2015b).

Himma, Marju 2014. Küberrünnaku alla sattunud eKooli töö on taastatud. *Uudised | ERR*. http://uudised.err.ee/v/613e1297-5749-4e0f-b333-b64789f38176 (Accessed April 7, 2015).

HITSA 2015. HITSA struktuuriüksused. http://www.hitsa.ee/sihtasutusest/struktuur (Accessed April 12, 2015).

HITSA *HITSA strateegia ja visioon*. http://www.hitsa.ee/sihtasutusest/visioon (Accessed April 12, 2015).

Horm, Ivari 2012. *Vaba tarkvara kasutamine võrgu teenuste osutamisel Tallinna munitsipaalkoolides*. Tallinn: Tallinna Tehnikaülikool.

Itoctopus 2012. How to Prevent SQL Injection in Joomla. http://www.itoctopus.com/how-to-prevent-sql-injection-in-joomla (Accessed December 16, 2013).

Jacoby, David 2010. Mass Defacements: the tools and tricks. http://securelist.com/analysis/36356/mass-defacements-the-tools-and-tricks/ (Accessed April 24, 2015).

Jeavons, Benjamin James ja Gregory James Knaddison 2010. *Drupal Security White Paper*. http://drupalsecurityreport.org/sites/drupalsecurityreport.org/files/drupal-security-white-paper-1-1.pdf (Accessed December 1, 2013).

Joomla! 2011. Joomla! 1.5.25 Released. http://www.joomla.org/announcements/release-news/5393-joomla-1525-released.html (Accessed April 12, 2015).

Kerner, Sean Michael 2013. HP Security Report: What is the Most Insecure CMS? http://www.esecurityplanet.com/news/article.php/3929901/HP-Security-Report-What-is-the-Most-Insecure-CMS.htm (Accessed December 15, 2013).

Kierznowski, David 2013. Survey Finds Most WordPress Blogs Vulnerable. http://blogsecurity.net/wordpress/articles/article-230507/ (Accessed November 12, 2013).

Klein Keane, Justin C. 2013. Drupal core XSS vulnerability. http://seclists.org/fulldisclosure/2013/Aug/158 (Accessed December 17, 2013).

Kovacs, Eduard 2013. Softpedia Interview: Alberto Redi, Head of Zone-H. *Softpedia*. http://news.softpedia.com/news/Softpedia-Interview-Alberto-Redi-Head-of-Zone-H-359499.shtml (Accessed April 7, 2015).

Kuus, Agnes 2009. 3 500 000 krooni oli peidus köögikapi taga. http://www.ohtuleht.ee/354624/3-500-000-krooni-oli-peidus-koogikapi-taga (Accessed April 17, 2015).

Lehesalu, Uku 2007. *PHP/SQL süstimine*. Tartu Ülikool. Matemaatika-informaatikateaduskond. Arvutiteaduse instituut.

Matthew, Sam 2015. Islamic extremists hack websites belonging to school and church. *Mail Online*. http://www.dailymail.co.uk/news/article-2898635/Islamic-extremists-hack-websites-primary-school-church-Yorkshire-replace-homepages-hate-message-against-U-S-Israel.html (Accessed April 7, 2015).

Mekaia 2013. Estonian web CMS market overview in March 2013. http://www.saurus.info/estonian-web-cms-market-overview-in-march-2013/ (Accessed November 15, 2013).

Mersereau, Dennis 2015. Angry Student Hacks County's Website. *The Vane*. http://thevane.gawker.com/angry-student-hacks-countys-website-to-apologize-for-sn-1677837740 (Accessed April 7, 2015).

Microsoft Site Security Planning. Threats from Outside. https://technet.microsoft.com/en-us/library/cc958365.aspx (Accessed April 12, 2015).

NovaInfosec 2009. Why Intranets Aren't As Safe As Everyone Thinks They Are. https://www.novainfosec.com/2009/04/15/why-intranets-aren%E2%80%99t-as-safe-as-everyone-thinks-they-are/ (Accessed April 12, 2015).

NSA 2011. Protect Against Cross Site Scripting (XSS) Attacks. http://www.nsa.gov/ia/_files/factsheets/xss_iad_factsheet_final_web.pdf (Accessed January 1, 2014).

OWASP 2013a. About The Open Web Application Security Project. https://www.owasp.org/index.php/About_OWASP (Accessed November 28, 2013).

OWASP 2013b. Broken Authentication and Session Management. https://www.owasp.org/index.php/Broken_Authentication_and_Session_Management (Accessed October 1, 2014).

OWASP 2013c. Cross-Site Request Forgery (CSRF). https://www.owasp.org/index.php/Cross-Site_Request_Forgery_%28CSRF%29 (Accessed January 10, 2014).

OWASP 2013d. OWASP Top Ten Project. https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project#tab=Project_Details (Accessed December 4, 2013).

OWASP 2013e. SQL Injection Prevention Cheat Sheet. https://www.owasp.org/index.php/SQL_Injection_Prevention_Cheat_Sheet#Defens

e_Option_1:_Prepared_Statements_.28Parameterized_Queries.29 (Accessed December 25, 2013).

OWASP 2013f. Top 10 2013-A5-Security Misconfiguration. https://www.owasp.org/index.php/Top_10_2013-A5-Security_Misconfiguration (Accessed January 12, 2014).

OWASP 2013g. Top 10 2013-A7-Missing Function Level Access Control. https://www.owasp.org/index.php/Top_10_2013-A7-Missing_Function_Level_Access_Control (Accessed January 10, 2014).

OWASP 2013h. Top 10 2013-A9-Using Components with Known Vulnerabilities. https://www.owasp.org/index.php/Top_10_2013-A9-Using_Components_with_Known_Vulnerabilities (Accessed January 10, 2014).

Postimees 2006. Häkkerid ründasid Eesti koolide veebilehti. http://www.postimees.ee/1600325/hakkerid-rundasid-eesti-koolide-veebilehti (Accessed April 7, 2015).

Pynnonen, Jouko 2014. WordPress 3 persistent script injection. http://seclists.org/fulldisclosure/2014/Nov/62 (Accessed April 12, 2015).

Randoja, Ruth 2012. Internetiturvalisuse koolitus õpetajatele. https://itcollege.ois.ee/diploma (Accessed April 18, 2015).

RIA 2009. IT-turbejuhend. Lühiülevaade olulisematest IT-turbe alastest turvameetmetest. https://www.ria.ee/public/ISKE/Infoturbe_soovituste_juhend_v1.pdf (Accessed April 12, 2015).

RIA 2012. Infosüsteemide turvameetmete süsteem ISKE. Infosüsteemide turvameetmete süsteem ISKE (Accessed April 12, 2015).

RIA Riigi Infosüsteemi Ameti üldinfo. https://www.ria.ee/ (Accessed April 12, 2015).

Riigi Teataja 2010. Põhikooli- ja gümnaasiumiseadus. https://www.riigiteataja.ee/akt/110072012020?leiaKehtiv (Accessed April 7, 2015).

Roon, Maarja 2014. E-kool oli küberrünnaku tõttu pea kolm tundi rivist väljas. *Uudised | ERR*. http://uudised.err.ee/v/35959ef3-3584-4e47-b84e-63f5d5b11072 (Accessed April 7, 2015).

Roosaar, Veste 2012. Programmeerimine jõuab iga koolilapseni. http://www.parnupostimees.ee/957428/programmeerimine-jouab-iga-koolilapseni (Accessed April 7, 2015).

Rothstein, David 2015. Releases for Drupal core. https://www.drupal.org/node/3060/release (Accessed April 20, 2015).

Schwartz, Mathew 2013. WordPress Site Hacks Continue. http://www.informationweek.com/security/attacks/wordpress-site-hacks-continue/240162060 (Accessed August 11, 2013).

Smith, Katlyn 2014. 2 Bartlett High School students hack into Elgin U-46 portal. *Daily Herald*. http://www.dailyherald.com/article/20141219/news/141218153/ (Accessed April 7, 2015).

TAAT TAATi üldinfo. http://taat.edu.ee/main/ (Accessed April 12, 2015).

Tallinna Haridusamet 2010a. IKT nõukogu koosoleku protokoll nr 1.-4/138. http://www.tallinn.ee/est/haridusasutused/g7436s54244 (Accessed April 12, 2015).

Tallinna Haridusamet 2010b. Tallinna munitsipaalharidusasutuste infotehnoloogilise keskkonna programm aastateks 2011–2015. http://www.tallinn.ee/est/haridusasutused/ikt_programm_2011-2015_.pdf (Accessed April 7, 2015).

Tallinna Haridusamet 2012. Tallinna koolide kodulehekülgede vormistamine. http://tallinnadhs.ml.ee/atp/?c_tpl=1092&command=details&pealkiri=lehek%FClg&juurdepaasupiirang=avalik&dok_id=1807616 (Accessed April 7, 2015).

Tohvelmann, Ivar 2011. Avaliku sektori veebilehtede käideldavuse uuring 2010. https://www.ria.ee/public/Programm/veebideuuring_aruanne_final.pdf (Accessed December 18, 2013).

Tokareva, Irina 2010. Häkker tekitas Narva koolile 300 000-kroonise arve - Krimi - Postimees.ee. *Postimees*. http://www.postimees.ee/214131/hakker-tekitas-narva-koolile-300-000-kroonise-arve (Accessed April 7, 2015).

W3Techs Usage of content management systems for websites. http://w3techs.com/technologies/overview/content_management/all (Accessed November 9, 2013).

Wikipedia 2013. Murdskriptimine. http://et.wikipedia.org/wiki/Murdskriptimine (Accessed December 1, 2013).

WordPress Updating WordPress. http://codex.wordpress.org/Updating_WordPress (Accessed November 15, 2013).

WordPress.org 2015. WordPress Versions. https://codex.wordpress.org/WordPress_Versions (Accessed April 22, 2015).

Lisa 1. Koolide veebilehtede turvalisuse küsitlus

1. Kas Teie kooli veebileht on ehitatud sisuhaldustarkvarale (CMS)? Näiteks Saurus, Drupal, Joomla, WordPress vm.

- Jah
- Ei
- Ei oska öelda

2. Kui jah, siis mis sisuhaldustarkvara kasutate?

- WordPress
- Joomla
- Drupal
- Saurus
- Hansanet
- Edicy
- Muu (palun täpsustage)

3. Mille põhjal Teie kool just selle CMS-i valis?

4. Kes selle veebilehe tegi (ettevõtte nimi, kooli töötaja, tuttav vm)?

5. Kas Teie kooli veebileht on seotud ka intranetiga (sisevõrguga)?

- Jah
- Ei
- Ei oska öelda

6. Kui rahul olete oma kooli veebilehe haldamisvõimalustega?

1 2 3 4 5 6 7 8 9 10

7. Palun põhjendage oma vastust

8. Mitmel inimesel on ligipääs Teie kooli veebilehe sisu toimetamiseks (administreerimiseks)?

- Ei oska öelda, ostame sisutoimetaja (administraatori) teenust sisse
- 1
- 2-3
- 4-6
- 7-9
- 10-...

9. Mitu inimest on veebilehte viimase kahe aasta jooksul uuendanud?

- Ei oska öelda, ostame sisutoimetaja (administraatori) teenust sisse
- 1

- 2-3
- 4-6
- 7-9
- 10-...

10. Mis veebimajutuskeskkonnas (serveriteenuse pakkuja juures) asub Teie kooli veebileht?

11. Hinnake, kui head erialased teadmised on Teie kooli veebilehe sisutoimetaja(te)l. Ka juhul, kui seda teete just Teie.

1 2 3 4 5 6 7 8 9 10

12. Kui oluliseks peate koolide veebilehtede turvalisust?

1 2 3 4 5 6 7 8 9 10

13. Mida teete selleks, et tagada oma kooli veebilehe turvalisust?

14. Kas ja milliseid turvaprobleeme on Teie kooli veebilehel esinenud? Näiteks, kas keegi võõras on sinna kunagi tahtmatut infot sisestanud, kas andmebaasile on ligi pääsetud, kas koduleheküIg on kunagi pahatahtlikult üle koormatud või maha võetud jne. Põhjalikud vastused on eriti oodatud.

15. Kas Teie meelest peaks olema Eesti haridusasutuste veebilehtede koostamine ja haldamine standarditud?

- Jah
- Ei
- Ei oska öelda

16. Palun jätke siia kommentaare ja lisainfot