Suosituimpien sisällönhallintajärjestelmien tietoturvallisuuden vertaaminen

Kasperi Hakokivi

Opinnäytetyö, AMK
Toukokuu 2022
Tietojenkäsittely ja tietoliikenne
Insinööri (AMK), tieto- ja viestintätekniikka

jamk | Jyväskylän ammattikorkeakoulu
University of Applied Sciences

jamk

Kuvailulehti

Kasperi Hakokivi

Suosituimpien sisällönhallintajärjestelmien tietoturvallisuuden vertaaminen

Jyväskylä: Jyväskylän ammattikorkeakoulu Toukokuu 2022, 39 sivua

Tietojenkäsittely ja tietoliikenne. Tieto- ja viestintätekniikan tutkinto-ohjelma. Opinnäytetyö Amk.

Julkaisun kieli: suomi

Verkkojulkaisulupa myönnetty: kyllä

Tiivistelmä

Opinnäytetyön lähtökohtana oli oma mielenkiinto aiheeseen. Opinnäytetyön ajatuksena oli selvittää sisällönhallintajärjestelmien tietoturvaa ja vertailla niitä keskenään. Asiaa miettimällä päädyttiin moneen keinoon mitata sisällönhallintajärjestelmien tietoturvallisuutta ja keinoja verrata niitä keskenään. Näin saatiin aikaan selkeitä ja hyödyllisiä tuloksia.

Työssä tutkittiin ja vertailtiin valittujen sisällönhallintajärjestelmien, WordPressin, Joomlan ja Drupalin tietoturvallisuutta käyttäen laadullisia, määrällisiä ja vertailevia tutkimusmenetelmiä. Vertailtavina ominaisuuksia tarkasteltiin esimerkiksi järjestelmien kokonaishaavoittuvuuksien määrää, haavoittuvuuksien vakavuuden keskiarvoa ja järjestelmien tietoturvatiimiä. Opinnäytetyössä annettiin myös vinkkejä verkkosivuston turvallisuuden parantamiseen. Työn tiedonkeruu lähteinä käytettiin nettijulkaisuja ja -artikkeleja. Tulosten perusteella suoritettiin sisällönhallintajärjestelmien tietoturvallisuuden vertailu ja päädyttiin johtopäätökseen.

Tuloksien perusteella Drupal sisällönhallintajärjestelmällä oli selkeästi paras tietoturva. Se pärjäsi parhaiten melkein kaikissa vertailtavissa ominaisuuksissa. Tuloksista selvisi myös, että sisällönhallintajärjestelmän tietoturvallisuudessa ei ole pelkästään kyse vain alustan valinnasta. Verkkosivuston tietoturvallisuus on sen omistajan vastuulla ja sisällönhallintajärjestelmä tarjoaa vain työkalut turvallisen nettisivuston rakentamiseen. Opinnäytetyön tuloksista on varmasti hyötyä uuden verkkosivuston rakentamista pohtiville kehittäjille.

Avainsanat (asiasanat)

Tietoturvallisuus, sisällönhallintajärjestelmä, verkkosivusto, haavoittuvuudet

Muut tiedot (salassa pidettävät liitteet)

jamk

Description

Kasperi Hakokivi

Comparison of the most popular content management system by information security

Jyvaskyla: JAMK University of Applied Sciences, May 2022, 39 pages

Data processing and telecommunications. Degree program of information- and communication technology. Bachelor's thesis

Permission for web publication: Yes

Language of publication: Finnish

Abstract

The starting point of the thesis was my own interest in the topic. The idea of the thesis was to compare the information security of content management systems and compare them. Thinking about it, we came up with many ways to measure information security of management systems, and ways to compare them. This produced clear and useful results.

In the work, we researched and compared the information security of the selected content management systems, WordPress, Joomla and Drupal using a qualitative, quantitative and comparative research methods. The comparable features were, for example, the total number of vulnerabilities in the systems, the average severity of the vulnerabilities, and the systems security team. The thesis also provided tips for improving the security of the website. Online publications and articles were used as data collection sources. Based on the results, a comparison of the information security of the content management systems was performed and a conclusion was reached.

The results showed that the Drupal content management system clearly has the best information security. It did best in almost all comparable features. The results also showed that the information security of a content management system is not just a matter of platform selection. The security of the website is the responsibility of its owner, and the content management system only provides the tools to build a secure website. The results of the thesis will certainly be useful for developers considering building a new website.

Keywords/tags (subjects)

Information security, content management system, website, vulnerabilities

Miscellaneous (Confidential information)

Sisältö

Kuviot

1 Johdanto

Internetin alkuaikoina verkkosivut olivat staattisia, yksinkertaisia HTML-tekstitiedostoja. Internetin suosion kasvaessa esiteltiin monia tapoja piristää nettisivuja, kuten kuvat, asioiden lataaminen sivuilta ja sivuille. Sen seurauksena verkkosivujen rakentamisesta ja ylläpitämisestä tuli entistä vaikeampaa ja tarvittiin uusi ja helpompi tapa hallita nettisivuston sisältöä. Vastauksena ongelmaan kehitettiin sisällönhallintajärjestelmä (Content management system (CMS)). Sen avulla voitiin automatisoida ja yksinkertaistaa verkkosivuston rakennusprosessia. Vuodesta 2003 lähtien helppokäyttöiset verkkosivujen rakennuksessa auttavat sisällönhallintajärjestelmät tarjosivat valmiita malleja ihmisille, joilla ei ollut aikaisempaa koodauskokemusta. Niitä olivat esimerkiksi vuonna 2003 julkaistut WordPress ja SquareSpace. (Heslop, B. 2018.)

Verkkosivujen määrän kasvaessa koko ajan kasvaa myös prosentti sivustoista, jotka joutuvat hyökkäyksen kohteiksi. Hyökkäykseen motivaatioita on monenlaisia. Hyökkääjät ovat esimerkiksi kehittäneet botteja, jotka käyvät järjestelmällisesti sivustoja läpi yrittäen käyttää hyväkseen tunnettuja haavoittuvuuksia. Ei ole myöskään pois suljettua, että kyseessä on suunniteltu ja kohdennettu hyökkäys organisaatiotasi vastaan. Paras tapa suojautua kyberhyökkäyksiltä on omaksua periaate, jonka mukaan sinua tai yritystäsi vastaan tulee kohdistumaan kyberhyökkäys jossain vaiheessa. Tämä ajattelutapa kannustaa sinua toimimaan vastuullisesti ja voi pelastaa sinut kyberhyökkäykseltä. (Buck, D n.d.)

Kaikkein haavoittuvimpia nettisivuista ovat ne, jotka käyttävät suosituimpia sisällönhallintajärjestelmiä. Suurin osa niiden haavoittuvuuksista johtuu juuri niiden suosiosta. Suosituimmista sisällönhallintajärjestelmistä kaikkein haavoittuvaisimpia tekee niistä niiden tunnettavuus. Kaikki tuntevat nämä alustat; myös hakkerit. Suosituimpiin sisällönhallintajärjestelmiin on myös hyökkääjien näkökulmasta katsottuna kannattavinta kohdistaa hyökkäys, sillä hyökkäyksen mahdollisia kohteita on enemmän. (Reshetilo, K. n.d.)

Tässä opinnäytetyössä valitaan kolme eri sisällönhallintajärjestelmää ja vertaillaan niiden tietoturvallisuutta. Lisäksi työssä esittelen yleisempiä tietoturvaongelmia tämän kaltaisissa järjestelmissä ja annan muutamia yleisiä neuvoja miten omasta sisällönhallintajärjestelmää käyttämästä

nettisivusta saa tehtyä turvallisemman. Työn tarkoitus on siis yrittää auttaa lukijaa valitsemaan itselleen turvallisin sisällönhallintajärjestelmä ja auttaa turvaamaan se.

2 Tutkimusasetelma

2.1 Tutkimuskysymys

Tässä tutkimuksessa tutkittiin sisällönhallintajärjestelmien tietoturvaa. Tutkimuskysymyksenä mietitään, mikä on valituista sisällönhallintajärjestelmistä tietoturvallisin? Opinnäytetyötä rajattiin seuraavilla tutkimuskysymyksillä:

1. Mitkä sisällönhallintajärjestelmät työhön valitaan?
2. Mitä tapoja tietoturvallisuuden vertailuun voi käyttää?
3. Millä valituista sisällönhallintajärjestelmistä on paras tietoturvallisuus?

2.2 Tutkimusmenetelmä ja tavoitteet

Opinnäytetyön tutkimusmenetelmä on kvalitatiivinen vertaileva tutkimus. Lisäksi vertailussa käytetään hyväksi myös määrällistä eli kvantitatiivista tutkimusta. Kvalitatiivisessa eli laadullisessa tutkimuksessa pyritään ymmärtämään jotain ilmiötä tai kohdetta kokonaisvaltaiseksi. (Laadullinen tutkimus. 2021.) Kvantitatiivisessa eli määrällisessä tutkimuksessa pyritään ymmärtämään ilmiötä tai kohdetta numeroiden ja tilastojen kautta. (Määrällinen tutkimus. 2015)

Vertailevassa tutkimuksessa pyritään vertailemaan joitain ilmiöitä tai kohteita. Tutkimuksessa hahmotellaan valittujen kohteiden eroja ja yhtäläisyyksiä. Vertailuun kelpaavat vaikkapa erilaiset kohteet tai ilmiöt, jotka ovat todettu jonkin ominaisuuksiensa tai kokonaisuutensa takia vertailuun sopiviksi. Vertaileva tutkimus perustuu yleensä tilastojen, kvalitatiivisten ja kvantitatiivisten aineistojen analyysiin. (Vertaileva tutkimus. 2015.)

Tässä tutkimuksessa pyritään ensin ymmärtämään sisällönhallintajärjestelmien tietoturvallisuutta kvalitatiivisen ja kvantitatiivisen tutkimuksen avulla, jonka jälkeen sisällönhallintajärjestelmien tietoturvallisuutta vertaillaan käyttäen vertailevaa tutkimusmenetelmää.

Työn lähteinä toimivat erilaiset internetartikkelit ja -julkaisut. Työn tavoitteena on auttaa nettisivujen tekijöitä valitsemaan tietoturvallisin vaihtoehto valituista sisällönhallintajärjestelmistä. Opinnäytetyön aihe on alun perin Jyväskylän ammattikorkeakoulun listalta otettu mutta sitä on hieman sovellettu itselleni sopivaksi.

3 Tietoperusta

3.1 Tietoturvallisuus

Tietoturvallisuus tarkoittaa pohjimmiltaan käytäntöä estää luvaton pääsy salattuihin tietoihin. Tietoturvallisuuteen sisältyy myös tarve estää tietojen luvaton käyttö, paljastaminen, häiriintyminen, muuttaminen, tarkastaminen, tallentaminen tai tuhoaminen. Tieto voi olla kahdessa eri muodossa, fyysisessä tai elektronisessa.

Tietoturvaohjelmat rakentuvat kolmen periaatteen ympärille, jotka tunnetaan yleisesti nimellä CIA (Confidentiality, Integrity, Availability) – luottamuksellisuus, eheys, saatavuus.

Luottamuksellisuus – Varmistus että tietoja ei luovuteta luvattomille henkilöille, yhteisöille tai prosessille.
Eheys – tarkoittaa tietojen tarkkuuden ja täydellisyyden säilyttämistä. Tietoja ei siis saa muokata luvatta.
Saatavuus – tarkoittaa, että tiedon on oltava saatavilla tarvittaessa.

(What is Information Security. 2017.)

3.2 Sisällönhallintajärjestelmä

Sisällönhallintajärjestelmä on ohjelmisto, joka auttaa käyttäjiä luomaan, hallitsemaan ja muokkaamaan verkkosivuston sisältöä. Siitä käytetään usein lyhennettä CMS (Content management

system). Sisällönhallintajärjestelmistä on erityisesti apua niille, jotka eivät ole teknisesti taitavia mutta haluavat tehdä omat nettisivunsa. Yksinkertaisemmin sanottuna sisällönhallintajärjestelmä on työkalu, jonka avulla voit rakentaa verkkosivuston ilman että sinun ei tarvitse osata koodata.

Sisällönhallintajärjestelmän avulla voit keskittyä olennaiseen, eli oman sisällön tuottamiseen verkkosivullesi, sen sijaan, että rakentaisit verkkosivun kokonaan alusta. Sisällönhallintajärjestelmä hoitaa perusinfrastruktuurin asiat puolestasi, jotta voit rauhassa keskittyä verkkosivustosi kehittämiseen. (What Is a Content Management System (CMS)?. 2019.)

3.3 Avoin lähdekoodi

Avoin lähdekoodi on tapa jakaa ja kehittää ohjelmistoja. Siinä asiakas saa vapaasti käyttää, muuntaa, jakaa ja kopioida avoimen lähdekoodin ohjelmaa. Open Source Iniative:n avoimen lähdekoodin määritelmän mukaan tulee täyttää seuraavat vaatimukset:

1. Ohjelman täytyy olla vapaasti levitettävissä ja välitettävissä.
2. Lähdekoodin täytyy tulla ohjelman mukana tai olla vapaasti saatavissa.
3. Myös johdettujen teosten luominen ja levitys pitää sallia.
4. Lisenssi voi rajoittaa muokatun lähdekoodin levittämistä vain siinä tapauksessa, että lisenssi sallii korjaustiedostojen ja niiden lähdekoodin levittämisen.
5. Yksilöitä tai ihmisryhmiä ei saa asettaa eriarvoiseen asemaan.
6. Käyttötarkoituksia ei saa rajoittaa.
7. Kaikilla ohjelman käsiinsä saaneilla on samat oikeudet.
8. Lisenssi ei saa olla riippuvainen laajemmasta ohjelmistokokonaisuudesta, jonka osana ohjelmaa levitetään, vaan ohjelmaan liittyvät oikeudet säilyvät, vaikka se irrotettaisiin kokonaisuudesta.
9. Lisenssi ei voi asettaa ehtoja muille ohjelmille. Ohjelmaa saa levittää myös yhdessä sellaisten ohjelmien kanssa, joiden lähdekoodi ei ole avointa.
10. Lisenssin sisällön pitää olla riippumaton teknisestä toteutuksesta.

(Avoin lähdekoodi, n.d.)

3.4 Omisteinen ohjelmisto

Omisteinen ohjelmisto on avoimen lähdekoodin ohjelmiston vastakohta. Se tarkoittaa käytännössä mitä tahansa ohjelmistoa, joka on tekijänoikeudella suojattu. Omisteisen ohjelmiston kehittäjä, myyjä tai julkaisija on asettanut rajoituksia ohjelmiston käytölle. Tuotetta saa jakaa, muokata tai käyttää vain tuotteen omistajan myöntämillä oikeuksilla. Omisteisia ohjelmistoja kutsutaan myös suljetun lähdekoodin ohjelmistoiksi, tai kaupallisiksi ohjelmistoiksi. (Proprietary Software. 2017.)

3.5 Haavoittuvuus

Haavoittuvuus tarkoittaa aukkoa tai heikkoutta sovelluksessa, jota hyökkääjät voivat käyttää hyväkseen aiheuttaakseen vahinkoja sovelluksen omistajille. Owasp pitää top 10 listaa kaikkein kriittisimmistä haavoittuvuuksista tällä hetkellä. Jokaisen nettisivun omistajan kannattaa käydä lista läpi ja varmistaa että oma nettisivu on turvassa ainakin listalla olevilta haavoittuvuuksilta. (Vulnerabilities. n.d.)

Tämänhetkinen (24.4.2022) top 10:

1. Broken access control
2. Cryptographic failures
3. Injection
4. Insecure design
5. Security misconfiguration
6. Vulnerable and outdated components
7. Identification and authentication failures
8. Software and data integrity failures
9. Security logging and monitoring failures
10. Server-side request forgery

(OWASP Top Ten. n.d.)

3.6 CVE

CVE (Common Vulnerabilities and Exposures) tarkoittaa tietokantaa, joka sisältää kaikki julkistetut haavoittuvuudet. CVE antaa luotettavan ja helpon tavan jakaa ja tarkastella haavoittuvuuksia asiantuntijoiden ja muiden asiasta kiinnostuneiden välillä. (What is a CVE?. n.d.)

3.7 CVSS score

CVSS (The Common Vulnerability Scoring System) kuvaa numeroilla (0-10) jonkin haavoittuvuuden vakavuuden. CVSS arvot voidaan luokitella seuraavasti:

- 0.0 - None
- 0.1-3.9 - Low
- 4.0-6.9 - Medium
- 7.0-8.9 - High
- 9.0-10.0 - Critical

(What are CVSS Scores. n.d.)

4 Keskeisimmät sisällönhallintajärjestelmät

4.1 Yleistä

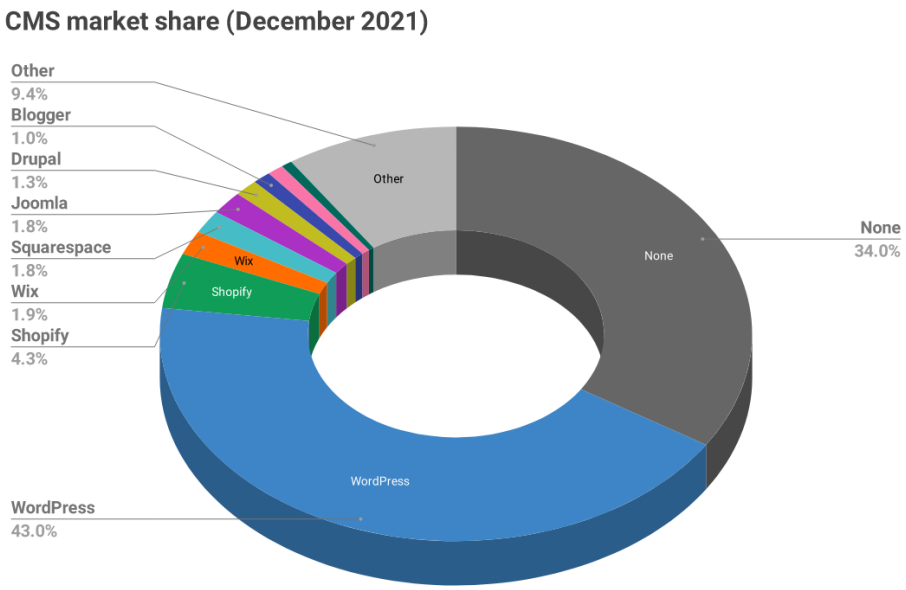
Tässä kappaleessa esitellään kriteerit, joiden perusteella sisällönhallintajärjestelmät valitaan ja esitellään valitut sisällönhallintajärjestelmät.

4.2 Kriteerit

4.2.1 Suosio

Kaikkia tietenkin kiinnostaa eniten käytetyimmät ja suosituimmat sisällönhallintajärjestelmät. Suosituimmissa järjestelmissä voi olettaa, että suurin osa haavoittuvuuksista on löydetty ja nähdään myös, miten haavoittuvuuksien löytämiseen reagoidaan jokaisen sovelluksen kohdalla.

Taulukosta (k.s kuvio 1). Näkee että suosituimpia sisällönhallintajärjestelmiä viime vuoden (2021) syyskuussa on ollut WordPress, Shopify ja Wix.



Kuvio 1. Suosituimmat sisällönhallintajärjestelmät vuoden 2021 joulukuussa. (CMS market share analysis. 2021.)

4.2.2 Avoin lähdekoodi vai kaupallinen

Yksi tärkeimmistä kysymyksistä, joita ihmiset kysyvät päättäessään sovelluksen turvallisuudesta, on, onko se avoimen lähdekoodin vai patentoitu. Avoimen lähdekoodin projektit ovat vahvasti riippuvaisia yhteisöstä ohjelmistojen kehittämisessä ja ylläpidossa, kun taas oikeita projekteja rakentaa ja ylläpitää yksi yritys, eivätkä ne yleensä salli pääsyä lähdekoodiin.

Jotkut päättelevät, että avoimen lähdekoodin projektien aiempien tietoturvaloukkausten vuoksi, jotka julkistettiin, avoin malli on luonnostaan turvaton. Tällaiseen johtopäätökseen päässeet uskovat, että koska lähdekoodi on julkisesti saatavilla, se on alttiina hakkereille, jotka tarkistavat koodia ja löytävät mahdollisia merkintöjä hyväksikäyttöön. Toisaalta avoimen mallin kannattajat uskovat,

että lähdekoodin läpinäkyvä luonne tarkoittaa, että projektin takana oleva yhteisö voi havaita vikoja ja tietoturva-aukkoja, jotka jäävät todennäköisesti huomaamatta pienemmälle ryhmälle.

Tosiasia on, että sekä patentoiduissa että avoimen lähdekoodin projekteissa on haavoittuvuuksia ja ne ovat alttiita tietoturvaloukkauksille. Olisi väärin väittää, että avoin lähdekoodi on enemmän tai vähemmän herkkä hyökkäyksille. Se, että omistusprojekteja toteuttavat palkatut ammattilaiset ja niitä tehdään voittoa tavoitellen, voi helposti tuudittaa ostajan väärään turvallisuuden tunteeseen. On tärkeää huomata, että tämä ei tee patentoiduista sisällönhallintajärjestelmistä täysin immuuneja tietoturvahyökkäyksille.

(Cybersecurity: What Risks if your Website is Based on a CMS?. 2020.)

Näin ollen vertailun kannalta olisi hyvä valita joko avoimen lähdekoodin tai kaupallinen sisällönhallintajärjestelmä. Koska avoimen lähdekoodin sisällönhallintajärjestelmillä on isompi markkinaosuus kuin kaupallisilla, ja avoimen lähdekoodin sisällönhallintajärjestelmistä löytyy paremmin tietoa, valitsen työhöni 3 suosituinta avoimen lähdekoodin sisällönhallintajärjestelmää.

4.3 Valinnan lopputulos

Näiden kriteerien perusteella suosituimmista sisällönhallintajärjestelmistä pois voi sulkea Shopify:n ja Wix:n, sillä ne eivät ole avoimen lähdekoodin sisällönhallintajärjestelmiä. Parhaiten vertailuuni soveltuu siis suosituimmista sisällönhallintajärjestelmistä avoimen lähdekoodin sisällönhallintajärjestelmät WordPress, Drupal ja Joomla!.

4.4 Sisällönhallintajärjestelmien esittely

Tässä kappaleessa esitellään lyhyesti valitut sisällönhallintajärjestelmät.

WordPress on vuonna 2003 perustettu kaikille suunniteltu ohjelmisto, joka korostaa saavutettavuutta, suorituskykyä, turvallisuutta ja helppokäyttöisyyttä. Alun perin nimellä b2/cafelog ja blogialustana tunnettu WordPress kehittyi pikkuhiljaa sisällönhallintajärjestelmäksi, jona se nyt tunnetaan. WordPress on rakennettu PHP:lle ja MySQL:lle ja lisensoitu GPLv2:lla. (About WordPress

n.d.) Sisällönhallintajärjestelmien hallitsijana pidetty WordPress on nykyään paljon edellä muita järjestelmiä, ja noin 35,6 prosenttia kaikista internetin sivuista käyttää WordPressiä. Muutamia suosittuja WordPressin käyttäjiä: The Walt Disney Company, Guggenheim, TechCrunch, BBC America. WordPressin sisällönhallintajärjestelmien markkinaosuus on 42 prosenttia. (Schäferhoff 2021a.)

Joomla! on ilmainen avoimen lähdekoodin sisällönhallintajärjestelmä sisällön tuottamiseen netissä. Kansainvälinen kehittäjien vapaaehtoisyhteisö on rakentanut ja kehittänyt Joomlaa sen julkaisusta 2005 lähtien. Tämä on tehnyt Joomlasta hyvin helppokäyttöisen, stabiilin ja turvallisen sisällönhallintajärjestelmän. (The Joomla Brochure n.d) Joomlan markkinaosuus sisällönhallintajärjestelmistä on 2,6 prosenttia. Muutamia suosittuja verkkosivustoja, jotka käyttävät Joomlaa: SendPulse, iTWire, UK National Crime Agency. (Schäferhoff 2021a.)

Drupal on vuonna 2001 julkaistu avoimen lähdekoodin sisällönhallintaohjelmisto, ja sillä on yksi suurimmista yhteisöistä maailmassa. Sitä käytetään monien päivittäisten verkkosivustojen ja sovellusten tekemiseen. Drupalin parissa työskentelee 1 000 000 intohimoista kehittäjää, suunnittelijaa, kouluttajaa, koordinaattoria, toimittajaa ja sponsoria. Drupalissa on loistavia vakio-ominaisuuksia, kuten helppo sisällöntuotto, luotettava suorituskyky ja erinomainen suojaus. (About Drupal n.d) Drupalin osuus sisällönhallintajärjestelmistä on 2 prosenttia. Muutamia suosittuja verkkosivuja, jotka käyttävät Drupalia: Council on Foreign Relations, Harvard University. (Schäferhoff 2021a.)

5 Yleisimmät tietoturvaongelmat sisällönhallintajärjestelmissä

5.1 Yleistä

Tässä kappaleessa esitellään yleisimmät haavoittuvuudet, joita sisällönhallintajärjestelmistä voi löytyä. Lisäksi kerron miten haavoittuvuuksia vastaan pystyy suojautumaan.

5.2 SQL-injektio

SQL-injektio on haavoittuvuus, jota esiintyy tietokantaa käyttävissä sovelluksissa. Siinä hakkeri onnistuu antamaan tietokantapalvelimelle komentoja, joita hänen ei pitäisi pystyä antamaan. Hyökkäys johtuu yleensä puutteellisen tai väärällä tavalla toteutetun syötteen tarkastuksen takia. On

tärkeää, että jokainen kohta missä lähetetään syötettä tietokantapalvelimelle, on suojattu SQL-injektion varalta. SQL-serveri suorittaa kaikki toiminnalliset käskyt mitä sinne syötetään. (SQL injection. n.d.)

Ainoa varma tapa estää SQL-injektio-hyökkäykset, on syötteiden validointi ja parametrisoidut kyselyt. Sovelluksen koodin ei pitäisi koskaan käyttää syötettä suoraan. Kehittäjän tulee varmistaa, että mikään syöte, joka sivustolta lähetetään ei pääse suodattamattomana SQL-serverille. SQL- injektion voi löytää hyvin käyttämällä sivullasi säännöllisesti haavoittuvuusskannausta. (SQL injection. n.d.)

5.3 Sivustojen välinen komentosarja

XSS eli sivustojen välinen komentosarja (Cross-site scripting) on tietoturva-aukko, jonka avulla voi suorittaa haitallista koodia turvallisella sivulla. Kun hyökkääjä löytää sivustojen väliselle komentosarjalle haavoittuvaisen sivuston, hän voi lisätä omaa koodiaan sivustolle, jolloin käyttäjän selain ajaa komennon aina kun joku käyttäjä käy sivustolla. Käyttäjän selain suorittaa komennon, koska luulee komennon olevan peräisin luotettavalta sivustolta. Tällä tavoin voi hyökkääjä saada haltuunsa uhrin tietoja kuten evästeen, kirjautumistunnukset tai istunnon tunnuksen. (Cross Site Scripting. n.d.)

Yleensä XSS-haavoittuvuuksien tehokas estäminen edellyttää seuraavien toimenpiteiden yhdistelmää:

- Suodata syöte saapuessaan. Kohdassa, jossa vastaanotetaan käyttäjän syöte, suodata mahdollisimman tarkasti odotetun tai kelvollisen syötteen perusteella.
- Tietojen koodaus. Kohdassa, jossa käyttäjän ohjattavissa oleva data tulostetaan HTTP-vastauksina, koodaa tuloste, ettei sitä tulkita aktiiviseksi sisällöksi.
- Sisällön suojauskäytäntö. Viimeisenä puolustuslinjana voit käyttää sisällön suojauskäytäntöä (CSP) vähentääksesi edelleen XSS-haavoittuvuuksien vakavuutta.

(Cross-site-scripting. n.d.)

5.4 Tiedostojen sisällyttäminen

Tiedostojen sisällyttämishaavoittuvuutta on kahta eri tyyppiä: etänä ja paikallisesti. Haavoittuvuudessa hakkeri pystyy lisäämään sisältöä tiedostoihin tai lataamaan haitallisia tiedostoja sivustolle. LFI:n(local file inclusion) avulla hyökkääjä voi lukea tai jopa suorittaa omia tiedostojaan uhrin koneella. RFI:n(remote file inclusion) avulla hyökkääjä voi suorittaa koodia, joka isännöidään hänen omalla koneellaan. (File inclusion vulnerabilities. n.d)

Tässä on muutamia tapoja estää LFI-hyökkäykset:

- Tunnusten määrittäminen. Tallenna tiedostopolut suojattuun tietokantaan ja anna jokaiselle tunnus, jolloin käyttäjät näkevät vain tunnuksensa ilman että he pystyisivät tarkastelemaan tai muuttamaan tunnuksen polkua
- Sallittujen luetteloiden lisääminen. Käytä vahvistettuja ja suojattuja luettelotiedostoja
- Käytä tietokantoja. Älä säilytä arkaluontoisia tiedostoja verkkopalvelimella vaan käytä siihen tarkoitukseen tietokantaa

(Kovacic,D. 2021.)

5.5 Hakemiston vaihtelu

Tässä haavoittuvuudessa hakkeri pyrkii saamaan pääsyn rajoitettuihin tiedostoihin ja suorittamaan komentoja web serverin root -hakemiston ulkopuolella. Käytännössä tämä tapahtuu siten, että hyökkääjä pyrkii muokkaamaan jotain pyyntöä siten, että komento hakisi tiedoston jostain muusta hakemistosta kuin normaalisti. (Path Traversal. n.d.)

Tehokkain tapa suojautua hakemiston vaihtelu -haavoittuvuudelta, on kokonaan välttää käyttäjän toimittaman syötteen välittämistä tiedostojärjestelmän sovellusliittymille. Monet sovellustoiminnot, jotka tekevät tämän, voidaan kirjoittaa uudelleen toimittamaan saman pyynnön turvallisemmalla tavalla. (Directory traversal. n.d.)

5.6 CSFR

CSRF (Cross Site Request Forgery) on hyökkäys, joka huijaa uhrin lähettämään haitallisen pyynnön. Se perii uhrin henkilöllisyyden ja oikeudet suorittaa ei-toivottu toiminto uhrin puolesta. Useimmilla sivustoilla selainpyynnöt sisältävät automaattisesti kaikki sivustoon liittyvät tunnistetiedot, kuten käyttäjän istuntoevästeen, IP-osoitteen, Windows-verkkotunnuksen tunnistetiedot ja niin edelleen. Siksi, jos käyttäjä on tällä hetkellä todennettu sivustolle, sivusto ei pysty erottamaan uhrin lähettämää väärennettyä pyyntöä uhrin lähettämästä laillisesta pyynnöstä.

CSRF-hyökkäys kohdistuu toimintoihin, joita nettisivulla voi tehdä uhrin puolesta kuten uhrin sähköpostiosoitteen tai salasanan vaihtamiseen. Lisäksi on voi olla mahdollista myös ostaa uhrin puolesta asioita, jos haavoittunut sivusto on verkkokauppa. Uhrin tietojen kalastelu ei hyödytä hyökkääjää, koska hyökkääjä ei saa vastausta, vaan uhri. (Cross Site Request Forgery. n.d.)

CSFR-haavoittuvuutta vastaan on monia keinoja suojautua. Suojautuminen CSRF:ltä edellyttää kahta asiaa: sen varmistamista, että GET-pyynnöillä ei ole muita vaikutuksia, ja sen varmistamista, että muut kuin GET-pyynnöt voivat olla peräisin vain asiakaspuolen koodista. (Protecting your users against csfr. n.d.)

5.7 Tietovuoto

Tietovuoto tapahtuu, kun verkkosivusto paljastaa tahattomasti arkaluontoisia asioita käyttäjilleen. Vuodettu tieto voi olla monenlaista. Esimerkiksi:

- Käyttäjätunnukset ja salasanat
- Arkaluontoiset tiedot yrityksestä tai verkkosivusta
- Luottokorttitiedot

(Information disclosure vulnerabilities. n.d.)

Tältä haavoittuvuudelta suojautuminen voi olla vaikeaa, kun tiedot voivat vuotaa lukuisilla eri tavoilla. Suojautumisessa vaaditaan siis erityistä tarkkuutta, että tiedot eivät pääse vuotamaan mitään kautta. Tässä muutamia hyviä käytäntöjä, joilla voit pyrkiä estämään tietovuotoja:

- Käytä geneerisiä virhe viestejä. Älä paljasta hyökkääjille turhaan sovelluksen toimintatapoja
- Opasta kaikkia organisaation henkilöitä toimimaan vastuullistesti arkaluontoisen tiedon käsittelyssä
- Tarkista kaikki koodi virheiden varalta

(Information disclosure vulnerabilities. n.d.)

5.8 Palvelunestohyökkäys

Dos (The Denial of Service) eli palvelunestohyökkäys on tietoverkkohyökkäys, joka voi kohdistua nettisivuihin ja palveluihin. Sen tarkoitus on tehdä kohteestaan käyttökelvoton eli “kaataa” se. Hyökkäyksen tarkoituksena on kohdistaa sivustolle enemmän liikennettä kuin sivusto pystyy käsittelemään. Hyökkääjät käyttävät hyökkäyksissään usein saastuneita tietokoneita. Liikenne voi koostua viesteistä, yhteyspyynnöistä tai vääristetyistä paketeista. DOS-hyökkäystä, joka tulee monesta eri lähteestä, kutsutaan lyhenteellä DDOS (Distributed Denial of Service). (Denial of Service. n.d)

Tässä hyviä käytäntöjä palvelinestohyökkäykseltä suojautumiseksi

- Valvo ja analysoi verkkoliikennettä. Verkkoliikennettä voidaan valvoa palomuurin tai tunkeutumisen havainnointijärjestelmän avulla. Järjestelmänvalvojat voivat määrittää sääntöjä, jotka luovat hälytyksiä epätavallisesta liikenteestä, tunnistavat liikenteen lähteitä tai pudottavat verkkopaketteja, jotka täyttävät tietyt kriteerit.
- Luo DoS-hyökkäysvastaussuunnitelma. Tärkeintä on luoda ja myös harjoitella hätäpalautussuunnitelma DoS-hyökkäyksien varalle. Suunnitelma sisältää yleensä viestinnän, lieventämisen ja palautuksen.

(Velimirovic, A. 2021.)

5.9 Todennuksen ohitus

Todennuksen ohituksen avulla hyökkääjä voi suorittaa haitallisia toimia sivustolla ohittamalla sivuston todennuksen. Todennuksen ohitus johtuu yleisimmin heikosta todennusmekanismista. (Information disclosure vulnerabilities. n.d.)

Paras suojautumiskeino on varmistaa, että käytössä on turvallinen ja vahva todennuskäytäntö. Muita hyviä tapoja on huolehtia, että sivustolla ei ole käytössä oletussalasanoja. Lisäksi sivusto kannattaa päivittää säännöllisesti. (Information disclosure vulnerabilities. n.d.)

5.10 Oikeuksien eskalointi

Oikeuksien eskaloinnissa käyttäjä saa oikeuksia, joita hänellä ei kuuluisi olla. On olemassa kahdentyyppistä oikeuden eskalointia: horisontaalista ja vertikaalista eskalointia. Horisontaalisessa tavassa käyttäjä saa oikeudet samalla tasolla olevaan toiseen käyttäjään itsensä lisäksi. Vertikaalisessa tavassa käyttäjä käyttää jotain systeemistä löytyvää vikaa saadakseen oikeuksia, joita ei ole tarkoitettu hänen tasoiselleen käyttäjälle. Täten käyttäjä voi käyttää oikeuksia vahingoittamaan systeemiä tai vaikka informaation varastamiseen. Vertikaalinen on tapa, jota yleensä tarkoitetaan, kun puhutaan oikeuksien eskaloinnista. (Rountree, D. 2011.)

Kun halutaan suojautua oikeuksien eskaloinnilta, käytetään yleensä tapaa, jota kutsutaan vähimpien oikeuksien periaatteeksi. Tämän periaatteen mukaan käyttöoikeudet tulee rajoittaa niin, että käyttäjän tai prosessin käyttöoikeudet ovat suppeimmat mahdolliset niin että ne kuitenkin suoriutuvat niille määrätyistä tehtävistä. Näin voidaan varmistaa, että käyttäjillä ei ole turhia oikeuksia, joita voidaan mahdollisesti käyttää verkkohyökkäyksissä hyväksi. On monia haavoittuvuuksia, jotka voivat johtaa oikeuksien eskalointiin kuten heikot salasanat, sivustojen välinen komentosarja ja evästeiden sopimaton käyttö. (Rountree, D. 2011.)

5.11 HTTP-vastauksen jako

Tämä haavoittuvuus syntyy, kun käyttäjän toimittamat tiedot kopioidaan vastauksen otsikkoon vaarallisella tavalla. Jos hyökkääjä voi lisätä rivinvaihtomerkkejä ja tyhjiä rivejä otsikkoon, hän voi

päästä eroon otsikosta ja siirtyä suoraan viestin runkoon. Tällöin hän voi kirjoittaa mahdollisesti haitallista sisältöä sovelluksen vastaukseen. (HTTP response header injection. n.d.)

Suojautumiseksi sovellusten tulisi välttää käyttäjän ohjaaman tiedon kopioimista HTTP-vastausotsikoihin. Jos tämä ei ole mahdollista, tiedot on tarkistettava tarkasti, jotta estetään vastausotsikon injektiohyökkäykset. Useimmissa tilanteissa on tarkoituksenmukaista sallia vain lyhyiden aakkosnumeeristen merkkijonojen kopioiminen otsikoihin, ja kaikki muut syötteet tulee hylätä. (HTTP response header injection. n.d.)

5.12 Haitallisen koodin suoritus

Haitallisen koodin suoritus tarkoittaa, että hyökkääjä on jotenkin saanut oikeudet suorittaa haitallista koodia verkkosivustolla. (What is Remote Code Execution (RCE)?. n.d.)

Tässä muutamia tapoja suojella itseäsi haitalliselta koodilta:

- Vältä komentorivien käyttöä
- Huolehdi että oikeutesi ovat kunnossa
- Varmista että Escape-syötteet ovat oikein

(Protecting against command execution attacks. n.d.)

5.13 Muita verkkosivuihin kohdistuvia uhkia

5.13.1 Väsytyshyökkäys

Väsytyshyökkäys on hakkerointimenetelmä, joka pyrkii yrityksen ja erehdyksen kautta murtamaan käyttäjätunnuksia, salasanoja ja salausavaimia. Hyökkäyksessä voi käyttää erilaisia ohjelmia, joiden avulla hyökkääjä voi kokeilla tuhansia eri tunnuksia ja salasanoja, jotta voisi saada selville käyttäjien kirjautumistunnuksia. (What Is a Brute Force Attack?. n.d)

Paras tapa suojautua väsytyshyökkäyksiltä on käyttää vahvaa salasanapolitiikkaa. Oletussalasanoja on vältettävä ja myös kirjautumisyritysten rajoittaminen esimerkiksi 3 kertaan minuutissa suojaa hyvin väsytyshyökkäyksiltä. (What Is a Brute Force Attack?. n.d)

6 Keskeisimpien sisällönhallintajärjestelmien turvallisuuden vertaaminen

6.1 Yleistä

Kun mietitään, onko jokin sisällönhallintajärjestelmä turvallinen, kannattaa miettiä asiaa seuraavien kysymyksien kautta:

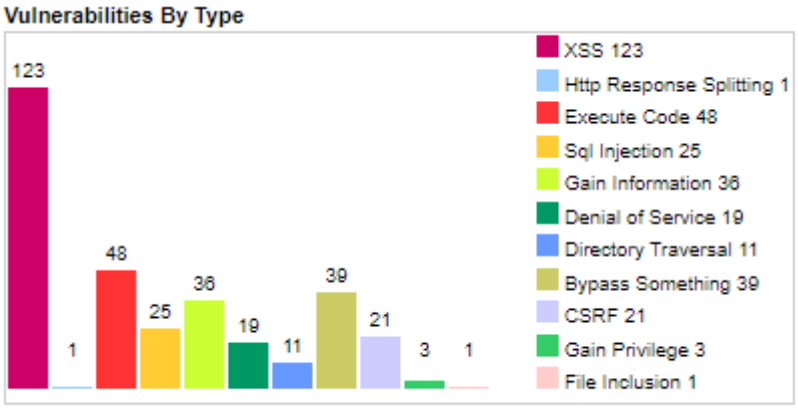
- Minkälaisia haavoittuvuuksia sisällönhallintajärjestelmillä on ollut ja kuinka paljon niitä on esiintynyt?
- Minkälainen tietoturvatiimi sisällönhallintajärjestelmillä on ja miten usein sisällönhallintajärjestelmät saavat päivityksiä?
- Mitä sisällönhallintajärjestelmiä saastuneet sivustot käyttävät?
- Jos käyttäjä käyttää kaikkia suositeltuja toimia sivustonsa turvaamiseksi, voiko sanoa, että sivusto on tietoturvallinen?

Tässä kappaleessa ei käsitellä vielä varsinaisia tuloksia, vaan tulokset esitetään erillisessä kappaleessa työn loppupuolella. Tässä kappaleessa siis vaan listataan tiedot minkä perusteella sisällönhallintajärjestelmien tietoturvallisuutta vertaillaan.

6.2 Hakkerointitavat

Tässä kappaleessa käsitellään millä tavoin sisällönhallintajärjestelmiä hakkeroidaan. Mitkä ovat yleisimmät tavat, joita käytetään, jotta palvelu pystytään hakkeroimaan. Mitkä ovat siis jokaisen sisällönhallintajärjestelmän haavoittuvuudet.

6.2.1 Wordpress



Kuvio 2. WordPressin haavoittuvuudet (WordPress Vulnerability Statistics. n.d.)

WordPressin yleisimpiä haavoittuvuuksia katsottaessa huomataan että, WordPressin yleisin haavoittuvuus on XSS eli sivustojen välinen komentosarja. Sitä on noin kolmas osa kaikista. Muita merkittäviä haavoittuvuuksia ovat haitallisen koodin suoritus ja tietovuoto. Yhteensä WordPressillä on ollut 343 haavoittuvuutta. (WordPress Vulnerability Statistics. n.d.)

Distribution of all vulnerabilities by CVSS Scores

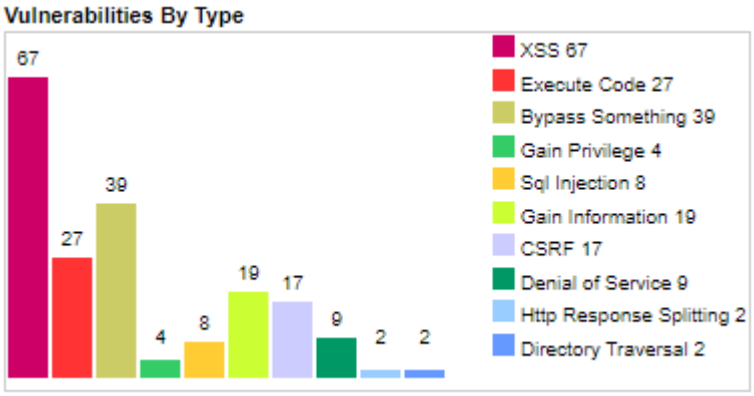
CVSS Score	Number Of Vulnerabilities	Percentage
0-1		0.00
1-2		0.00
2-3	8	2.30
3-4	29	8.40
4-5	122	35.50
5-6	83	24.10
6-7	48	14.00
7-8	42	12.20
8-9	1	0.30
9-10	11	3.20
Total	344	

Weighted Average CVSS Score: **5.9**

Kuvio 3. WordPressin haavoittuvuuksien CVSS Score:n keskiarvo (WordPress Vulnerability Statistics. n.d.)

WordPressin CVSS Score:n keskiarvo kaikista haavoittuvuuksista on 5.9.

6.2.2 Drupal



Kuvio 4. Drupalin haavoittuvuudet (Drupal Vulnerability Statistics. n.d.)

Drupalin tilastoista huomaa heti että niitä huomattavasti vähemmän kuin WordPressillä ja Joomlalla. Yleisin haavoittuvuus myös Drupalilla on ollut XSS. Seuraavaksi yleisimpiä ovat todennuksen ohitus ja haitallisen koodin suoritus. Yhteensä Drupalilla on ollut 228 haavoittuvuutta. (Drupal Vulnerability Statistics. n.d.)

Distribution of all vulnerabilities by CVSS Scores

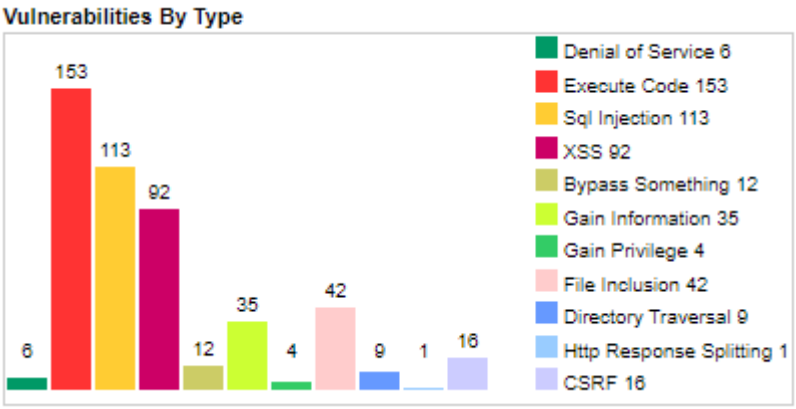
CVSS Score	Number Of Vulnerabilities	Percentage
0-1		0.00
1-2		0.00
2-3	10	4.40
3-4	21	9.20
4-5	79	34.60
5-6	66	28.90
6-7	27	11.80
7-8	22	9.60
8-9	1	0.40
9-10	2	0.90
Total	228	

Weighted Average CVSS Score: **5.7**

Kuvio 5. Drupalin haavoittuvuuksien CVSS Score:n keskiarvo (Drupal Vulnerability Statistics. n.d.)

Drupalin haavoittuvuuksien CVSS Score:n keskiarvo on 5.7.

6.2.3 Joomla!



Kuvio 6. Joomlan haavoittuvuudet (Joomla Vulnerability Statistics. n.d.)

Joomlalla on ollut selvästi eniten haavoittuvuuksia kaikista sisällönhallintajärjestelmistä. Joomla on kärsinyt etenkin haitallisen koodin suorittamisesta, Sql-injektioista ja XSS:sta kuten myös Drupal ja WordPress. Muita huomioita ovat tiedostojen sisällyttäminen -haavoittuvuuksien paljous, joita

WordPressillä oli vain 1 ja Drupalilla ei yhtään. Yhteensä Joomlalla on ollut 439 haavoittuvuutta. (Joomla Vulnerability Statistics. n.d.)

Distribution of all vulnerabilities by CVSS Scores

CVSS Score	Number Of Vulnerabilities	Percentage
0-1		0.00
1-2		0.00
2-3	2	0.50
3-4	7	1.60
4-5	98	22.30
5-6	92	21.00
6-7	63	14.40
7-8	169	38.50
8-9		0.00
9-10	8	1.80
Total	439	

Weighted Average CVSS Score: **6.7**

Kuvio 7. Joomlalla haavoittuvuuksien CVSS Score:n keskiarvo (Joomla Vulnerability Statistics. n.d.)

Joomlan haavoittuvuuksien CVSS Score:n keskiarvo 6.7.

6.3 Tietoturvatiimi

Tässä kappaleessa käydään läpi jokaisen verrattavan sisällönhallintajärjestelmän tietoturvatiimi ja kuinka usein jokaiselle alustalle julkaistaan päivityksiä.

6.3.1 Wordpress

WordPress julkaisee pieniä päivityksiä 50–60 päivän välein. Nämä päivitykset kuitenkin korjaavat vain pieniä virheitä ja ongelmia. Isoja päivityksiä tulee muutama kerta vuodessa. (Schäferhoff, N. 2019b) WordPressin tiimissä on noin 50 tietoturvallisuuseksperttiä. (About/Security. WordPress. N.d.)

6.3.2 Drupal

Drupal julkaisee kerran kuussa korjauksia ja tietoturvallisuus päivityksen. Isoja päivityksiä tulee noin kaksi kertaa vuodessa. (Schäferhoff, N. 2019b) Drupalin tiimissä on 36 tietoturvallisuuseksperttiä. (Drupal security team. N.d.)

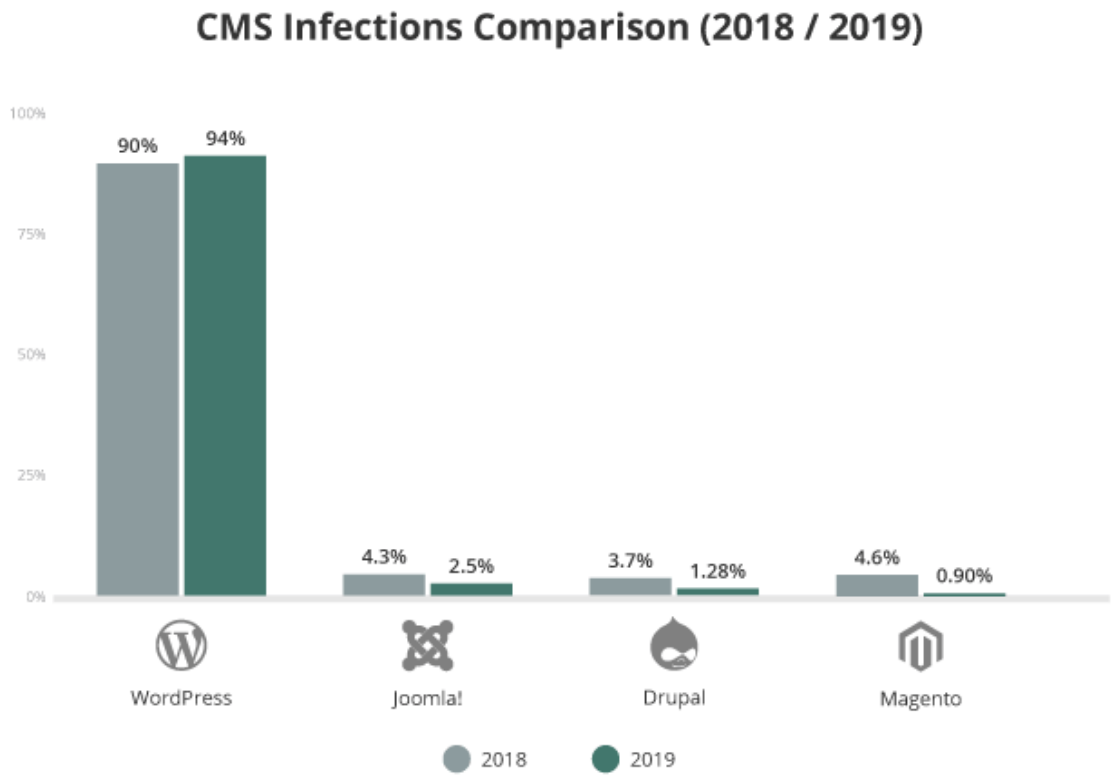
6.3.3 Joomla

Joomla julkaisee pienempiä päivityksiä 10–40 päivän välein, mutta isommissa päivityksissä voi mennä jopa vuosi. (Schäferhoff, N. 2019b) Joomlan tiimissä on 13 tietoturvallisuus eksperttiä. (Walker, A. 2022.)

6.4 Onko palvelu turvallinen?

Jokainen näistä palvelunhallintajärjestelmistä on turvallinen, kunhan ne on oikein asennettu.

6.5 Saastuneet sisällönhallintajärjestelmät



Kuvio 8. Saastuneiden sisällönhallintajärjestelmien prosenttiosuudet vuonna 2018 ja 2019 (2019 Website Threat Research Report. 2019.)

2019 vuonna 94 prosenttia kaikista tartunnan saaneista sisällönhallintajärjestelmällä toimivista sivustoista toimi WordPressillä. WordPressin karmeisiin lukuihin on monta selitystä. Ensinnäkin se on suosituin sisällönhallintajärjestelmä, jonka takia luku on luonnollisesti suurin. Suosituimman sisällönhallintajärjestelmän asema antaa hyökkääjille suuremman hyökkäyspinta-alan ja enemmän mahdollisia kohteita. WordPressiä vastaan kehitetyllä hyökkäyksillä voi hyökätä tuhansiin sivuihin, mutta Drupalia ja Joomlaa vastaan kehitetyillä hyökkäyksillä voi hyökätä huomattavasti vähempään määrään sivustoja, koska niitä ei yksinkertaisesti ole niin paljon. Tämä luonnollisesti motivoi hyökkääjät ottamaan kohteekseen mieluummin WordPress-järjestelmillä toimivia nettisivuja suuremman tuoton toivossa. Drupalilla ja Joomlalla on ollut WordPressiin verrattuna paljon vähemmän tartunnan saaneita sivustoja, mutta kun niitä verrataan keskenään, Drupal erottuu selkeästi edukseen.

Tämän uudempaa tilastoa ei ole julkaistu mutta voidaan olettaa, että WordPressin prosenttiosuudet ovat vielä kasvaneet vuoden 2019 luvuista.

7 Turvallisuuden parantamisen keinoja

7.1 Yleistä

Tässä kappaleessa esitellään vielä lyhyesti muutamia tärkeitä käytäntöjä, joilla sisällönhallintajärjestelmistä saa tehtyä turvallisemman.

7.2 Päivitykset

Tärkeä tapa pitää sisällönhallintajärjestelmä suojattuna, on muistaa päivittää sitä säännöllisesti. Ota uudet päivitykset käyttöön mahdollisimman pian. Päivitysten nopea käyttöönotto on erityisen tärkeää, jos niiden tarkoituksena on korjata jokin haavoittuvuus. Hyökkääjät seuraavat aktiivisesti haavoittuvuuksia. He yrittävät löytää verkkosivuja, jotka eivät ole vielä päivitettyjä ja ovat siksi haavoittuvuudelle alttiita. (Arnzt, P. 2018.)

7.3 Lisäosat

Riippumatta siitä, kuinka suojattu sisällönhallintajärjestelmä on, sinun on silti lisättävä sivustollesi lisäturvatoimia. Lisäosan/laajennuksen käyttäminen on helpoin tapa varmistaa, että palvelin on määritetty oikein. Se myös lisää verkkosivuston turvallisuutta. Tässä muutamia lisäosia esitylleille sisällönhallintajärjestelmille:

WordPress

- Sucuri
- WordFence

Joomla

- Akeeba Admin Tools

- RS FireWall

Drupal

- Security Kit

(Keeping your CMS safe and secure. 2022.)

7.3.1 Lisäosien ongelmat

Lisäosien mahdollisuuksien lisäksi niiden lisääminen verkkosivullesi sisältää myös riskejä. Lisäosat voivat sisällönhallintajärjestelmän tavoin sisältää haavoittuvuuksia, ja voivat tehdä verkkosivustosta entistä haavoittuvamman. Siksi on tärkeä miettiä mitkä lisäosat ovat erityisen tärkeitä sivustollesi, ja mitkä ovat mukana vain lisäämässä hyökkäyspinta-alaa. Tässä muutamia riskitekijöitä, jotka voivat vaarantaa lisäosan turvallisuuden:

- Lisäosa on huonosti koodattu.
- Kehittäjä ei ylläpidä lisäosaa.
- Lisäosa on huijaus.

(Keeping your CMS safe and secure. 2022.)

7.4 Varmuuskopiointi

On tapana luoda varmuuskopio ennen päivitysten tai muiden tärkeiden muutosten käyttöönottoa. On kuitenkin myös suositeltavaa luoda säännölliset varmuuskopiot. Varmuuskopiointi on hyvä tehdä viikoittain tai jopa päivittäin, sen mukaan kuinka usein julkaiset sisällönhallintajärjestelmään. Viikoittainen varmuuskopio riittää useille verkkosivustoille. (Arnzt, P. 2018.)

Varmuuskopiointi ei auta vain verkkohyökkäyksiä vastaan. Siitä voi olla myös hyötyä myös esimerkiksi tapauksessa, jossa pieni ohjelmointivirhe voi aiheuttaa tietojen menetyksiä. Varmuuskopiointi voi olla saatavana laajennuksena, lisäosana tai kolmannen osapuolen palveluna sisällönhallintajärjestelmääsi. (Das, A. 2021.)

7.5 SSL-sertifikaatti ja HTTPS

HTTPS (Hypertext Transfer Protocol Secure) on protokolla, jota käytetään suojaamaan Internetissä. HTTPS estää sieppauksia ja keskeytyksiä sisällön siirron aikana. Jotta voit luoda suojatun verkkoyhteyden, verkkosivustosi tarvitsee SSL-sertifikaatin. Jos verkkosivustosi pyytää käyttäjiltä mitä tahansa tietoja, sinun on salattava yhteytesi. Muuten kaikki tiedot ovat salaamattomia ja helposti hakkeroitavissa.

SSL (Secure Sockets Layer) on toinen välttämätön sivustoprotokolla. Tämä siirtää vierailijan henkilökohtaiset tiedot verkkosivustosi ja tietokantasi välillä. SSL salaa tiedot, jotta muut eivät voi lukea niitä siirron aikana.

(What is an SSL Certificate?. n.d.)

7.6 Haavoittuvuusskannaukset

Verkkosivustojen tietoturva-aukot eivät aina näy paljaalle silmälle. Se tekee haavoittuvuuksien skannauksesta olennaisen osan verkkosivuston turvallisuutta. Haittaohjelmien skannaukset ovat helppo ja yksinkertainen tapa parantaa verkkosivun turvallisuutta. Haavoittuvuusskannauksia kannattaa suorittaa säännöllisin väliajoin. Se voi säästää verkkosivusi isoilta ongelmilta ja tietovuodoilta. (Ghadawala, M. n.d)

8 Tulokset ja niiden analyysi

Tässä kappaleessa käydään läpi vertailun tuloksia.

Kaikkien sisällönhallintajärjestelmien CVSS Score:n keskiarvo sijoittuu medium –tasoiseen haavoittuvuuteen. Drupalin ja WordPressin luvun ero on melko pieni, vain 0,2. Merkittävää näissä luvuissa on kuitenkin Joomlan erottuminen näinkin paljon. Kun Joomlalla on ollut haavoittuvuuksia, ne ovat

olleet huomattavasti vakavampia kuin Drupalin ja WordPressin vastaavat vakavuudet. Eli keskimääräinen haavoittuvuus on Joomlaa käyttävällä verkkosivulla voinut aiheuttaa huomattavasti enemmän vahinkoa sivuston omistajalle kuin Drupalia tai WordPressiä käyttävälle sivustolle.

Drupal on selkeästi pystynyt parhaiten pitämään haavoittuvuudet minimissään, sillä on yli 100 haavoittuvuutta vähemmän kuin WordPressillä ja yli 200 vähemmän kuin Joomlalla. Joomlaa käyttävä verkkosivuston omistaja olisi siis joutunut kohtaamaan selvästi eniten haavoittuvuuksia kaikista vertailtavista sisällönhallintajärjestelmistä, melkein tuplasti enemmän kuin Drupal. WordPress pärjäsi tässä tilastossa keskinkertaisesti.

Kun katsotaan minkä tyyppisiä haavoittuvuuksia sisällönhallintajärjestelmillä on ollut, Drupal loistaa tässä tilastossa selkeästi. Drupalilla on ollut muutamaa poikkeusta lukuun ottamatta vähiten kaikkia haavoittuvuuksia. Ainoastaan todennuksen ohituksessa Joomlalla on selvästi ollut vähemmän haavoittuvuuksia kuin Drupalilla. Kaikissa muissa Drupalilla on ollut joko vähiten tai saman verran haavoittuvuuksia kuin muilla sisällönhallintajärjestelmillä. WordPressiä ja Joomlaa verrattaessa Joomlan WordPressillä on selvästi vähemmän ollut erityyppisiä haavoittuvuuksia. Joomlalla on kuitenkin ollut vähemmän XSS, todennuksen ohitus ja CSFR haavoittuvuuksia kuin WordPressillä.

Tietoturvatiimeissä WordPressillä on suurin tiimi, johon kuuluu 50 jäsentä. Toiseksi eniten on Drupalilla, jolla on 36 jäsentä. Joomlalla on vain 13 jäsentä tietoturvallisuus tiimissään. Tämän osion voittajana siis WordPress. Huomiona kuitenkin, että Drupalilla on kokoonsa nähden hyvin suuri määrä jäseniä tiimissään.

Päivityksiä kaikilla sisällönhallintajärjestelmillä tulee melko säännöllisesti mutta parhaiten tässä asiassa on kuntoutunut Drupal ja WordPress.

Saastuneimmista sivustoista 94 prosenttia vuonna 2019 oli WordPressiä käyttäviä verkkosivuja. Joomlaa oli vuonna 2019 2,5 prosenttia saastuneista sivuista ja Drupalia 1.28 prosenttia. Tämä tilasto on tietenkin jo muutaman vuoden vanha ja tilastossa korostuu suosituimman

sisällönhallintajärjestelmän merkitys. Tilastosta on kuitenkin huomattavissa, kuinka vähän Drupalia on Joomlaan verrattuna.

Tuloksista näkee selvästi, että Drupal on näistä valituista sisällönhallintajärjestelmistä eniten tietoturvallisuuteen panostanut sisällönhallintajärjestelmä. Drupalilla on vuosien saatossa ollut selkeästi vähiten haavoittuvuuksia verrattavista sisällönhallintajärjestelmistä, Drupalin haavoittuvuudet ovat keskimäärin olleet lievimpiä ja Drupalin tietoturvatiimi on selkeästi suurin verrattuna sisällönhallintajärjestelmien suosioon. Vastauksena siis tutkimuskysymyksiin, millä valituista sisällönhallintajärjestelmistä on paras tietoturvallisuus, on selvästi Drupal. Drupal on paras valinta, jos verkkosivun tekijä miettii mikä olisi avoimista sisällönhallintajärjestelmistä paras vaihtoehto tietoturvallisuutta ajatellen.

Vaikka Drupal onkin selvästi turvallisin vaihtoehto verratuista, on verkkosivun turvallisuus silti suurin osin käyttäjän vastuulla. Jokainen verratuista sisällönhallintajärjestelmistä noudattaa asianmukaisia turvallisuusstandardeja. Verkkosivun tekijä voi siis huoletta tehdä sivunsa myös Joomlalla tai WordPressillä, jos ne vaikuttavat muuten paremmilta vaihtoehdoilta. Loppujen lopuksi verkkosivuston turvallisuudessa on kyse sen tekijän turvallisuusvalinnoista. Sisällönhallintajärjestelmät tarjoavat vain työkaluja turvallisemman verkkosivuston rakentamiseen.

9 Pohdinta

Valitsin opinnäytetyön aiheen koulun antamasta listasta. Opinnäyteaiheeni alkuperäinen nimi oli ”Turvalliset verkkosivut – ohjeita yrittäjälle”. Tämä ei kuitenkaan aiheena sellaisenaan käynyt, joten mietimme opinnäytetyön ohjaajani kanssa jotain sovellusta aiheesta. Loppujen lopuksi päädyimme aiheeseen: ”Suosituimpien sisällönhallintajärjestelmien tietoturvallisuuden vertaaminen”. Aihe sopi minulle hyvin, sillä sisällönhallintajärjestelmät eivät olleet ennestään kovin tuttuja mutta aihe kiinnosti minua kovasti. Koulun kursseilla sisällönhallintajärjestelmistä WordPress oli jonkin verran tullut minulle tutuksi aikaisemmilla kursseilla mutta en ollut ennen käyttänyt Drupalia tai Joomlaa. Koin, että työn tekemisestä oli minulle itselleni paljon hyötyä ja opin paljon sisällönhallintajärjestelmistä ja niiden tietoturvakäytännöistä työtä tehdessä. Uskon että työstä voi olla paljon hyötyä myös uusille verkkosivuston rakentajille, jotka suunnittelevat ensimmäisen verkkosivunsa

rakentamista ja miettivät miten voida varmistaa, että heidän sivustonsa on mahdollisimman tieto-turvallinen.

Lähteet

2019 Website Threat Research Report. 2019. Viitattu 01.05.2022. https://sucuri.net/reports/2019-hacked-website-report/

About Drupal. N.d. Viitattu 01.05.2022. https://www.drupal.org/about

About WordPress. N.d. Viitattu 01.05.2022. https://wordpress.org/about/

About/security WordPress. N.d. Viitattu 01.05.2022. https://fi.wordpress.org/about/security/

Arnzt, P. 2018. How to secure your content management system. Viitattu 01.05.2022 https://blog.malwarebytes.com/security-world/technology/2018/08/secure-content-management-system/

Avoinlähde koodi. N.d. Viitattu 24.04.2022. https://coss.fi/avoimuus/avoin-lahdekoodi/

Buck, D. N.d. CMS Security: How to Keep Your Website Safe. Viitattu 12.04.2022. https://www.brandextract.com/Insights/Articles/CMS-Security-How-to-Keep-Your-Website-Safe/

CMS market share analysis. 2021. Viitattu 01.05.2022. https://joost.blog/cms-market-share/

Cross Site Request Forgery. N.d. Viitattu 01.05.2022. https://owasp.org/www-community/attacks/csrf

Cross Site Scripting. N.d. Viitattu 01.05.2022. https://owasp.org/www-community/attacks/xss/

Cross-Site Scripting. N.d. Viitattu 24.04.2022. https://portswigger.net/web-security/cross-site-scripting

Cybersecurity: What Risks if your Website is Based on a CMS?. 2020. Viitattu 05.05.2022.

https://www.vaadata.com/blog/cybersecurity-what-risks-for-websites-built-with-a-cms/#:~:text=However%2C%20CMS%20do%20present%20the,to%20come%20across%20vulnerable%20sites

Das, A. 2021. How to secure your content management system. Viitattu 03.05.2022 https://www.makeuseof.com/how-to-secure-your-content-management-system/

Denial of Service. N.d. Viitattu 24.04.2022 https://owasp.org/www-community/attacks/Denial_of_Service

Directory traversal. N.d. Viitattu 05.05.2022. https://portswigger.net/web-security/file-path-traversal

Drupal security team. N.d. Viitattu 01.05.2022. https://www.drupal.org/drupal-security-team

Drupal Vulnerability. Statistics. N.d. Viitattu 01.05.2022. https://www.cvedetails.com/product/2387/Drupal-Drupal.html?vendor_id=1367

File inclusion vulnerabilities. N.d. Viitattu 23.04.2022 https://www.offensive-security.com/metasploit-unleashed/file-inclusion-vulnerabilities/

Ghadawala, M. N.d. Top 8 Website Security Tips. Viitattu 04.05.2022. https://www.ccsinet.com/blog/top-8-website-security-tips/

Grinaker, S. 2018. What is the difference between a proprietary CMS and an open-source CMS?. Viitattu 04.05.2022. https://enonic.com/blog/difference-proprietary-open-source-cms#:~:text=A%20proprietary%20CMS%20keeps%20the,the%20licence%20and%20access%20rights).

Hendricks, D. N.d. 10 Essential Steps To Improve Your Website Security. Viitattu 24.05.2022. https://www.computer.org/publications/tech-news/trends/10-essential-steps-to-improve-your-website-security

Heslop, B. 2018. History of Content Management Systems and Rise of Headless CMS. Viitattu 01.05.2022 https://www.contentstack.com/blog/all-about-headless/content-management-systems-history-and-headless-cms/

HTTP response header injection. N.d. Viitattu 01.05.2022. https://portswigger.net/kb/issues/00200200_http-response-header-injection

Information disclosure vulnerabilities. N.d. Viitattu 01.05.2022. https://portswigger.net/web-security/information-disclosure

Johnson, M. 2017. Security tips for CMS. Viitattu 23.04.2022. https://www.cmscritic.com/security-tips-for-cms/

Joomla Vulnerability Statistics. N.d. Viitattu 01.05.2022. https://www.cvedetails.com/vendor/3496/Joomla.html

Keeping your CMS safe and secure. 2022. Viitattu 04.05.2022. https://blog.cpanel.com/keeping-your-cms-safe-and-secure/

Kovacic, D. 2021. Local File Inclusion (LFI): Understanding and Preventing LFI Attacks. Viitattu 13.03.2022. https://brightsec.com/blog/local-file-inclusion-lfi/

Laadullinen tutkimus. 2021. Viitattu 05.05.2022. https://koppa.jyu.fi/avoimet/hum/menetelmapolkuja/menetelmapolku/tutkimusstrategiat/laadullinen-tutkimus

Määrällinen tutkimus. 2015. Viitattu 15.05.2022. https://koppa.jyu.fi/avoimet/hum/menetelmapolkuja/menetelmapolku/tutkimusstrategiat/maarallinen-tutkimus

OWASP Top Ten. N.d. Viitattu 24.04.2022. https://owasp.org/www-project-top-ten/

Path Traversal. N.d. Viitattu 01.05.2022 https://owasp.org/www-community/attacks/Path_Traversal

Proprietary Software. 2017. Viitattu 05.05.2022. https://www.techopedia.com/definition/4333/proprietary-software

Protecting your users against csfr. N.d. Viitattu 01.5.2022. https://www.hacksplaining.com/prevention/csrf

Protecting against command execution attacks. N.d. Viitattu 03.05.2022. https://www.hacksplaining.com/prevention/command-execution

Reshetilo, K. N.d. These important security tips for your cms will make your website nearly bulletproof. Viitattu 26.04.2022. https://greenice.net/security-tips-cms-will-make-website-nearly-bulletproof/

Sheelu, G. 2019. Ways to secure CMS websites. Viitattu 04.05.2022. https://www.fortunesoftit.com/ways-to-secure-cms-websites/

Rountree, D. 2011. Privilage escalation. Viitattu 26.04.2022 https://www.sciencedirect.com/topics/computer-science/privilege-escalation

SQL Injection. N.d. Viitattu 24.02.2022. https://docs.microsoft.com/en-us/previous-versions/sql/sql-server-2008-r2/ms161953(v=sql.105)?redirectedfrom=MSDN

Schäferhoff, N. 2019a. Popular CMS by Market Share. Viitattu 26.04.2022 https://website-setup.org/popular-cms/

Schäferhoff, N. 2019b. WordPress vs. Joomla vs. Drupal. Viitattu 26.04.2022. https://website-setup.org/cms-comparison-wordpress-vs-joomla-drupal/

The Joomla Brochure. N.d. Viitattu 21.4.2022 https://www.joomla.org/images/press-logos/Brochure-Joomla-2019.pdf

Velimirovic, A. 2021. How to Prevent DDoS Attacks: 7 Tried-and-Tested Methods Viitattu 15.05.2022. https://phoenixnap.com/blog/prevent-ddos-attacks

Vertaileva tutkimus. 2015. Viitattu 04.05.2022. https://koppa.jyu.fi/avoimet/hum/menetelmapolkuja/menetelmapolku/tutkimusstrategiat/vertaileva-tutkimus

Vulnerabilities. N.d. Viitattu 21.4.2022 https://owasp.org/www-community/vulnerabilities/

Walker, A. 2022. Joomla vs WordPress vs Drupal | CMS Comparison in 2022. Viitattu 28.04.2022. https://www.guru99.com/joomla-vs-wordpress-vs-drupal.html

What is a CVE?. N.d. Viitattu 21.4.2022 https://www.balbix.com/insights/what-is-a-cve/

What is a content management system?. 2019. Viitattu 21.04.2022. https://kinsta.com/knowledgebase/content-management-system/

What is an SSL Certificate?. N.d. Viitattu 05.05.2020. https://www.websecurity.digicert.com/security-topics/what-is-ssl-tls-https

What are CVSS Scores. N.d. Viitattu 04.05.2022. https://www.balbix.com/insights/understanding-cvss-scores/

What is a Brute Force Attack?. N.d. Viitattu 05.05.2022. https://www.fortinet.com/resources/cyberglossary/brute-force-attack

What is Information Security. 2017. Viitattu 05.05.2022. https://www.geeksforgeeks.org/what-is-information-security/

What is Remote Code Execution (RCE)?. N.d. Viitattu 05.05.2022. https://www.checkpoint.com/cyber-hub/cyber-security/what-is-remote-code-execution-rce/

WordPress Vulnerability Statistics. N.d. Viitattu 01.05.2022. https://www.cvedetails.com/product/4096/Wordpress-Wordpress.html?vendor_id=2337