TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

VILLE VIRONMÄKI

TIETOTURVALLISEN ILMOITTAUTUMISSIVUN SUUNNITTELU JA TOTEUTUS - CASE POWERLAN

SISÄLLYSLUETTELO

1. JOHDANTO

Tämä dokumentti käsittelee lyhykäisesti TIE-30600 Turvallinen Ohjelmointi – kurssilla tehtyä harjoitustyötä. Aluksi käydään läpi käytetyt alustat sekä ohjelmisto-, että laitepuolelta, minkä jälkeen käydään läpi ohjelman tekoa ja tietoturvan ottamista huomioon sen suunnittelussa.

2. KÄYTETYT ALUSTAT JA OHJELMISTOT

Tässä kappaleessa esitellään käytetyt tekniikat serverillä pyörivistä ohjelmista frontendin toteuttamiseen käytettyihin kirjastoihin.

2.1 Serveriohjelmistot

Tehty ohjelmisto ”pyörii” DigitalOceanin [1] tarjoamalla virtuaaliprivateserverillä, johon on asennettu käyttöjärjestelmäksi Ubuntu 14.04. Backendiksi on valittu Django 1.8, joka tarvitsee serveriohjelmistoa varten adapteriksi Gunicornin. Gunicornin tarjoilemat sivut päätyvät Nginx-serveriohjelmistoon, joka tarjoaa sivut käyttäjille. Tietokantana käytetään PostgreSQL:lää, joka toimii suoraan Djangon kanssa. [2]

2.2 Backend

Ilmoittautumisohjelman backend on tehty käyttäen hyväksi Django – sovelluskehystä, jonka avulla voi ohjelmoida erilaisia sivustoja käyttäen Python – ohjelmointikieltä. Djangon ohjelmointifilosofia on lähellä MVC – arkkitehtuuria, siinä tosin ovat komponentteina model, template ja view.

Djangon avulla tietokantaohjelmointi on helppoa: kehys tarjoaa python-abstraktiotason tietokannan ja ohjelmoijan välille, jolloin SQL-kieleen ei tarvitse koskea muuten, kuin kannan pystytyksessä. Tietokannan taulut tehdään käyttäen Djangon modeleita, jotka ovat kuin Pythonissa käytetyt luokat.

Djangon avulla voidaan myös tehdä näkyvät nettisivut osittain templateiden avulla. Templatet ovat html-tiedostoja, joissa on erilaisia tageja, joihin dynaamisesti Django asettaa materiaalia.

2.3 Frontend

Pelkkiä Djangon templateita käyttämällä sivut olisivat suhteellisen ankeita. Tätä varten ilmoittautumissivuilla käytetään Bootstrap:piä [3] CSS-tyylittelyyn ja Crispy Formsia formien muotoiluun [4]. Myös JavaScriptiä löytyy varauksen vahvistamisen yhteydestä.

3. OHJELMISTON TOTEUTUS

3.1 Ohjelman osat

Ilmoittautumissivu koostuu etusivusta, rekisteriselostesivusta ja itse ilmoittautumissivusta. Ilmoittautumissivustolla näytetään X-kirjaimilla paikat, jotka on jo varattu CSS-muotoilua käyttäen. Varattujen paikkojen tiedot haetaan tietokannasta ja templatesta generoidaan lopullinen sivu käyttäen näitä tietoja hyväksi. Koodi tähän toimintoon löytyy views.py:stä.

Ilmoittautumisformi on kuvan alapuolella. Tässä annetut tiedot tarkistetaan forms.py:ssä ja asetetaan model-rajapinnan kautta tietokantaan, jos tiedot ovat oikeassa muodossa. Tietojen vastaanottamisen jälkeen käytetään vielä Djangon tarjoamaa rajapintaa sähköpostin lähettämiseen. Tässä on käytetty Gmailia, jonka palvelimeen Django ottaa yhteyttä.

3.2 Ilmoittautumissivun tietoturva

3.2.1 Toteutuneet tietoturvaominaisuudet

Sivusto ei tulosta suoraan mitään käyttäjien antamaa tietoa, joten XSS-haavoittuvuuksia ei pitäisi olla.

CSRF-haavoittuvuuksia ei periaatteessa voi tulla, koska käyttäjillä ei ole tunnuksia, mutta mahdollista laajentamista ajatellen ne estetään käyttämällä Djangoon sisäänrakennettua suojausmekanismia [5]. Tämä toimii siten, että Django antaa jokaisella käyttökerralla käyttäjälle uuden CSRF-keksin, joka tarkastetaan formin tietoja lähetettäessä.

SQL-injektioilta suojaudutaan myös automaattisesti käyttämällä Djangon tietokantarajapintaa, joka ei laita mitään putsaamattomana tietokantaan.

Tietokantaan menevän tiedon tarkistus on oikeastaan ainoa asia, joka tehdään itse. Tässä käytetään Pythonin perusrakenteita, kuten try-except.

Itse serverin tietoturva on otettu huomioon mm. siten, että root-sisäänkirjautuminen on estetty SSH:n kautta, ilmoittautumisohjelma ei pyöri root-käyttäjällä ja SSH-kirjautumisessa käytetään SSH-avaimia, eikä käyttäjätunnus-salasana -paria. Lisäksi serverillä pyörii UFW-niminen palomuuri, joka sallii sisääntulon ainoastaan porteista 25 ja 80. [6]

3.2.2 Tietoturvan laajentaminen

Eräs hyvä lisä sivulla olisi ollut CAPTCHA, esim. Googlen reCAPTCHAA käyttäen, mutta ajanpuutteen vuoksi tätä ei ehditty lisäämään systeemiin.

Myös tiedot olisi voitu salata, mutta tälle ei nähty suurta tarvetta, koska käyttäjistä ei tallenneta mitään ”raskauttavaa” tietoa.

SSL-sertifikaatit olisivat tuoneet sivulle lisää luotettavuutta, joten ne olisi ollut perusteltua tehdä. Tässä olisi voitu käyttää OpenSSL:lää ja Namecheapin [7] tarjoamaa SSL-sertifikointipalvelua.

Lisäturvaa serverille olisi saanut käyttämällä Fail2Ban-ohjelmaa, jolla voidaan estää tietyistä IP-osoitteista tulevat pyynnöt. Järjestelmään tunkeutumisen havannointiohjelmana (IDS) olisi voitu käyttää esim. TripWirea.

LÄHTEET

[1] DigitalOcean. Saatavissa: https://www.digitalocean.com/

[2] How to set up Django with Postgres, Nginx and Gunicorn on Ubuntu 14.04. Saatavissa: https://www.digitalocean.com/community/tutorials/how-to-set-up-django-with-postgres-nginx-and-gunicorn-on-ubuntu-14-04

[3] Bootstrap. Saatavissa: http://getbootstrap.com/

[4] CrispyForms. Saatavissa: http://django-crispy-forms.readthedocs.org/en/latest/

[5] Security in Django. Saatavissa: https://docs.djangoproject.com/en/1.8/topics/security/

[6] Initial Server Setup with Ubuntu 14.04. Saatavissa: https://www.digitalocean.com/community/tutorials/initial-server-setup-with-ubuntu-14-04

[7] Namecheap. Saatavissa: https://www.namecheap.com/