

Web Segurtasuna

Mikel Egaña Aranguren

mikel-egana-aranguren.github.io

mikel.egana@ehu.eus



Web Segurtasuna

<https://doi.org/10.5281/zenodo.4302267>

<https://github.com/mikel-egana-aranguren/EHU-SGSSI-01>



Ahultasun nagusiak web sistemetan

Aplikazioak seguruak izan behar dira

[Open Web Application Security Project-ek \(OWASP\)](#) ahulezia ohikoenak aztertzen ditu

Aldizkako txostena: OWASP Top (~~2017~~, [2021](#))

A1 - Injekzioa

Fidagarriak ez diren datuak interprete bati bidaltzen zaizkio kontsulta baten parte gisa

Interpretea engainatu daiteke eta baimendu gabeko datuetarako sarbidea lortu

Ezagunena SQL Injection

[A1:2017-Injection](#)

[Testing for SQL injection](#)

A1 - Injekzioa

Vector	Incidents	Description
1 and 1=2 union select password from qianbo_admin	634,566	Trying to query passwords
1'A=0	125,365	Probing
N;O=D and 1=1 nessus= or 1=1- CONCAT('whs(';)SQLi')	76,155	Probing by vulnerability scanners: Veracode, Nessus and WhiteHat Security, respectively
' union select unhex(hex(version())) —	848,096	Attempting to discover database version
;WAITFOR DELAY '00:00:28';	1,226,955	Blind probing — testing for delay in response

[SQL Injection Attacks: So Old, but Still So Relevant. Here's Why \(Charts\)](#)

A1 - Injekzioa

Nola ekidin?

- Sarrerak baliozkotzea karaktere "arriskutsuak" saihestuz
- Administratzaile-pribilegioak dituzten kontuak ez erabiltzea
- Behar baino informazio gehiago ez ematea (erabiltzaileari akatsen berri ez ematea)
- SQL sententziak ez eraikitzea zuzenean jasotako balioekin, sententzia parametrizatuak erabiltzea

A1 - Injekzioa

Nola ekidin? Datuak iragazi

- Formularioan username eremua badugu, eta badakigu erabiltzaileak letra eta zenbakiz bakarrik osatuta egon daitezkeela, ez da onartu behar "" edo "=" bezalako karaktererik
- E-mail eremua bada, expresio erregularrak erabil ditzakegu baliozkotzeko, hala nola `preg_match ('/ 2,3} .\+. @ +.!} $/, $_POST ['email'])`

A1 - Injekzioa

Nola ekidin? Datuak iragazi

- "Escape" funtzioak SQL sententzia batean erabiltzeko:
mysql_real_escape_string () barra alderantzikatuak jartzen ditu honako karaktere hauen aurretik:\x00,\n,\r,\,',"Eta\x1a
- Addslashes (), (PHP magic_quotes_gpc zuzentaraua lehenetsita aktibatuta dago, eta addslashes () funtzioa GET, POST eta COOKIE datu guztietan exekutatzen du)

A2 - Autentifikazioa galtzea

Erabiltzaile zilegi batek ezarritako saio batez baliatzea

Ikusgarriak diren saio-identifikatzaileak erabiltzen badira, a posteriori kopiatu eta erabil daitezke

Saioaren identifikatzailea cookie batean gordetzen bada, eskura daiteke

- XSS eraso baten bidez
- Sarean entzute baten bidez

A2 - Autentifikazioa galtzea

Saioa ixten ez bada, URL berean sar daiteke erabiltzaile legitimoak bere makina utzi duenean

A2 - Autentifikazioa galtzea

Arrazoi posibleak

- Pasahitz erraz eta ezagunak erabiltzea (1234, admin, etab.)
- Segurtasun-galdera errazak edo konfigurazio gutxikoak erabiltzea
- Pasahitzak ez dira enkriptatzen algoritmo seguru batekin
- Pasahitza erakusten da url-ean
- Saioen kudeaketa okerra

A2 - Autentifikazioa galtzea

Cookien arazoak

- Saioaren datuak modu iraunkorrean gordetzen badituzte, edonork eskura ditzake
- Script lengoaia erabiliz, fitxategian eta haren balioetan sar daiteke webgunean nabigatzen den bitartean
- Cookiearen datuak zifratu gabe bidaltzen badira, edonork entzun ditzake sarean

A2 - Autentifikazioa galtzea

Nola ekidin?

- Saioa itxi erabiltzailea deslogeatzen denean
- Saioa ixtea erabiltzaileak denbora pixka bat pasatzen duenean ezer egin gabe (timeout)
- Script lengoaien bidez sartzea saihesteko, **httponly** atributua erabili

A2 - Autentifikazioa galtzea

';--have i been pwned?

A3 - Datuen esposizioa

- Datuak zifratu gabe biltegitratzea
- Zifratu gabeko datuen transmisioa
- Zifratze-algoritmo ahulak erabiltzea

A3 - Datuen esposizioa

Nola ekidin?

- Informazio sentikorra zifratuta biltegitzen
- Informazio sentikorra protokolo seguruen bidez bidaltzen dela ziurtatuz
- Zifratze-algoritmo sendoak erabiliz

A4 - Kanpoko XML entitateak

Web zerbitzu askok dituzten XML prozesadoreetako ahultasunez baliatzea

- Kode txertatua duten XML fitxategiak igotzen
- XML fitxategien gelak erabiliz

A4 - Kanpoko XML entitateak

Nola ekidin?

- Ez erabili XML formatua (JSON erabili)
- XML fitxategi oro baliozkotzea, prozesatu aurretik
- XML prozesadoreak eguneratzea, ahultasun ezagunak saihesteko
- XML fitxategiak bidaltzeko jatorriak mugatzea

A5 - Sarbide-kontrola haustea

Autentifikaziorik gabe edo objektuekiko autentifikazio desegokiarekin sartzea

`www.sitio.com/consultar_datos.php?dni=45` (NAN = 45 duen erabiltzaileak bakarrik ikusi beharko lituzke bere datuak. Ez da nahikoa jakitea sartu nahi duen erabiltzailea logeatuta dagoen, NANa = 45 den jakin behar da)

A5 - Sarbide-kontrola haustea

Nola ekidin?

- Cookieak edo saioak erabiliz

A6 - Segurtasun-konfigurazio okerra

Aplikazioaren konfigurazio ona ez izatea

ezjakintasunagatik/despisteagatik/utzikeriagatik

- Lehenetsita sortzen diren kontuak/zerbitzuak mantentzen al dira?
- Gaituta al daude behar ez luketen funtzionalitateak?
- Administrazioaile-baimena duten erabiltzaileekin lan egiten da?
- Eguneratu gabeko softwarearekin lan egiten da?
- Zilegi al da pasahitz txarrak erabiltzea?

A6 - Segurtasun-konfigurazio okerra

Nola ekidin?

- Erabiltzen ez diren zerbitzu-kontuak ezabatzea
- Programa askok berez dakartzaten pasahitzak aldatzea
- Erabiltzen ez diren portuak/zerbitzuak desgaitzea
- Aplikazioak eguneratzea
- Erabiltzaileak pasahitz "seguruak" erabiltzera behartzea

A6 - Segurtasun-konfigurazio okerra

Nola ekidin?

- Aplikazio bakoitzerako erabiltzaile espezifikoak definitzea aplikazio horri dagozkion baimenekin (MySQL aplikazioan, konexioetarako root erabiltzailea ez erabiltzea. Aplikaziorako erabiltzaile bat edo batzuk sortzea, dagozkien taulen gainean dagozkien baimenekin)

A7 - Komandoen sekuentzia leku gurutzatuetan (XSS)

Exekutatu erabiltzaileak berak sar dezakeen kodea Script lengoaian

Metodorik errazena, formulario-eremuen bidez

Iraunkorrak izan daitezke scripta datu-base batean biltegitratzen bada:

- Biltegitratzen den eremu gisa sartzen da
- Eremu horren balioa erakusten den bakoitzean exekutatzeko da

A7 - Komandoen sekuentzia leku gurutzatuetan (XSS)

Nola ekidin?

- Erabiltzaileak sartzen duen testua garbituz ("Escape"), karaktere arriskutsuak kenduz
- URLtik lortzen ditugun balioak garbituz, karaktere arriskutsuak ezabatuz
- Funtzio batzuek "Escape" lana testu garbi bat itzuliz egiten dute

A8 - Deserializazio ez-segurua

Datuak paketatzeko prozesuaz aprobetxatzea datu faltsuak/maltzurak sartuz

Datuak prozesatzen dituzten formatuetan datozenean, horietatik abiatuta objektuak eraikitzeko, nahi ez diren objektuak sortzen dituzten datuak sar daitezke

A8 - Deserializazio ez-segurua

Nola ekidin?

- Datu primitiboetan soilik datuak onartzen dituzten formatuak erabiliz (string, int, etab.), adibidez JSON
- Sinatuta datozen datuak soilik paketatuz
- Jatorri fidagarrietatik datozen datuak bakarrik deserializatuz
- Deserializazio-prozesua aplikazioaren gainerakotik kanpo eta ingurune kontrolatu batean gauzatuz

A9 - Ahulezia ezagunak dituzten osagaiak erabiltzea

Nola ekidin?

- Adi egon kalteberatasunak argitaratzen direnean
- Atera bezain laster, dagozkien adabakiak aplikatuz
- Kalteberatasunik ez duten alternatibak bilatzen

A10 - Logeo eta monitorizazio ez nahikoak

Ez erregistratu sartzeko saiakerak, arrakasta izan edo ez

Sartzeko saiakeren informazioa oso baliotsua izan daiteke

- Jokabide susmagarriak detektatzeko (langileak sisteman sartzen, behar ez dutenean)
- Erasoak detektatzeko (erabiltzaile berarekin huts egindako hainbat saiakera)
- Kasu batzuetan, ezinbestekoa da informazio hori gordetzea, DBLO (LOPD) dela eta

A10 - Logeo eta monitorizazio ez nahikoak

Nola ekidin?

- Konexio arrakastatsuei eta huts egindakoei buruzko informazioa biltegitratuz
- Konexioei buruzko informazioa aztertzen
- Erabiltzailea blokeatu X saiakera huts jarraian baditu