Szegedi Tudományegyetem

Informatikai Intézet

**SQL injection detektálás természetes nyelvi feldolgozó rendszerek és gépi tanulás által**

**SQL injection detection using natural language processing systems and machine learning**

Szakdolgozat

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Készítette: |  | Témavezető: |  |
|  | Stiller Marianna |  | Dr. Vidács László |  |
|  | programtervező informatika szakos hallgató |  | egyetemi oktató |  |

Szeged

2021

## Feladatkiírás

Az SQL Injection napjaink leggyakoribb és legveszélyesebb sérülékenysége, amely a felhasználói input elégtelen mértékű validációjából ered, következménye pedig SQL lekérdezések futtatása érzékeny adatokon.

A rendszer természetes nyelvi feldolgozó rendszerek és gépi tanulás által képes lesz forrásódban SQL Injection sérülékenységeket keresni. Működésének alapja, hogy valós PHP alapú forráskódokból reprezentál absztrakt szintaxis fákat, melyekből mintákat nyer ki, amit később megtanul.

## Tartalmi összefoglaló

A téma megnevezése:

SQL injection detektálás természetes nyelvi feldolgozó rendszerek és gépi tanulás által.

A megadott feladat megfogalmazása:

Minták által tanított program képes reális becslést adni egy kód SQL injectionbeli sebezhetőségével kapcsolatban.

A megoldási mód:

PHP kódokból generálok absztrakt szintaxis fákat, amelyeket felcímkézek, mint sérülékeny és nem sérülékeny példák. Ezek átesnek egy vektorizáláson, majd egy kifejezett modell segítségével – amely a legkevesebb hibás döntéseket hozza - a program megtanulja felismerni a sérülékenység jeleit. Ennek hatására egy nem tanuló kódról képes kiértékelést készíteni.

Alkalmazott eszközök, módszerek:

A IntelliJ fejlesztői környezetet, emellett a gépi tanulás során Python, a grafikus asztali felület megvalósításához Java programozási nyelvet alkalmaztam.

Elért eredmények:

A program a bemenetként megadott PHP kódra kimenetként egy százalékos értéket ad, amely a kód sérülékenységének mértéke.

Kulcsszavak:

Felügyelt tanulás, train/test készlet, K-Fold, confusion mátrix, recall.

## Tartalomjegyzék

[Feladatkiírás 2](#_Toc70106693)

[Tartalmi összefoglaló 3](#_Toc70106694)

[Tartalomjegyzék 4](#_Toc70106695)

[BEVEZETÉS 5](#_Toc70106696)

[1. SQL INJECTION 6](#_Toc70106697)

[1.1. Meghatározása 6](#_Toc70106698)

[1.1.1. Példa 6](#_Toc70106699)

[1.2. Következményei 6](#_Toc70106700)

[1.3. Megelőzési módjai 6](#_Toc70106701)

[1.3.1. Példa 6](#_Toc70106702)

[2. GÉPI TANULÁS 7](#_Toc70106703)

[3. PROGRAM FELÉPÍTÉSE 7](#_Toc70106704)

[4. EREDMÉNY ÉS ANALÍZIS 7](#_Toc70106705)

[Irodalomjegyzék 8](#_Toc70106706)

[Nyilatkozat 9](#_Toc70106707)

[Köszönetnyilvánítás 10](#_Toc70106708)

## BEVEZETÉS

A mai világban, az informatika világában, a digitalizálás visszafordíthatatlan, illetve nem okszerű. Az informatika meglehetősen gyors fejlődése miatt, egy generáció már születésétől kezdve természetesnek vélheti az internethasználatot. A társadalom kiélvezi az összes kényelmi funkcióját, lebegnek a felszínen, a boldog tudatlanságban, az adataik pedig másodpercről másodpercre szivárognak le, a felszín alá. Az információk megtartása és biztonsága ezáltal nyert a tudományágában külön területet. Mivel értékesebbek lettek, mint a pénz maga.

Információbiztonságra szakosodó programtervező informatikusként állítom, hogy a szoftverfejlesztés nem minden. A bemeneti mezőket nem elég implementálni, meg is kell védeni őket. Ehhez pedig már több módszer van, annál is több példával. A programom mesterséges intelligencia segítségével ezeknek egy elenyésző részét használja fel a tanuláshoz és annak hasznosításához lehetőséget adva ezzel a fejlesztőknek arra, hogy biztonságos programkódot helyezzenek a „használók” elé.

A témaválasztásom egyszerű célja, hogy egy olyan kezdetleges rendszert készítsek, amely lehetőséget ad megvédi azt, ami a pénznél is drágább. Bár az injektálás csak egy része a biztonsági kockázatoknak, ám a legfenyegetőbb. A dolgozatban ki fogom fejteni többek között ezt a fajta sérülékenységet, a tanulási módszereket és a kettő ötvözetét egy programon keresztül, amely kimutatja a biztonságtechnikai fenyegetettséget.

1. SQL INJECTION

## Meghatározása

Az SQL Injection napjaink leggyakoribb és legveszélyesebb sérülékenysége, amely a felhasználói input elégtelen mértékű validációjából ered.

* + 1. Példa

## Következményei

Sikeres SQL Injection támadás képes:

* érzékeny adatok kiolvasására az adatbázisból,
* adatbázis béli adatok módosítására (beszúrás, frissítés, törlés),
* adminisztrációs műveletek végrehajtására az adatbázisban (például a DBMS leállítására),
* a DBMS fájlban található adott fájl tartalmának helyreállítására rendszerben, és
* egyes esetekben parancsokat ad ki az operációs rendszernek.

## Megelőzési módjai

* + 1. Példa

1. GÉPI TANULÁS
2. PROGRAM FELÉPÍTÉSE
3. EREDMÉNY ÉS ANALÍZIS

## Irodalomjegyzék

## Nyilatkozat

Alulírott Stiller Marianna programtervező informatikus BSc szakos hallgató, kijelentem, hogy a dolgozatomat a Szegedi Tudományegyetem, Informatikai Intézet Szoftverfejlesztés Tanszékén készítettem, programtervező informatikus BSc diploma megszerzése érdekében. Kijelentem, hogy a dolgozatot más szakon korábban nem védtem meg, saját munkám eredménye, és csak a hivatkozott forrásokat (szakirodalom, eszközök stb.) használtam fel. Tudomásul veszem, hogy szakdolgozatomat / diplomamunkámat a Szegedi Tudományegyetem Informatikai Intézet könyvtárában, a helyben olvasható könyvek között helyezik el.

2021.

## Köszönetnyilvánítás