Budapesti Műszaki és Gazdaságtudományi Egyetem
Méréstechnika és Információs Rendszerek Tanszék
Hibatűrő Rendszerek Kutatócsoport

Statikus és dinamikus analízis JavaScript-környezetben

Lucz Tamás Soma
Konzulens: Honfi Dávid doktorandusz
MIT, Informatikai technológiák szakirány, Rendszertervezés ágazat
Önálló laboratórium, 2015/2016, II. félév

Szoftvereink kódját emberek írják. Az emberek természetes tulajdonsága, hogy hibákat követnek el, amik a megfelelő eszköztárak hiányában felfedezetlenek maradhatnak. Ezen fejlesztői hibák fokozott kockázatot jelenthetnek a készülő szoftverre, hiszen a logikailag esetlegesen helytelen működés mellett jelentős biztonsági réseket eredményezhetnek; kiaknázásuk a szoftver nemkívánatos viselkedését idézheti elő. Ez rosszindulatú támadóknak lehetőséget nyújt arra, hogy a szoftvert számukra kedvező, a fejlesztők számára kedvezőtlen módon, de mindenképpen a szándékolttól eltérő módon futtassák.

Feladatom volt a félév során, hogy a fenti szempontokat figyelembe véve egy olyan komplex analízis-eszköztár kifejlesztésének elméleti és gyakorlati lehetőségeit vizsgáljam, amely vállalati JavaScript-kódtárak elemzésével fejlesztői hibák jelenlétére hívja fel a figyelmet, csökkenteni igyekezvén ezzel a készülő szoftverbe kerülő biztonsági kockázatokat előidéző hibák számát.

Ennek első lépéseként megismerkedtem a forráskódanalízis általános fogalmaival, valamint a JavaScript-forráskódok statikus és dinamikus elemzésének lehetőségeivel. A nyelv különféle változatainak, szabványainak mélyebb megismerése után konkrét, kurrens technológiai eszközöket kerestem, amelyek lehetővé teszik egy testreszabható, automatizált munkafolyamat létrehozását a fentebb közölt probléma megoldására.

Beszámolómban a forráskódanalízis általános módszereinek áttekintése után betekintést nyújtok a JavaScript programozási nyelv sajátosságaiba, majd egy rövid történeti kitekintés és a nyelv szabványosításának bemutatása után ismertetem a JavaScript-specifikus kódanalízis módszereinek egy részhalmazát.

Ezek után bemutatom a félév során általam megismert eszközök főbb működésmódjait, jellegzetességeit, a legtöbb eszköztár funkcionális lehetőségeit saját példán illusztrálva.

Összefoglalásként felvázolok egy, a korábban bemutatott eszközökre támaszkodó, azokat összekapcsoltan, moduláris és bővíthető munkafolyamatban használó hibrid analízist, melynek eredménye egy JavaScript-szoftver komplex, testreszabható analitikai áttekintése. A munkafolyamatra építendő IDE-plugin konkrét fejlesztői hibákat lesz képes feltárni, jelentős mértékben lecsökkentve ezzel az éles környezetbe kikerülő szoftver használatának biztonsági kockázatát.