HRVATSKA AKADEMSKA I ISTRAŽIVAČKA MREŽA
CROATIAN ACADEMIC AND RESEARCH NETWORK

Provjera XSS i *SQL Injection* ranjivosti *Exploit Me* skupom alata

CCERT-PUBDOC-2008-01-215

CARNet CERT u suradnji s LS&S

Sigurnosni problemi u računalnim programima i operativnim sustavima područje je na kojem CARNet CERT kontinuirano radi.

Rezultat toga rada ovaj je dokument, koji je nastao suradnjom CARNet CERT-a i LS&S-a, a za koji se nadamo se da će Vam koristiti u poboljšanju sigurnosti Vašeg sustava.

CARNet CERT, www.cert.hr - nacionalno središte za **sigurnost računalnih mreža** i sustava.

LS&S, www.lss.hr - laboratorij za sustave i signale pri Zavodu za elektroničke sustave i obradbu informacija Fakulteta elektrotehnike i računarstva Sveučilišta u Zagrebu.

Sadržaj

1. Uvod

Razvojem visokih tehnologija modernog doba, većina se organizacija u velikoj mjeri oslanja na računala i računalne mreže. Takvi sustavi olakšavaju i pospješuju rad tvrtkama, ali nesumnjivo unose i visok stupanj rizika od kompromitacije računala i podataka na njima. Računalna mreža i entiteti koji se njome koriste za međusobnu komunikaciju nužno uključuju određeni stupanj ranjivosti koje zlonamjerne osobe mogu iskoristiti.

Provjera ranjivosti je postupak otkrivanja poznatih ranjivosti i slabosti u mrežnim sredinama i računalima, a provodi se pomoću specijaliziranih alata koji, u procesu provjere ranjivosti, obavljaju niz testova na segmentu računalne mreže definiranom na početku testa.

Provjera ranjivosti je automatizirani postupak te je upotreba alata za provjeru sigurnosti jedini način da se u složenoj računalnoj mreži dobiju informacije o stupnju ranjivosti pojedinih računala ili poslužitelja.

Alati za provjeru ranjivosti mogu se podijeliti u dvije skupine, alati za lokalnu provjeru ranjivosti i alati za udaljenu provjeru ranjivosti. Alati za lokalnu provjeru ranjivosti pokreću se isključivo na sustavu koji se ispituje, dok alati za udaljenu provjeru ranjivosti ispitivanja obavljaju preko mreže pa ne trebaju nužno biti pokrenuti na sustavu koji se ispituje.

Automatizirani testni programi nisu u stanju do kraja razjasniti i utvrditi važnost pojedine ranjivosti te ju staviti u kontekst okoline u kojoj se ona javlja pa je potreban ljudski faktor za prepoznavanje i vrednovanje ranjivosti sustava.

Neki od sigurnosnih propusta koji se mogu pojaviti na računalnoj mreži su ranjivosti na XSS i *SQL injection* napade. Uvjeti i načini izvođenja su ukratko objašnjeni u slijedećim poglavljima. Postoji mnogo alata za provjeru XSS i *SQL injection* ranjivosti, a jedan od njih je *Exploit Me*. Radi se o skupu alata za udaljenu provjeru ranjivosti koji se sastoji od *XSS Me* te *SQL Inject Me* programa, a oni su opisani u poglavljima koja slijede. Također, u opis programa uključeni su i primjeri izvođenja te interpretacija rezultata.

2. XSS i *SQL injection* napadi

2.1. XSS napad

XSS (eng. *Cross-site scripting*) je napadačka tehnika koja web aplikaciju prisiljava da korisničkom pregledniku proslijedi preoblikovan odgovor, koji se zatim učitava i prikazuje. Preoblikovani odgovor obično uključuje dijelove programskog koda pisanog u programskom jeziku *JavaScript*, ali može biti kreiran i nekim drugim jezikom kojeg podržava korisnikov web preglednik. Neki od tih jezika su HTML, *VBScript*, *ActiveX*, *Java* i *Flash*.

XSS ranjivost se javlja uslijed neodgovarajuće ocjene ispravnosti ulaznih podataka web aplikacije. Svaka web stranica koja korisniku omogućuje unos nekih podataka potencijalno sadrži XSS ranjivost te se često javlja kod pretraživanja web stranica, kod podnošenja podataka na obrascima web stranica, na forumima te internetskim dnevnicima (eng. *blog*).

Kada napadač uspije potaknuti korisnički web preglednik na izvođenje podmetnutog programskog koda, on će se izvoditi unutar tzv. sigurnosne zone web aplikacije. Koristeći ovaj pristup, napadač može čitati ili promijeniti osjetljive podatke dostupne web pregledniku, tj. može ukrasti korisničke račune (krađa *cookie* datoteka), sjednice, usmjeriti web preglednik na neke druge lokacije, ili proslijediti štetan sadržaj iz druge web aplikacije. Osim toga, XSS napadi ugrožavaju povjerljivi odnos između korisnika i web aplikacije.

Postoje tri vrste XSS napada:

- neustrajni (eng. *non-persistent*),
- temeljeni na DOM (eng. *Document Object Model*) modelu i
- ustrajni (eng. *persistent*).

Neustrajni napadi navode korisnika na posjećivanje posebno oblikovanih internetskih poveznica (eng. *link*) za koje je vezan štetni programski kod. Ukoliko korisnik slijedi poveznicu i posjeti stranicu, programski kod koji je pohranjen unutar URL (eng. *Uniform Resource Locator*) polja izvršit će se u korisnikovom web pregledniku. Oprezniji korisnici mogu prepoznati opasnost ako u sadržaju poveznice primijete skriptu, odnosno programski kod kojem tu nije mjesto. Zbog toga napadači za prikrivanje tragova i zavaravanje korisnika programski kod najčešće izobliče koristeći heksadekadsko kodiranje.

Napadi temeljeni na DOM modelu (platforma i standard za prikaz HTML ili XML sadržaja te ostalih sličnih sadržaja) slični su neustrajnim napadima. Ovakve napade napadač može izvesti kada propust web preglednika omogućuje tretiranje prikazane stranice kao datoteke smještene na lokalnom računalu korisnika. Ukoliko napadač postavi zlonamjerno oblikovanu web stranicu koja sadrži poveznicu na neku lokaciju na korisnikovom računalu i korisnik ju posjeti, napadač može pokrenuti štetan programski kod na osjetljivom računalu.

Za razliku od neustrajnih, ustrajni se napadi događaju kada je određeni štetni programski kod pohranjen unutar same web aplikacije. To su najčešće web portali i aplikacije za rukovanje elektroničkom poštom ili razgovor putem Interneta. Pri tome nije nužno da korisnik slijedi neku poveznicu, već je dovoljno da jednostavno pregleda sadržaj web stranice koja sadrži štetan kod.

Jedan od načina sprečavanja XSS napada uključuje izmjenu koda web aplikacija, odnosno dodavanje programskog koda koji ignorira određene HTML oznake, kao što su <SCRIPT>, <OBJECT>, <APPLET>, <EMBED> i <FORM>.

Postoje i dodatne sigurnosne mjere za izbjegavanje XSS napada:

- Onemogućavanje izvođenja *JavaScript* programskog koda - ovako se sprečava izvođenje programskog koda na klijentskoj strani, ali i dalje postoji opasnost od posebno oblikovanih HTML dokumenata koji se najčešće prosljeđuju korisniku putem elektroničke pošte.
- Filtriranje korisničkih zahtjeva - web aplikacija će provjeravati korisničke zahtjeve i filtrirati posebne meta znakove definirane HTML specifikacijom. Ukoliko zahtjev sadrži skriptu, web aplikacija će spriječiti prikazivanje HTML dokumenta unutar korisničkog web preglednika.
- Kodiranje stranica - pravilnim kodiranjem stranica na web poslužitelju može se onemogućiti nenamjerno pokretanje štetnog programskog koda.

- Vezanje sjedničkih *cookie* datoteka za IP adresu korisnika - na ovaj način moguće je spriječiti krađu sjednica. *Cookie* datoteka je vezana uz IP adresu prijavljenog korisnika, tako da web aplikacija dozvoljava upotrebu isključivo te *cookie* datoteke.

Na Internetu je moguće pronaći mnogo web stranica koje sadrže XSS ranjivost, a neki od popularnih napada u posljednje vrijeme su upadi u sustave video nadzora kojima se može pristupiti putem web aplikacija.

2.2. *SQL injection* napad

SQL injection je napadačka tehnika koja koristi sigurnosnu ranjivost kod pristupa web aplikacije bazi podataka. Na taj način moguće je ugroziti sigurnost web aplikacije koja konstruira SQL upite iz podataka unesenih od strane korisnika.

SQL (eng. *Structured Query Language*) je specijalizirani programski jezik namijenjen rukovanju bazama podataka putem izjava i upita. SQL je ANSI i ISO standard te je najraširenij jezik za rukovanje bazama podataka koji koristi većina današnjih web aplikacija. Web aplikacije mogu koristiti korisnički unesene podatke kako bi stvorile posebne SQL upite za rad s dinamički generiranim web stranicama.

Ranjivost se javlja uslijed neodgovarajućeg filtriranja znakova posebne namjene od kojih se kreira SQL izjava ili u slučaju kada nisu implementirane određene restrikcije vezane uz ulazne podatke od kojih se stvaraju SQL upiti. Napadač spomenute ranjivosti može iskoristiti za izmjenu konstrukcije pozadinskog SQL upita. Ukoliko napadač uspije izmijeniti upit, on će se izvesti s dozvolama izvorne SQL naredbe. Posljedica napada je preuzimanje kontrole nad podacima u bazi podataka.

Neki od nizova znakova koje napadač može unijeti su:

- ' OR "='
- ' OR 1=1—
- 1 AND 1=1
- 1'1

Na primjer, ukoliko se od korisnika traži unos korisničkog imena i zaporke na web stranici, napadač može upisati jedan od prethodno spomenutih nizova znakova. Rezultat konstruirane SQL izjave je uvijek istinit i napadač će se uspjeti prijaviti na sustav kao prvi korisnik u korisničkoj tablici.

Postoje dvije vrste *SQL injection* napada, slijepi i normalni. Slijepi *SQL injection* napad je zapravo metoda pokušaja i pogrešaka. Provodi se tzv. ručnim pretraživanjem, odnosno proučavanjem različitih aplikacijskih ulaznih podataka i ubacivanjem posebnih znakova. Pri radu s bazama podataka najčešće se povratna informacija o pogrešci dobije u obliku web stranice s opisom pogreške pa potencijalni napadač može približno odrediti sintaksu SQL izjava nad određenom bazom podataka i izvesti *SQL injection* napad. Upravo je zbog toga kod razvoja web aplikacija važno obratiti posebnu pozornost detaljnosti opisa pogreške u sadržaju stranica, kako se ne bi razotkrile suvišne informacije te tako pomoglo napadaču.

Kako bi se minimizirala mogućnost pojave *SQL injection* napada uvijek je potrebno ocijeniti ispravnost ulaznih podataka testiranjem tipa podataka, formata, duljine niza itd.

Neke od sigurnosnih mjera za sprečavanje *SQL injection* napada su:

- ispitivanje veličine, tipa i sadržaja ulaznih podataka - npr., testiranje ponašanja aplikacije ukoliko korisnik unese MPEG datoteku veličine 10 MB umjesto poštanskog broja,
- testiranje sadržaja nizova znakova i prihvaćanje isključivo očekivanih vrijednosti odbacivanjem unosa koji sadrže binarne podatke i posebne znakove moguće je spriječiti napadača u pokušaju pokretanja proizvoljnog programskog koda,
- validacija XML datoteka i spajanje ulaznih podataka isključivo nakon validacije,
- upotreba pohranjenih procedura,
- postavljanje sigurnosne zone kod višedretvenih aplikacija - podaci koji ne prođu kroz sigurnosnu zonu se odbacuju uz prijavu pogreške. te
- ne prihvaćanje nekih nizova od kojih se konstruiraju nazivi datoteka kao što su AUX, CLOCK$, COM1 do COM8, CON, CONFIG$, LPT1 do LPT8, NUL, i PRN.

Neki od nedavnih poznatijih napada su oni izvedeni nad stranicama Ujednjenih Naroda, u kolovozu 2007. te nad stranicama tvrtke Microsoft U.K, u lipnju 2007.

3. *Exploit Me* alati

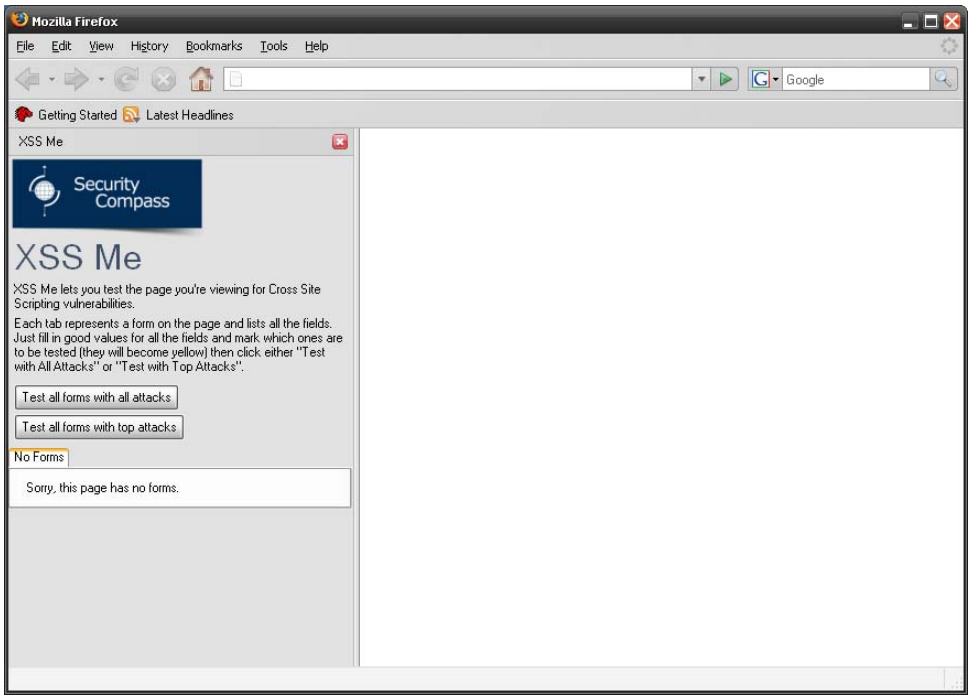
Exploit Me je besplatan skup alata za provjeru ranjivosti web aplikacija namijenjen pokretanju u sklopu *Firefox* programskog paketa. Programski paket je proizvod tvrtke *Security Compass* koju održavaju i mnogi njihovi suradnici, a izdaje se pod GPL (eng. *GNU Public License*) licencom. Sastoji se od dva alata, *XSS Me* i *SQL Inject Me*. *XSS Me* je namijenjen provjeravanju postojanja XSS ranjivosti, a *SQL Inject Me* otkrivanju ranjivosti na *SQL injection* napade. *Exploit Me* alati su dizajnirani za laku i jednostavnu uporabu te ne zahtijevaju korištenje posrednog poslužitelja (eng. *proxy*) kao mnogi drugi alati za testiranje spomenutih ranjivosti. Instalacijom se integriraju u *Firefox* web preglednik.

3.1. *XSS Me*

XSS Me je, kako je već ranije spomenuto, alat za testiranje web aplikacija na XSS ranjivosti. On ne pokušava ugroziti sigurnost sustava koji se testira, već traži moguće ulazne točke za napad na sustav. Ne skeniraju se mrežni priključci (eng. *ports*), ne analiziraju se paketi koji se šalju različitim protokolima, ne razbijaju se zaporke niti se napada vatrozid (eng. *firewall*) sustava koji se testira. Rad alata je moguće zamisliti kao ručno ispitivanje ranjivosti, odnosno kao da zlonamjeran korisnik upisuje različite nizove znakova u polja koja nudi web aplikacija.

XSS Me detektira ranjivosti upisivanjem nizova znakova specifičnih za XSS napad u polja namijenjena upisu podataka u web aplikaciju. Osim ove metode, postoje i mnogi drugi načini izvođenja XSS napada koje ovaj alat ne otkriva, a to su na primjer pokretanje pohranjenih skripti, podmetanje posebno oblikovanih *cookie* datoteka, internetskih poveznica ili HTTP zaglavlja. Svakim danom napadači otkrivaju nove načine izvođenja XSS napada na web sustave, tako da nije moguće realizirati otkrivanje ranjivosti na svaki mogući način napada.

XSS Me alat sadrži popis posebnih nizova znakova koji se testiraju. Ti nizovi znakova, odnosno napadački izrazi preuzeti su sa *RSnake's XSS cheat sheet* popisa koji je načinjan u rujnu 2007. Ovaj popis smatra se osnovnim popisom XSS napadačkih nizova znakova. Ipak, uvijek postoji mogućnost da zlonamjerni napadači poznaju i neke druge izraze za napade na ranjive sustave, nepoznate sigurnosnoj industriji. Također, mnogo je načina na koje se mogu kodirati napadački izrazi. Takvi kodirani izrazi mogu proći kroz filtere postavljene na web aplikaciji namijenjene upravo zaštiti od napada. Trenutna inačica XSS Me alata nema ugrađenu podršku za različite načine kodiranja napadačkih nizova znakova, ali postoje planovi ugrađivanja takve funkcionalnosti u program. Osim korištenja ugrađenog popisa, osoba koja provjerava ranjivosti XSS Me alatom može zadati i vlastite napadačke izraze čime se povećava temeljitost testiranja web aplikacija.

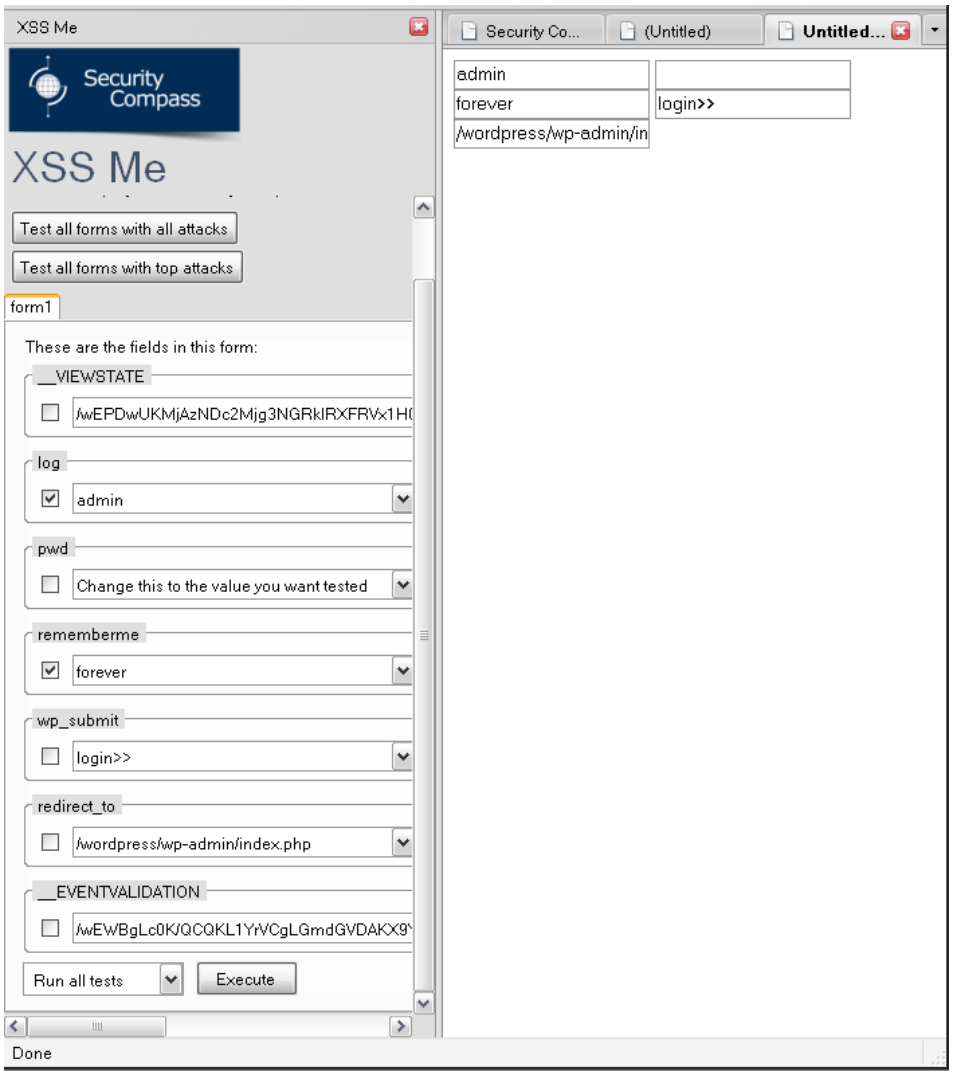


Slika 1. Izgled *XSS Me* alata u *Firefox* web pregledniku

3.1.1. *XSS Me* – instalacija i opcije

XSS Me alat se instalacijom integrira u *Firefox* web preglednik. Za rad alata potrebno je imati inačicu 2.0.0.9 *Firefox* web preglednika, ili neku noviju. Alat se instalira tako da se nakon preuzimanja jednostavno pokrene *Firefox* preglednikom i program se automatski uklapa u nj. Nakon instalacije potrebno je ponovno pokrenuti web preglednik.

XSS Me se pokreće odabirom iz izbornika: *Tools* → *XSS-Me* → *Open XSS Me Sidebar* ili se može koristiti izbornik koji se pojavi desnim klikom miša.



Slika 2. *XSS Me* alat nakon učitavanja web stranice

Moguće je uočiti da se svi dijelovi učitane web stranice u koje se mogu upisati neke ulazni podaci pojavljuju u *XSS Me* alatu s lijeve strane web preglednika. *XSS Me* pruža mogućnost upisa podataka u polja vidljiva korisniku, ali i u ona polja koja korisnik ne vidi na učitanoj stranici (npr. _EVENTVALIDATION polje ili _VIEWSTATE).

Trenutna vrijednost svakog elementa za upis podataka pojavljuje se u izborniku programa prikazanom sa strane web preglednika. Vrijednosti koje se upisuju u te elemente korisnik može mijenjati izravno u *XSS Me* alatu, a one će se pojaviti i na učitanoj stranici (kao što je moguće vidjeti na slici). Pretpostavljena vrijednost svakog polja jest ona vrijednost koja se nalazi u elementu u trenutku učitavanja stranice. Ukoliko ona nije postavljena alat umjesto nje prikazuje izraz: "*Change this to the value you want tested*" (na slici je to, primjerice, polje "pwd").

Ako se postavi kvačica pored polja u koje se unose ulazne vrijednosti, tada je to polje označeno za testiranje na XSS ranjivost. Ukoliko kvačice nema, tada se polje neće testirati, a trenutna vrijednost koja se nalazi u polju ostat će ista tokom svakog testiranja.

XSS Me radi tako da ispituje svaku označenu vrijednost, jednu po jednu. U primjeru na slici 2. alat će testirati polje "log" pa zatim "rememberme". Pri tome parametri koji se unose u web stranicu u slučaju testiranja "log" polja izgledaju ovako:

```
log=XSS_ATTACK_STRING&pwd=&rememberme=forever&wp-submit=Login
>>&redirect_to=/wordpress/wp-admin/index.php
```

U slučaju testiranja "rememberme" polja imaju ovakav izgled:

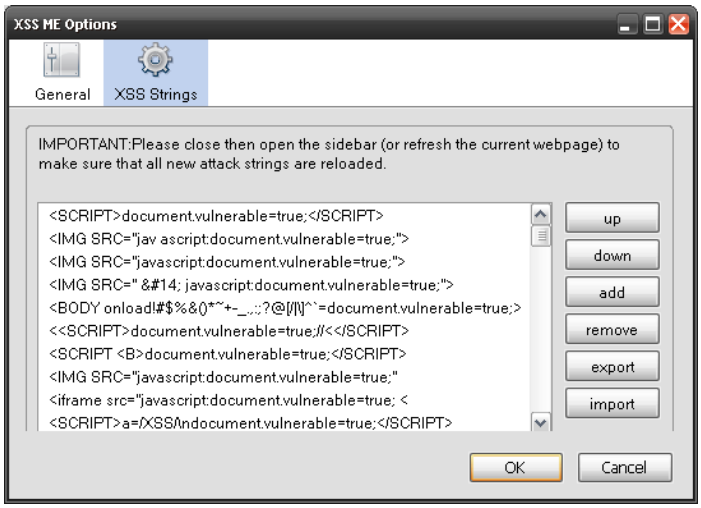
```
log=admin&pwd=&rememberme=XSS_ATTACK_STRING&wp-submit=Login
>>&redirect_to=/wordpress/wp-admin/index.php
```

U gornjim izrazima moguće je uočiti izraz "XSS_ATTACK_STRING". Kod testiranja umjesto ovog izraza upisuju se vrijednosti iz polja koja se žele testirati. Ovaj postupak se naziva *fuzzing*. Program pruža mogućnost zamjene svih izraza "XSS_ATTACK_STRING" odabirom *Run all tests* opcije ili pak zamjene samo nekih izraza odabirom *Run top X attacks* opcije. Pokretanje svih testova s izrazima iz pretpostavljene liste napadačkih izraza može biti dugotrajno, pogotovo ako su poslužitelji koji odgovaraju na zahtjeve spori, ili ako je potrebno testirati nekoliko ulaznih elemenata. Pokretanje alata uz opciju *Run top X attacks* nije toliko opsežno te se testiranje odvija mnogo brže, u ovisnosti o količini označenih napada za izvođenje.
Osim spomenutih opcija postoje i dvije tipke s natpisima *Test all forms with all attacks* i Test *all forms with top attacks*. Pritiskom na tipku *Test all forms with all attacks* automatski će se testirati sva polja na svakom obrascu na web stranici sa svim pretpostavljenim napadačkim izrazima, a klikom na tipku *Test all forms with top attacks* testirat će se samo sa upisanim vrijednostima u označena polja. Ukoliko se odabere jedna od dvije tipke, ignoriraju se opcije selektirane na dnu stranice.
Do dodatnih opcija programa može se doći odabirom u izborniku *Firefox* preglednika *Tools* → *XSS-Me* → *Options*.

Slika 3. Opcije *XSS Me* alata

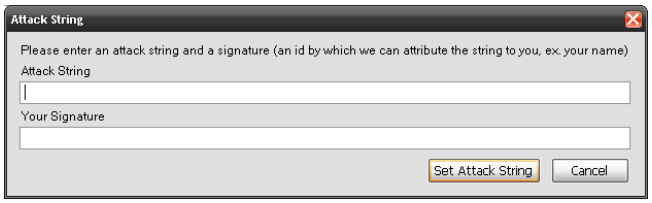
Slika prikazuje tri opcije:

- Željeni broj testnih napada (*Preferred Number of Attacks to Test*). Tu se određuje koliko će se napada izvesti kada je odabrana opcija *Test All Forms with Top Attacks* ili *Run Top X Attacks*. Ukoliko se, primjerice, vrijednost polja postavi na "9", tada će se testirati prvih 9 vrijednosti iz tablice napadačkih izraza.
- Odabir prikaza izbornika kada se klikne desna tipka miša.
- Broj kartica (eng. *tabs*) koje se koriste za testove. Opcija specificira koliko se kartica istovremeno pokreće za izvođenje testova za XSS ranjivost. Otvaranje više kartica znači manje vremena za testiranje, ali konzumira se mnogo memorije pa treba odabrati optimalan broj kartica u ovisnosti o računalu na kojem se testovi izvode. Ukoliko se otvori prevelik broj kartica može doći do rušenja *Firefox* preglednika.



Slika 4. XSS *Strings* opcije

Slika 4. prikazuje izraze koji se upisuju u ulazna polja sa obrasca na učitanoj web stranici. To su izrazi koji će se testirati. Alat testira sve izraze, od prvog do zadnjeg ukoliko je odabrana opcija *Test All Forms with Top Attacks*, a testira samo određeni broj izraza ukoliko je odabrana opcija *Run Top X Attacks* ("X" se definira u prethodno spomenutim opcijama). Ako se želi promijeniti redoslijed izvođenja testnih izraza, to se može postići tipkama *Up* i *Down*. Također, moguće je dodati ili ukloniti pojedini niz znakova te presnimiti cijelu listu na neku drugu lokaciju ili učitati neku drugu listu izraza.
Alat omogućuje dodavanje proizvoljnih napadačkih izraza:



Slika 5. Dodavanje proizvoljnih testnih izraza

Željeni se izraz upisuje u tekstualno polje *Attack String*. Potrebno je napomenuti da proizvoljno upisani niz znakova treba u sebi sadržavati i pokretati naredbu "document.vulnerable=true" u stranici koja se testira. Ukoliko naredba nije sadržana, alat neće moći ispravno raditi. Slijedi primjer ispravnog izraza:

```
<script>document.vulnerable=true</script>
```

U tekstualno polje *Your signature* upisuje se određeni nadimak ili ime koje se povezuje s napadačkim izrazom. Ovo polje postoji kako bi se ljudima koji su smislili napadački izraz mogle dodijeliti zasluge za njihov doprinos programu.

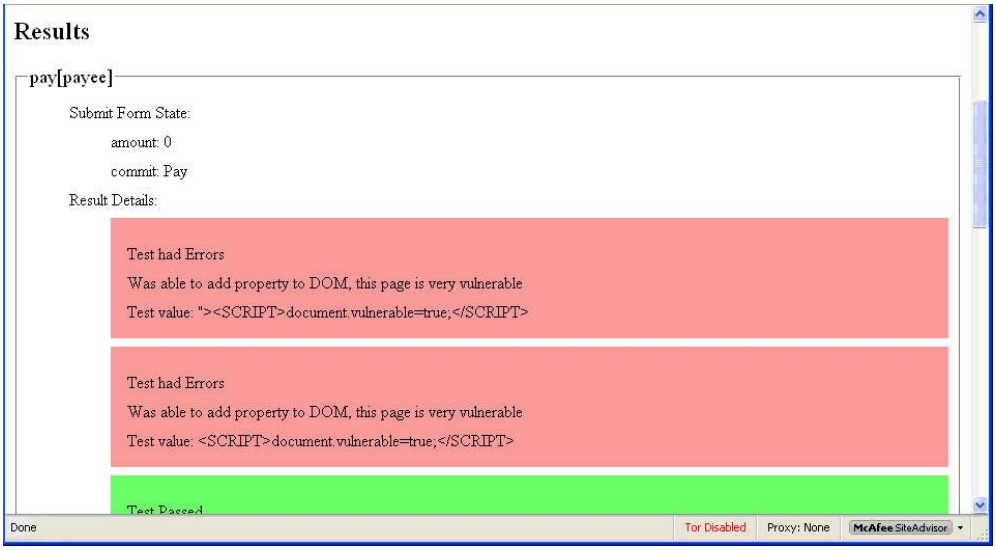
3.1.2. *XSS Me* – primjer izvođenja i interpretacija rezultata

Alat radi tako da nakon učitavanja web stranice u Firefox preglednik zamjenjuje vrijednosti ulaznih polja prikazane stranice s nizovima znakova koji su tipični za XSS napad. Ukoliko rezultantna HTML stranica postavlja specifičnu *JavaScript* vrijednost ((document.vulnerable=true)), tada alat označava web stranicu kao ranjivu na XSS napad za upisani izraz.
Nakon što se odabere stranica koja će se testirati na XSS napad, odaberu se i postave opcije koje nudi alat, upišu se napadački izrazi i pokrene se test ranjivosti. Kad je testiranje završeno *XSS Me* prezentira rezultate izvođenja u novoj kartici u web pregledniku. Postoje tri tipa rezultata koja *XSS Me* prikazuje, a to su promašaji (eng. *Failures*), upozorenja (eng. *Warnings*) i prolazi (eng. *Passes*). Promašaji predstavljaju broj testova koji su sigurno detektirali XSS ranjivost, dok se upozorenjima obavještava

ispitivača o broju testova koji možda sadrže XSS ranjivost (na primjer testom se nije uspio promijeniti DOM objekt u *Firefox* web pregledniku, ali je možda moguće izvesti napad iz nekog drugog web preglednika. Prolaz označava broj testova koji nisu otkrili XSS ranjivost.
Slijedeća slika prikazuje rezultate jednog testiranja:

Slika 6. Prikaz broja padova, upozorenja i prolaza

Svi su rezultati detaljno prikazan po odjeljcima i grupirani su prema nazivu ulaznog polja. Prvo su popisani promašaji, zatim upozorenja pa tek onda prolazi:



Slika 7. Detaljan prikaz rezultata ispitivanja

Za svako ulazno polje dani su slijedeći detalji:

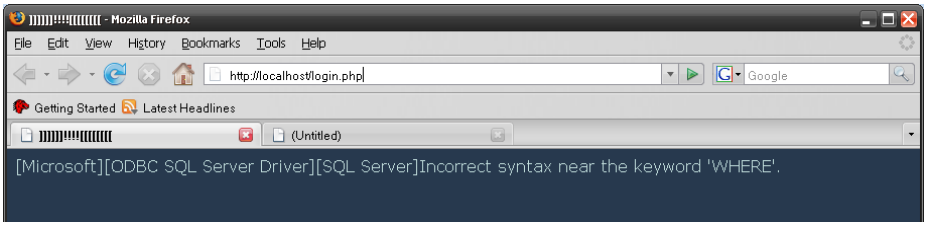
- stanje obrasca - prikazuje vrijednosti svih parametara tokom podnošenja obrasca, te
- detalji rezultata - sadrži opis pojedinih neuspjeha, upozorenja i prolaza, kao i testni izraz koji je uzrokovao promašaj. Ove su informacije bitne zbog određivanja na koji način je pojedino ulazno polje ranjivo. Ispitivač može uzeti testne izraze koji su rezultirali promašajem na nekom od testova i napisati proizvoljni *JavaScript* kod (npr. `alert("XSS")`) te ručno provjeriti ranjivost.

Rezultati testiranja pokazuju moguću ranjivost na DOM model napada.

3.2. SQL Inject Me

SQL Inject Me je alat za testiranje web aplikacija na *SQL injection* ranjivosti. Način rada analogan je *XSS Me* alatu, dakle traži moguće ulazne točke za napad na sustav. Rad alata moguće je usporediti ručnim ispitivanjem. Testiranje ranjivosti izgleda kao da zlonamjeran korisnik upisuje različite nizove znakova u polja obrazaca na web aplikaciji, pri čemu to alat izvodi mnogo brže i efikasnije. *SQL Inject Me* ne analizira niti izvorni kod, niti računalnu mrežu već isključivo testira kreirane web aplikacije. Programom nije moguće kompromitirati ispitivani sustav, niti izvesti napad na njega.

Alat traži neočekivane odgovore poslužitelja, odnosno posebne izraze koji se koriste tokom dojave pogreške te je zbog toga mogućnost otkrivanja *SQL Injection* ranjivosti ograničena sadržajem i detaljima primljenih odgovora. Neke pogreške koje dojavi poslužitelj, alat neće prepoznati jer je pogreška dojavljena sa izrazom koga *SQL Inject Me* ne provjerava. Testiranje na slijepe *SQL Injection* napade zahtijeva dodatno ručno testiranje, primjerice iskušavanje zaobilaženja autentikacije. Slijedeća slika prikazuje jedan neprimjereni odgovor poslužitelja koga *SQL Inject Me* nije prepoznao kao moguću ranjivost:

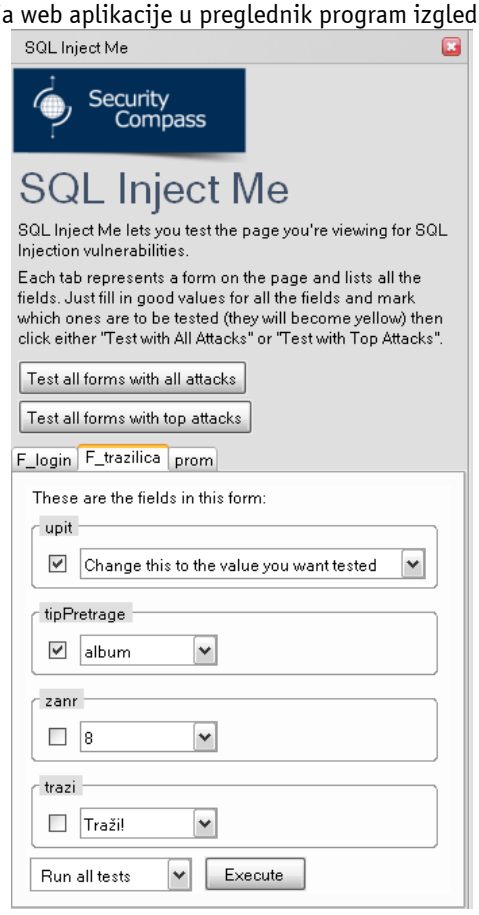


Slika 8. Odgovor poslužitelja nakon napadačkog izraza '1 OR 1=1'

Iako *SQL Inject Me* odgovor poslužitelja nije prepoznao kao moguću ranjivost, takvi odgovori su neprilagođeni i otkrivaju informacije o strukturi SQL upita koje napadač može zlouporabiti. Ipak, *SQL Inject Me* ispitivaču nudi opciju dodavanja proizvoljnih odgovora koje ranjivi poslužitelj može poslati.

3.2.1. *SQL Inject Me* – pokretanje i opcije

SQL Inject Me alat pokreće se analogno *XSS Me* alatu, tj. odabirom u izborniku *Firefox* web preglednika: *Tools → SQL Inject Me → Open SQL Inject Me Sidebar*.
Nakon pokretanja i učitavanja web aplikacije u preglednik program izgleda ovako:



Slika 9. SQL Inject Me alat nakon pokretanja

Kao i kod *XSS Me* alata, *SQL Inject Me* alat prikazuje trenutne vrijednosti ulaznih polja učitane stranice. Također je potrebno staviti kvačicu pored polja koje se želi testirati.
Program testira jednu po jednu kvačicom označenu vrijednost, a u primjeru na prethodnoj slici testirat će se polja "upit" i "tipPretrage". Parametri koji se pri tome postavljaju interno kod testiranja izgledaju ovako:

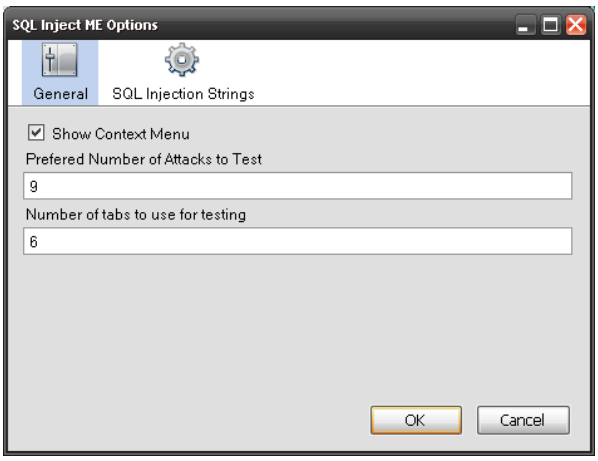
- U slučaju testiranja ulaznog polja "upit":

```
upit=SQLInjection_ATTACK_STRING&searchType=web
```

- U slučaju testiranja polja "upit" i "tipPretrage":

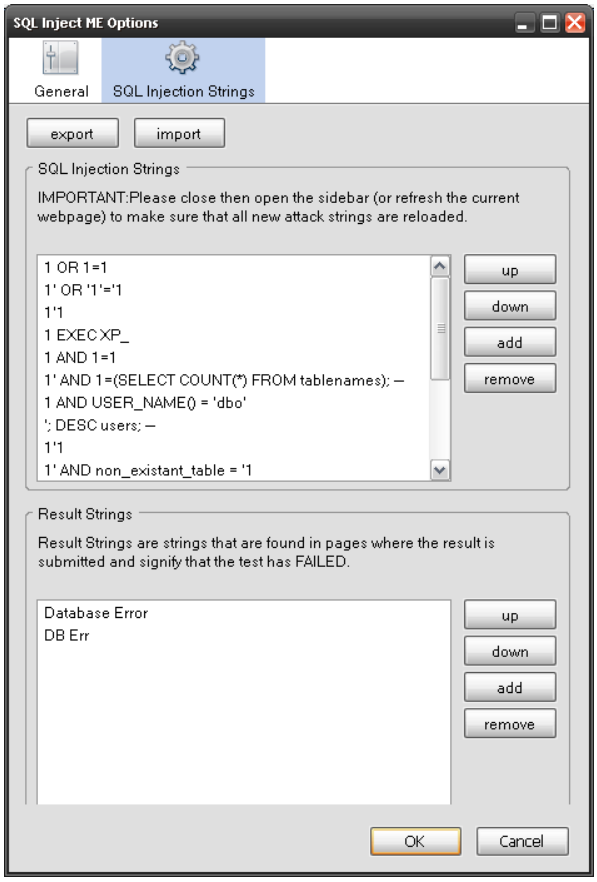
```
upit=&tipPretrage=SQLInjection_ATTACK_STRING
```

Dio izraza i kojem je zapisan niz znakova "SQLInjection_ATTACK_STRING" zamjenjuje se listom odabranih nizova znakova specificiranim u opcijama. Ova zamjena se naziva *fuzzing*, kao i kod *XSS Me* alata. Opcije odabira testova sa svim napadačkim izrazima ili samo s određenim rade jednako kao kod *XSS Me* alata.
Postoji nekoliko opcija *SQL Inject Me* alata koje se razlikuju od opcija *XSS Me* alata, a vezane su uz napadačke nizove. Do tih postavki moguće je doći iz izbornika web preglednika: *Tools→ SQL Inject Me → Options*. Tada će se ukazati sučelje kao na slijedećoj slici.



Slika 10. Opcije *SQL Inject Me* alata

Opcije prve kartice pod nazivom *General* su jednake kao kod *XSS Me* alata, ali opcije pod nazivom *SQL Injection Strings* se razlikuju. Program unosi napadačke izraze definirane na popisu. Popis je moguće spremiti na neku proizvoljnu lokaciju na računalu, a korisnik ga može mijenjati i nadopunjavati prema vlastitim potrebama.
Osim definiranja napadačkih nizova znakova, moguće je dodavati ili mijenjati i izraze koje alat treba tražiti u sadržaju odgovora od poslužitelja web stranica koje se testiraju. Dakle, *SQL Me* traži zadane nizove znakova u HTTP odgovorima poslužitelja. Ukoliko je bilo koji od tih dodanih izraza (poruka o pogrešci) pronađen, stranica se stavlja u kategoriju onih za koje postoji vrlo velika vjerojatnost da sadrže *SQL Injection* ranjivost.



Slika 11. *SQL Injection Strings* opcije

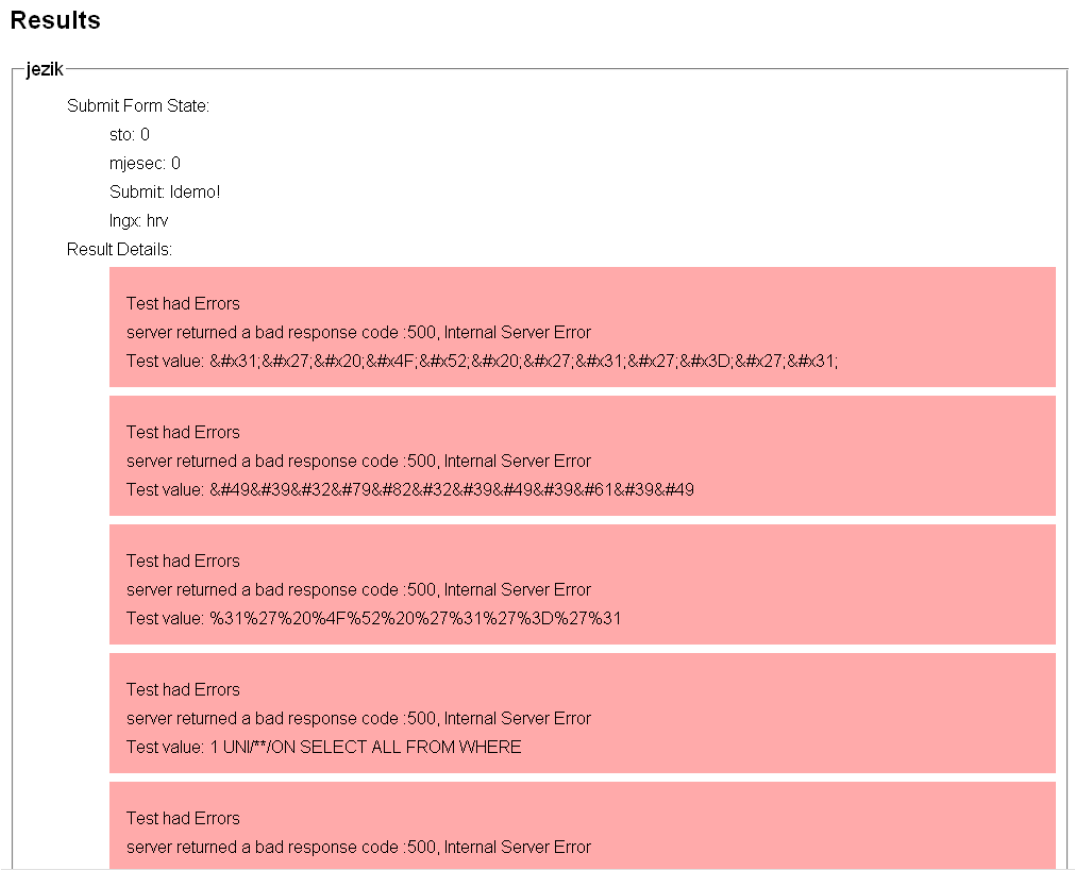
3.2.2. *SQL Inject Me* – primjer izvođenja i interpretacija rezultata

Alat radi tako da na obrascima učitanih stranica zamijeni ulazne vrijednosti sa napadačkim nizovima znakova reprezentativnim za *SQL Injection* napade.
Kao i kod *XSS Me* alata postoje tri vrste rezultata:

- Promašaji (eng. *Failures*) – broj testova koji su detektirali postojanje potencijalne ranjivosti na *SQL Injection* napad.
- Upozorenja (eng. *Warnings*) – broj testova čiji rezultat upućuje na mogućnost postojanja *SQL Injection* ranjivosti (npr. postoji razlika u odgovoru poslužitelja nakon što je upisana uobičajena ulazna vrijednost i nakon testiranja napadačkog izraza).
- Prolaz (eng. *Passes*) – broj testova koji nisu otkrili postojanje *SQL Injection* ranjivosti.

Slijedeće slike prikazuju rezultate izvođenja jednog testa (vidljiv je pronalazak 46 mogućih *SQL Injection* ranjivosti):

Slika 12. Rezultati nakon testiranja



Slika 13. Rezultati nakon testiranja

Kad se pogledaju detalji rezultata testa, vidljivo je da je *SQL Inject Me* zapravo pročitao odgovore ranjivog poslužitelja. Takvi odgovori su neprimjereni i trebali bi, u svrhu sigurnosti web aplikacija, poslužitelja i sustava biti skriveni. Upisom u postavkama za niz znakova dodan je izraz *Incorrect syntax near the keyword* te je ponovno pokrenuto testiranje vratilo je još više promašaja, točnije pedeset. *SQL Inject Me* uz dojavu o mogućoj *SQL Injection* ranjivosti prikazuje i testni izraz, tako da ispitivač može točno utvrditi gdje se moguća ranjivost nalazi i ispraviti ju.

4. Zaključak

U današnje je vrijeme sigurnost aplikacija bitan faktor u procesu njihove izgradnje. Kako bi se očuvali podaci i onemogućio pristup neovlaštenim korisnicima, posebno je važno smanjiti mogućnosti XSS i *SQL injection* napada. Mnogo je razloga zašto programeri pišu ranjive aplikacije, a većina ih se može svesti na nedovoljnu educiranost i zanemarivanje važnosti sigurnosne komponente. Izučavanje računalne sigurnosti može biti naporan posao koji iziskuje znatno vrijeme potrebno za dodatne analize napada i smišljanje njihove prevencije. Svaka web aplikacija može sadržavati sigurnosni propust koji je rezultat najobičnije nepažnje, a koji može biti poguban, kako za stabilnost aplikacije, tako i za podatke koje ona štiti.

Alati za otkrivanje ranjivosti web aplikacija od velike su pomoći programerima pri stvaranju sigurnih aplikacija. Opisani *Exploit Me* jedan je od takvih. Korištenjem njegovih funkcionalnosti programeri mogu otkriti točan uzrok problema, a sve to daleko prije završetka razvoja, odnosno daleko prije eventualnih napadača. Unatoč tome, *Exploit Me* skup alata ima i svoja ograničenja te rezultate njegovog rada nikako ne treba uzeti zdravo za gotovo kao jamstvo neranjivosti, već samo kao korisnu sugestiju o mogućim problemima.

5. Reference

[1] XSS napadi, http://en.wikipedia.org/wiki/Cross-site_scripting, siječanj 2008.

[2] *SQL Injection* napadi, http://en.wikipedia.org/wiki/SQL_injection, siječanj 2008.

[3] Što je "slijepi" *SQL Injection* napad, http://www.cgisecurity.com/questions/blindsql.shtml, siječanj 2008.

[4] *SQL Injection*, http://technet.microsoft.com/en-us/library/ms161953.aspx, rujan 2007.

[5] *XSS Me* – često postavljana pitanja, http://www.securitycompass.com/exploit_me/xssme/xssme_faq.shtml, siječanj 2008.