The OWASP Foundation
http://www.owasp.org

OWASP Slovenija
(Open Web Application Security Project)

Predstavitev največjih tveganj spletnih aplikacij - OWASP TOP 10

Jure Škofič

Maribor, 29.3.2012

7 nasvetov SI-CERTa

The OWASP Foundation
http://www.owasp.org

1. **Redno in strokovno kompetentno vzdrževanje omrežnih naprav in strežnikov** je pomemben del zagotavljanja informacijske varnosti v podjetjih in vseh drugih ustanovah.
2. **Organizacije naj načrtujejo izvedbo penetracijskega testa** v lastno omrežje s pomočjo neodvisnega zunanjega podjetja, ki ima ustrezne strokovne reference.
3. **Razvoj spletnih aplikacij naj sledi smernicam združenja OWASP.** Skladnost z njimi naj bo del razpisne dokumentacije za proračunsko financirana naročila.
4. **Razvoju spletnih aplikacij naj sledi strokovni varnostni pregled izvorne kode.** Na aplikaciji naj se izvede varnostni pregled. Zahteva za izvedbo neodvisnega varnostnega pregleda naj bo del proračunsko financiranih naročil.
5. **Neodvisni varnostni pregled in penetracijski test naj postaneta nujna** za vse sisteme, ki so del kritične infrastrukture. Varnostna ocena naj bo izvedena tudi za sisteme za upravljanje pametnih števcev na daljavo.
6. **Digitalni podpis omogoča dodatno preverjanje vira sporočila** in zmanjšuje možnost okužbe preko zlonamernih priponk. Državne ustanove naj pričnejo z uporabo digitalno podpisanih sporočil v vsej komunikaciji po elektronski pošti.
7. **Občutljivi podatki naj bodo** pri hranjenju in posredovanju **šifrirani**.

Copyright © The OWASP Foundation
Permission is granted to copy, distribute and/or modify this document under the terms of the OWASP License.

VIR: Poročilo o omrežni varnosti za I. 2012, ARNES

The OWASP Foundation
http://www.owasp.org

OWASP Top 10

Jure Škofič
jure.skofic@acrossecurity.com

OWASP Top 10

- Najbolj kritična tveganja za spletne aplikacije

Agent	Težavnost vektorja napada	Razširjenost tveganja	Težavnost odkrivanja tveganja	Tehnični vpliv	Vpliv na poslovanje
?	Lahka	Zelo razširjena	Lahka	Resen	?
	Povprečna	Običajna	Povprečna	Zmeren	
	Težka	Neobičajna	Težka	Lažji	

- Odličen uvod v aplikacijsko varnost
- Aktualen seznam objavljen leta 2010
- Razširjen in dobro znan
- Kljub temu večina aplikacij še vedno ima kritične ranljivosti

OWASP Top 10

A1: Injection

A2: Cross-Site Scripting

A3: Broken Authentication and Session Management

A4: Insecure Direct Object References

A5: Cross-Site Request Forgery (CSRF)

A6: Security Misconfiguration

A7: Insecure Cryptographic Storage

A8: Failure to Restrict URL Access

A9: Insufficient Transport Layer Protection

A10: Unvalidated Redirects or Forwards

OWASP Top 10

A1: Vrivanje

A2: Podtikanje skript

A3: Napaka pri avtentikaciji in upravljanju sej

A4: Nezavarovan neposreden dostop do objektov

A5: Potvarjanje spletnih zahtevkov

A6: Napake v varnostnih nastavitvah

A7: Nezadostna zaščita kriptografskih podatkov pri hrambi

A8: Neprimerna zaščita neposrednega dostopa do URL-ja

A9: Nezadostna zaščita podatkov pri prenosu

A10: Nepreverjene preusmeritve brskalnika

A10: Nepreverjene preusmeritve brskalnika

A10: Nepreverjene preusmeritve brskalnika

Težavnost vektorja napada	Razširjenost tveganja	Težavnost odkrivanja tveganja	Tehnični vpliv
Povprečna	Neobičajna	Lahka	Zmeren

- Preusmeritve pri spletnih aplikacijah so nekaj vsakdanjega
  - Ciljni URL-ji mnogokrat vsebujejo parametre, ki jih posreduje uporabnik
  - Če teh podatkov ne validiramo, lahko napadalec žrtev preusmeri na kateri koli spletni naslov
- Posredovanja
  - Pošljejo notranji zahtevek za novo stran na isti aplikaciji
  - Včasih parametri določajo ciljno stran
  - Če jih ne validiramo, bi lahko napadalec s pomočjo posredovanja obšel določene avtentikacijske ali validacijske mehanizme
- Posledice
  - Žrtev preusmeri na stran, ki vsebuje zlonamerno kodo
  - Napadalčev zahtevek zaobide varnostna preverjanja in omogoči napadalcu nepooblaščen dostop do funkcij ali podatkov

A10: Nepreverjene preusmeritve brskalnika - zaščita

- Na voljo je več možnosti:
  - Izogibanje preusmeritvam in posredovanjem
  - Če se jim ne moremo izogniti, ne vključujmo parametrov, ki jih posreduje uporabnik
  - Če je vključevanje parametrov absolutno potrebno, potem:
    a) Validirajte vsak parameter, da zagotovite pravilnost in pooblaščenost za trenutnega uporabnika ali,
    b) Uporabite preslikave na strežniški strani
  - Zaščita v globino: Preverite vsak že obdelan preusmeritveni naslov, če kaže na pooblaščeno zunanjo stran
  - ESAPI – SecurityWrapperResponse.sendRedirect(URL)

A9: Nezadostna zaščita podatkov pri prenosu

A9: Nezadostna zaščita podatkov pri prenosu

Težavnost vektorja napada	Razširjenost tveganja	Težavnost odkrivanja tveganja	Tehnični vpliv
Težka	Običajna	Lahka	Zmeren

- Nevarno prenašanje občutljivih podatkov
  - Ne prepoznamo vseh občutljivih podatkov
  - Ne poznamo vseh lokacij, kamor se občutljivi podatki prenašajo
  - Ne uspe nam zavarovati podatkov na vseh lokacijah
- Posledice
  - Napadalec lahko spremeni poverilnice in zasebne podatke (kreditne kartice, zdravstveni podatki, finančni podatki)
  - Napadalec lahko pridobi skrivnosti, ki mu pomagajo pri nadaljnem napadu
  - Podjetje v zadregi, uporabniki nezadovoljni, izguba zaupanja
  - Stroški saniranja incidenta (forenzika, opravičilna pisma, novo izdajanje kreditnih kartic...)
  - Podjetje se lahko sooča s tožbo oz. kaznijo

A9: Nezadostna zaščita podatkov pri prenosu - zaščita

- Zaščita s primernimi mehanizmi
  - Uporaba TLS na vseh povezavah, preko katerih potujejo občutljivi podatki
  - Individualna enkripcija sporočil pred pošiljanjem (npr. XML enkripcija)
  - Podpisovanje sporočil pred pošiljanjem (npr. XML podpis)
- Pravilna uporaba mehanizmov
  - Uporaba standardnih in močnih algoritmov (onemogočite stare SSL algoritme)
  - Pravilno upravljanje s ključi/certifikati
  - Preverjanje SSL certifikatov pred uporabo
  - Uporaba preverjenih mehanizmov, če zadostujejo
- http://www.owasp.org/index.php/Transport_Layer_Protection_Cheat_Sheet

A8: Neprimerna zaščita neposrednega dostopa do URL-ja

A8: Neprimerna zaščita neposrednega dostopa do URL-ja

Težavnost vektorja napada	Razširjenost tveganja	Težavnost odkrivanja tveganja	Tehnični vpliv
Lahka	Neobičajna	Povprečna	Zmeren

- Kako zaščitimo dostop do določenih URL-jev?
  - Del uveljavljanja pravilne avtorizacije skupaj z A4: Nevarno Neposredno Naslavljanje Objektov
- Pogosta napaka
  - Prikazovanje (ne omejevanje) samo avtoriziranih povezav in izbir v meniju
  - "Presentation layer access control" – neučinkovito
  - Napadalec preprosto oblikuje povezavo do neavtoriziranih strani
- Posledice
  - Napadalec proži funkcije in storitve, do katerih naj ne bi imel dostopa
  - Dostop do uporabniških računov in podatkov
  - Izvajanje privilegiranih akcij

A8: Neprimerna zaščita neposrednega dostopa do URL-ja - zaščita

- Za vsak URL je potrebno storiti 3 stvari
  - Omejiti dostop na pooblaščene uporabnike
  - Uveljavljati vsakršne omejitve dostopa
  - Onemogočiti vsakršne zahtevke za nepooblaščene strani
- Preverite arhitekturo
  - Uporabite preprost pozitiven model na vsakem nivoju
  - Prepričajte se, da mehanizem uporabljate res na vsakem nivoju

A8: Neprimerna zaščita neposrednega dostopa do URL-ja - zaščita

- Preverite implementacijo
  - Uporaba avtomatiziranih pristopov je neprimerna
  - Preverite, da je vsak URL vaše aplikacije zaščiten z zunanjim filtrom (npr. Java EE web.xml) ali z vašo kodo (npr. Z uporabo ESAPI isAuthorizedForURL())
  - Preverite, da strežnik ne dovoljuje zahtevkov za nepooblaščene datotečne tipe
  - Uporabite WebScarab ali brskalnik za kreiranje nepooblaščenih zahtevkov

A7: Nezadostna zaščita kriptografskih podatkov pri hrambi

SONY MUSIC

A7: Nezadostna zaščita kriptografskih podatkov pri hrambi

Težavnost vektorja napada	Razširjenost tveganja	Težavnost odkrivanja tveganja	Tehnični vpliv
Težka	Neobičajna	Težka	Resen

- Ne prepoznamo vseh občutljivih podatkov
  - Ne najdemo vseh mest, kjer se shranjujejo občutljivi podatki (baze, datoteke, direktoriji, dnevniški zapisi, varnostne kopije...)
  - Pomanjkanje zadostne zaščite na vseh lokacijah

A7: Nezadostna zaščita kriptografskih podatkov pri hrambi

- Posledice
  - Napadalec lahko spremeni poverilnice in zasebne podatke (kreditne kartice, zdravstveni podatki, finančni podatki)
  - Napadalec lahko pridobi skrivnosti, ki mu pomagajo pri nadaljnem napadu
  - Podjetje v zadregi, uporabniki nezadovoljni, izguba zaupanja
  - Stroški saniranja incidenta (forenzika, opravičilna pisma, novo izdajanje kreditnih kartic...)
  - Podjetje se lahko sooča s tožbo oz. kaznijo

A7: Nezadostna zaščita kriptografskih podatkov pri hrambi - zaščita

- Preglejte arhitekturo
  - Identificirajte občutljive podatke
  - Identificirajte vse lokacije, kamor občutljive podatke shranjujete
  - Zagotovite, da model groženj (threat model) pokriva tovrstne napade
  - Enkripcijo uprabite, da preprečite napade
- Uporabite primerne mehanizme
  - Enkripcija datotek, baze, podatkovnih elementov

A7: Nezadostna zaščita kriptografskih podatkov pri hrambi - zaščita

- Pravilno uporabite mehanizme
  - Uporabljajte standardne močne algoritme
  - Generirajte, distribuirajte in ščitite ključe na pravilen način
  - Bodite pripravljeni na zamenjavo ključev
- Preverite implementacijo
  - Preverite, da se uporablja standarden močen algoritem in da je algoritem primeren za dano situacijo
  - Preverite, da so vsi ključi, certifikati in gesla shranjena na pravilen način in zaščitena.
  - Zagotovite varno razpečevanje ključev in učinkovit načrt za zamenjavo ključev
  - Varnostno preglejte kodo, ki je odgovorna za kriptiranje.

A6: Napake v varnostnih nastavitvah

A6: Napake v varnostnih nastavitvah

Težavnost vektorja napada	Razširjenost tveganja	Težavnost odkrivanja tveganja	Tehnični vpliv
Lahka	Običajna	Lahka	Zmeren

- Aplikacije se zanašajo na varne temelje
  - Vse od operacijskega sistema do strežnika
  - Vse knjižnice, ki jih aplikacija uporablja
- Je vaša izvorna koda skrivnost?
  - Premislite kje vse se vaša izvorna koda nahaja
  - Varnost ne bi smela biti odvisna od tajnosti izvorne kode

A6: Napake v varnostnih nastavitvah

- Tipične posledice
  - Stranska vrata zaradi manjkajočih varnostnih popravkov
  - Razne ranljivosti zaradi manjkajočih popravkov aplikacijskega ogrodja
  - Nepooblaščen dostop do privzetih računov, funkcionalnosti ali podatkov ali neuporabljenih funkcionalnosti, ki so na voljo zaradi slabe konfiguracije

A6: Napake v varnostnih nastavitvah - zaščita

- Upravljanje konfiguracije se naj uporablja za vse dele aplikacije
  - Zamenjava vseh poverilnic v produkcijski fazi
- Preverite upravljanje konfiguracije sistema
  - Upoštevajte smernice za utrjevanje varne konfiguracije (avtomatizacija zelo koristna)
  - Naj pokriva celotno platformo in aplikacijo
  - Sprotno nameščanje popravkov za VSE komponente
  - Analiza vpliva sprememb na varnost
- Preverite implementacijo konfiguracije

A5: Potvarjanje spletnih zahtevkov

A5: Potvarjanje spletnih zahtevkov

Težavnost vektorja napada	Razširjenost tveganja	Težavnost odkrivanja tveganja	Tehnični vpliv
Povprečna	Zelo razširjena	Lahka	Zmeren

- Je napad, pri katerem napadalec žrtvin brskalnik pripravi do tega, da pošlje zahtevek ranljivi spletni aplikaciji.
- Do ranljivosti pride zaradi samodejnega vključevanja avtentikacijskih podatkov v zahtevke (identifikator seje, IP naslov, Windows domenske poverilnice...).
- Napadalec lahko naredi vse, kar bi lahko naredili vi z miško in tipkovnico
- Med posledice spadajo:
  - Proženje transakcij
  - Dostop do občutljivih informacij
  - Spreminjanje informacij na računu

A5: Potvarjanje spletnih zahtevkov

- Problem:
  - Brskalniki vsakemu zahtevku avtomatsko pripnejo poverilnice.
  - Tudi če zahtevek prihaja s strani forme, skripte, ali druge spletne strani.
- Vse strani, ki se zanašajo izključno na avtomatske poverilnice, so ranljive.
- Avtomatske poverilnice:
  - Sejni piškotki
  - Zaglavje "Basic authentication"
  - IP naslov
  - Odjemalski SSL certifikati
  - Windows domenska avtentikacija

A5: Potvarjanje spletnih zahtevkov – zaščita

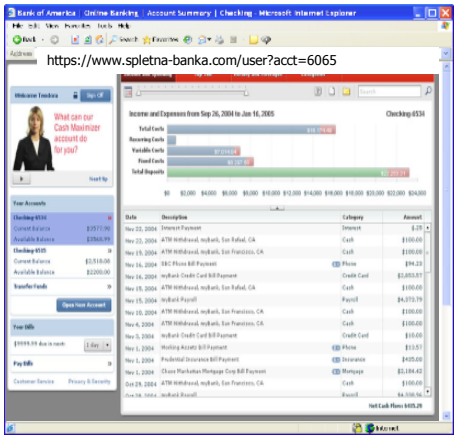
- Dodajanje tajnega žetona, ki se ne pošilja avtomatsko ob vsakem zahtevku, na vse kritične zahtevke.
  - Napadalec tako ne more ponarediti zahtevka (brez prisotnosti XSS ranljivosti)
  - Žetoni naj bodo kriptografsko močni in naključni
- Možnosti
  - CSRF žeton shranite v sejo in ga uporabite pri vseh formah in povezavah.
  - Ne izpostavljajte žetona v "referer header"
  - Lahko se uporabi edinstven žeton za vsako funkcijo
  - Sekundarna avtentikacija za kritične akcije
- Onemogočite "cross-site scripting" napade
  - Pravilno kodiranje vnosov
  - www.owasp.org/index.php/CSRF_Prevention_Cheat_Sheet

A4: Nezavarovan neposreden dostop do objektov

Sparkasse

citibank

A4: Nezavarovan neposreden dostop do objektov



- Napadalec opazi, da je parameter acct 6065

?acct=6065

- Parameter spremeni na 6066

?acct=6066

- Napadalec dobi dostop do računa žrtve

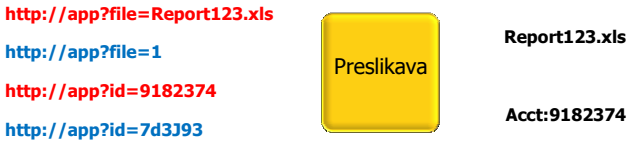
A4: Nezavarovan neposreden dostop do objektov

Težavnost vektorja napada	Razširjenost tveganja	Težavnost odkrivanja tveganja	Tehnični vpliv
Lahka	Običajna	Lahka	Zmeren

- Kako ščitite dostop do vaših podatkov?
  - Spada pod avtorizacijo, skupaj z A8: Neuspelo omejevanje dostopa preko URL-jev
- Zelo pogosta napaka...
  - Prikaz samo "avtoriziranih" objektov uporabniku ali
  - skrivanje referenc na objekte v skrita polja
  - ... in nato omejitev ne upoštevati na strani strežnika
  - Temu pravimo nadzor dostopa na predstavitvenem nivoju, kar ne deluje
  - Napadalec preprosto spremeni parametre
- Posledično lahko napadalec dostopa do podatkov, za katere ni avtoriziran

A4: Nezavarovan neposreden dostop do objektov - zaščita

- Odstranite neposredno naslavljanje objektov
  - Neposredno referenco zamenjajte z začasno vrednostjo
  - ESAPI nudi podporo za numerične in naključne preslikave



- Validirajte neposredno naslavljanje objektov
  - Preverite, če je vrednost parametra pravilno oblikovana
  - Preverite, če uporabnik lahko dostopa do objekta
  - Preverite, če je zahtevan način dostopa za ciljen objekt dovoljen (npr. branje, pisanje, brisanje)

A3: Napaka pri avtentikaciji in upravljanju sej

A3: Napaka pri avtentikaciji in upravljanju sej

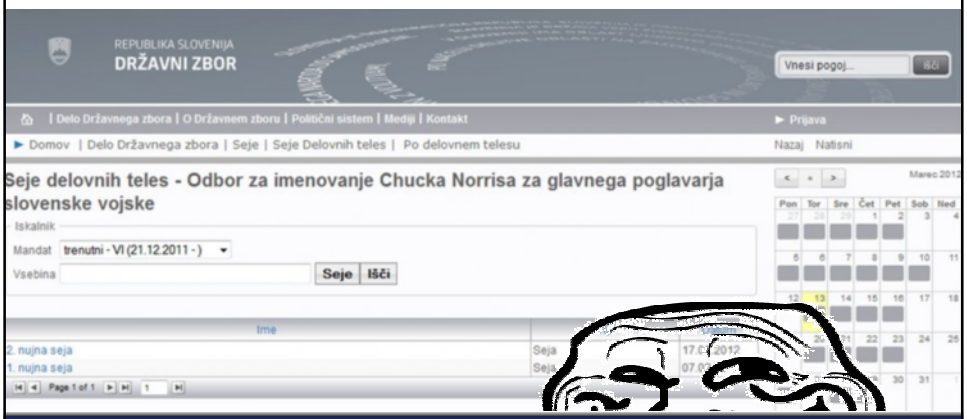
Težavnost vektorja napada	Razširjenost tveganja	Težavnost odkrivanja tveganja	Tehnični vpliv
Povprečna	Običajna	Povprečna	Resen

- HTTP je protokol, ki ne pozna stanja
  - Poverilnice se pošiljajo z vsakim zahtevkom
  - SSL bi se moral uporabljati pri vsaki akciji, ki zahteva avtentikacijo
- Napake pri upravljanju s sejo
  - Identifikator seje se uporablja za sledenje stanja. Ti identifikatorji so za napadalca vredni enako kot poverilnice
  - Identifikator seje je izpostavljen spletu
- Stranska vrata (mehanizmi za spremembo gesla, pozabljena gesla, skrivna vprašanja, trajanje seje, itd.)
- Posledice so ponavadi kraja računa ali seje

A3: Napaka pri avtentikaciji in upravljanju sej - zaščita

- Preglejte arhitekturo
  - Avtentikacija naj bo preprosta, centralizirana in predvsem standardizirana
  - Uporabite standardni sejni identifikator
  - Prepričajte se, da SSL ves čas ščiti tako poverilnice kot tudi sejne identifikatorje
- Preglejte implementacijo
  - Avtomatizacija odpade!
  - Preverite SSL certifikat
  - Dobro preglejte vse funkcije, povezane z avtentikacijo
  - Zagotovite, da odjava dejansko uniči sejo
- http://www.owasp.org/index.php/Authentication_Cheat_Sheet

A2: Podtikanje skript



A2: Podtikanje skript

Težavnost vektorja napada	Razširjenost tveganja	Težavnost odkrivanja tveganja	Tehnični vpliv
Povprečna	Prekomerno razširjena	Lahka	Zmeren

- Vrivanje zlonamerne vsebine/kode v spletne strani
- Najpogosteje srečamo kombinacijo HTML/javascript. Med ostalimi ranljivimi tehnologijami najdemo tudi:
  - Java, ActiveX, Flash, RSS, Atom, ...
- Prisoten pri 64% vseh spletnih aplikacij v letu 2010
- Mogoč v parametrih, piškotkih, podatkih v bazi oz. pri vseh podatkih, ki pridejo s strani uporabnika v surovi obliki.
- Napadalec lahko ukrade uporabniško sejo ali podatke, spremeni izgled spletne strani, preusmeri uporabnika na zlonamerno spletno stran, ...

A2: Podtikanje skript - zaščita

- Kako preprečiti?
  - Izognite se prikazovanju podatkov, ki pridejo od uporabnika
  - Validacija vnosa po "whitelist" principu
  - Kodiranje izhoda (ESAPI)
  - Če mora aplikacija podpirati HTML s strani uporabnika, uporabite preverjene knjižnice
- OWASP XSS Prevention Cheat Sheet
  - http://www.owasp.org/index.php/XSS_(Cross Site Scripting) Prevention Cheat Sheet

A2: Podtikanje skript - zaščita

- ESAPI nudi kodiranje vnosa za vseh 5 kontekstov, ki lahko vsebujejo podatke s strani uporabnika:
  - Vrivanje v HTML element (npr. XSS)
  - Vrivanje v HTML atribut (npr. <input name='person' type='TEXT' value='XSS'>)
  - Vrivanje v javascript
  - Vrivanje v HTML style
  - Vrivanje v URI atribute

A1: Vrivanje

A1: Vrivanje

Težavnost vektorja napada	Razširjenost tveganja	Težavnost odkrivanja tveganja	Tehnični vpliv
Lahka	Običajna	Povprečna	Resen

Banalen primer:


```
String query = "SELECT * FROM accounts WHERE custID='" + request.getParameter("id") +"'" ;
String query = "SELECT * FROM accounts WHERE custID='' OR 1=1 --" +"'";
```

Večina primerov ni banalnih!

A1: Vrivanje

- Pretenta aplikacijo, da vključi nenamerne ukaze v podatke, ki jih pošlje interpreterju.
- Interpreterji tekstovne nize obravnavajo kot ukaze.
- SQL, ukazna lupina operacijskega sistema, LDAP, Xpath, Hibernate, itd.
- SQL vrivanje je še vedno med najbolj razširjenimi problemi, čeprav je zelo lahko rešljiv.

A1: Vrivanje

- Ponavadi je vpliv zelo resen, saj med posledice SQL vrivanja spada:
  - Branje celotne baze
  - Pisanje v bazo
  - Dostop do računov
  - Izvajanje kode na operacijskem sistemu
  - Onesposobitev (denial of service)
  - Eskalacija privilegijev

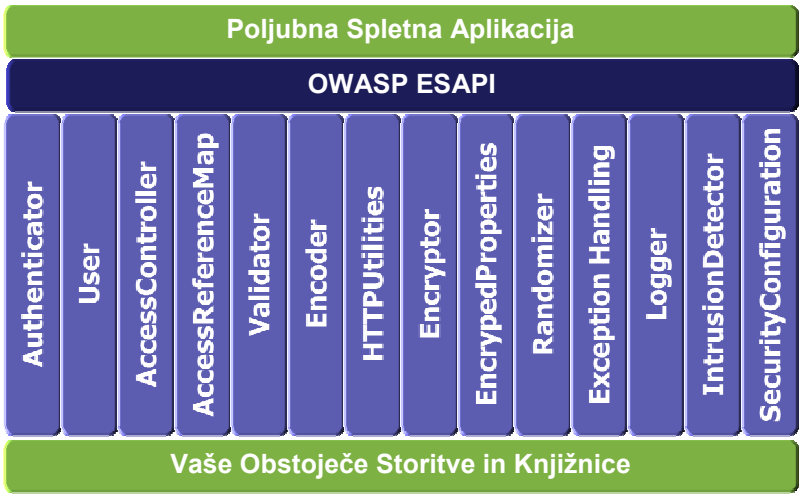
A1: Vrivanje - zaščita

- Kako preprečiti?
  - Popolno izogibanje interpreterju
  - Uporaba vmesnika, ki dovoljuje vezane spremenljivke (prepared statements, stored procedures)
    - Vezane spremenljivke omogočajo interpreterju, da razlikuje med kodo in podatki
  - Zakodiranje vseh vhodnih podatkov, preden se posredujejo interpreterju
    - Konsistentna uporaba "white list" validacije vnosa na vseh podatkih, ki prihajajo s strani uporabnika
    - Uporaba najmanjših možnih privilegijev na bazi, da zmanjšamo potencialne posledice napada
- Reference
  - http://www.owasp.org/index.php/SQL_Injection_Prevention_Cheat_Sheet

Kako se lotiti teh problemov?

- Razvijanje varne kode
  - Sledite dobrim praksam v "OWASP's Guide to Building Secure Web Applications" (http://www.owasp.org/index.php/Guide)
  - Uporabite "OWASP's Application Security Verification Standard" kot vodič za to, kaj aplikacija potrebuje, da bo varna (http://www.owasp.org/index.php/ASVS)
  - Uporabljajte standardne varnostne komponente, ki so primerne za vašo organizacijo
    - OWASP ESAPI kot osnova za standardne komponente
    - http://www.owasp.org/index.php/ESAPI
- Preglejte vaše aplikacije
  - Pregled aplikacije prepustite varnostnim strokovnjakom

OWASP (ESAPI)



http://www.owasp.org/index.php/ESAPI

Reference na OWASP Top 10

- PCI DSS
  - Payment Card Industry Data Security Standard je informacijsko varnostni standard za organizacije, ki delajo s podatki imetnikov plačilnih kartic.
  - Standard se sklicuje na OWASP Top 10.
- Med ostalimi organizacijami, ki se sklicujejo na OWASP Top 10, najdemo tudi:
  - MITRE
  - DISA
  - FTC
  - mnoge druge

Vprašanja?

https://www.owasp.org/index.php/Top_10_2010