Сигурност и администрација база података

Бранкица Сладоје
студент другог циклуса студија
Електротехнички факултет
Источно Сарајево, Босна и Херцеговина
brankicasladoje@gmail.com

Садржај—**Предмет овог рада је анализа основних приступа сигурности и саме администрације база података. У раду се разматрају механизми заштите база података са укљученим бројним сигурносним мјерама, које осигуравају заштиту података ускладиштеним у бази података, заштиту Web апликација које користе ресурсе базе података. Обрађени су основни концепти система и алата за управљање базама података, као и хардвера на којима се налази база података и припадних телекомуникационих линија комуникације са спољним свијетом, а све у сврху очувања повјерљивости, интегритета и доступности података који се налазе у базама података.**

Кључне ријечи-сигурност; база података; аутентификација; испитивање; VPD; Brute force; ауторизација; Content spoofing; SQL injection; контрола приступа; мрежне баријере; енкрипција; SSL

1. Увод

Сигурност база података је важна компонента посла администратора база података, из разлога што је сама количина осјетљивих информација смјештеним у тим базама велика и што о њој често зависи јако велики број људи. Без детаљног и свеобухватног плана и примјене безбједоносних мјера, интегритет база података ће се довести у питање. Сваки администратор мора познавати безбједоносне механизме у свом окружењу да би обезбједио да само овлашћени корисници приступају подацима у бази и врше њихова ажурирања.

Посљедице напада на базе података могу резултирати крађом идентита, бројева кредитних картица, финансијским губицима, губицима приватности, нарушавањем националне сигурности те другим бројним опасним посљедицама које су резултат приступа осјетљивим подацима. Како су се развијали DBMS-ови (*енг. Data Base Management System*), тако је постојала све очитија потреба за њиховом сигурношћу. Сваком новом верзијом (код свих произвођача) додаване су бројне сигурносне опције, као и сигурносне надоградње које би уклањале рањивости. Све већом употребом Интернета, приватно и на радном мјесту, јавио се императив осигуравања база података од приступа из спољашњег свијета. Ове мјере безбједности тежиле су да обезбједе тајност, непромјенљивост и доступност података овлашћеним лицима [1].

Кроз овај рад биће описана сигурност цјелокупног система база података, која се не односи само на сигурност саме базе података, већ и на:

- Сигурност базе података
- Сигурност DBMS-а
- Сигурност апликација које раде са базама података
- Сигурност оперативног система
- Сигурност web сервера
- Сигурност мрежног окружења

О свим наведеним сигурностима ће бити више описано у поглављима која слиједе.

2. Сигурност база података

Сигурност база података представља систем процеса и поступака којима се штити база података од нежељених активности. Под нежељеном активношћу се може сматрати злоупотреба легалног корисника, злонамјерни напад, или грешка изазвана непажњом, а коју је начинио легални корисник или процес [2].

Код заштите виталних информација на нивоу базе података, постоје три кључна елемента, тзв. велико тројство сигурности. Ти елементи (Сл. 1) су: [1]

- повјерљивост - обезбјеђење да је информација доступна само онима који имају овлашћени приступ дотичној информацији, тј. заштита података од неовлашћеног прегледавања,
- интегритет - заштита постојања, тачности и комплетности информације као и процесних метода, тј- заштита од недозвољеног приступа подацима,
- доступност - осигурање да ауторизовани корисници имају могућност приступа информацији и припадајућим средствима када је услуга потребна.

Слика 1. CIA тројство

Најосновнија метода заштите осјетљивих информација које се чувају у бази података је ограничење приступа подацима само одређеној групи корисника. На овај начин се осигурава повјерљивост података. Контрола приступа може се остварити на два начина: [1]

1. Аутентификацијом односно овјеравањем корисничког имена или лозинке.
2. Давањем посебних привилегија и права специфичним објектима и скуповима података.
3. Унутар базе података то су обичне табеле, прегледи, редови и колоне, а права која им се додјељују су читања, писања или обоје.

Аутентификација је процес идентификације и верификације идентитета. Ово је први корак у провођењу сигурносне политике у домену управљања интеракције корисника/процеса са информационом системом.
Прије утврђивања идентитета није могуће донијети одлуку о даљем поступању (Сл. 2) [3].

Слика 2. Важност процеса аутентификације

Методе које се могу користити при аутентификацији корисника су: [1]

- **елементи које корисник зна -** корисничко име и лозинка (најчешће се користи),
- **елемент које корисник посједује -** паметна картица *(енг. smart card)* или сертификат,
- **елемент који је атрибут самог корисника -** нпр. биометријска аутентификација помоћу отиска прста или узорка рожњаче.

2.1 Додјељивање и одузимање овлашћења

Администратор базе података контролише безбједност и дата овлашћења коришћењем језика за контролу података *(енг. Data Control Language, DCL)*. DCL наредбе се користе за контролу који корисници могу приступати којим објектима базе података или командама. Овим наредбама се правила сигурности спроводе у пракси.

Оне обухватају два типа наредби: [4]

- GRANT додјељује овлашћења кориснику базе података,
- REVOKE одузима додјељена овлашћења кориснику базе података.

Постоје различити типови овлашћења која се могу додјелити и одузети од корисника базе података. Сваки систем обезбјеђује одређене основне типове овлашћења, као што је могућност приступа подацима, креирање објеката базе података и извршавање функција система. Данашњи системи обично обезбјеђују руковање следећим овлашћењима: [4]

- ТАБЕЛЕ контрола приступа и измјене података у табелама.
- ОБЈЕКТИ БАЗЕ ПОДАТАКА контрола креирања нових објеката базе података и брисања постојећих.
- СИСТЕМ контрола коришћења одређених системских функција.
- ПРОГРАМ контрола креирања, измјена и коришћења програма у бази података.
- ЗАПАМЋЕНЕ ПРОЦЕДУРЕ контрола извршавања одређених функција и запамћених процедура.

Могу се одобрити следећа овлашћења за табеле и погледе: [4]

- SELECT омогућава кориснику да чита податке из табеле/погледа.
- INSERT омогућава кориснику да уписује слогове у табелу/поглед.
- UPDATE омогућава кориснику да ажурира табелу/поглед.
- DELETE омогућава кориснику да брише слогове из табеле/погледа.
- ALL омогућава кориснику да чита, уписује, ажурира и брише податке из табеле/погледа.

Нека овлашћења за табеле могу бити задата на нивоу колона, као што су овлашћења SELECT и UPDATE. То је пожељна опција која се користи када се одређеним корисницима дозвољава да ажурирају одређене колоне табеле [4].

Већина овлашћења за приступ подацима у радном окружењу треба да се контролише преко привилегија за програме или ускладиштене процедуре, прије него преко директних овлашћења за табеле [4].

3. СИГУРНОСТ DBMS-А

Данас се организације поуздају у DBMS-ове када се ради о спремању и осигуравању свих, па тако и оних најтајнијих и највриједнијих података. Због тога за тим системима расте занимање криминалне заједнице, а самим тим и потреба да се системи учине сигурнијима. Уграђивање сигурносних елемената директно у DBMS-ове и њихова исправна примјена једини су прави начин за уклањање рањивости.

Испитивање или контрола рада је функција система која се користи за праћење приступа бази података и активности корисника. Такође, испитивање се може користити за идентификацију корисника који приступа објектима у бази података те које акције се обављају и који подаци су промјењени [1].

Нажалост, испитивање не доноси одбрану од напада али придоноси рачунарској форензици након напада како би се лакше идентификовао пропуст. Процес записивања активности се обавља послије извршене акције. Записани трагови помажу у одржавању интегритета података тако што омогућавају детекцију пробоја безбједоносних мјера и детекцију упада у систем. Систем који се на овакав начин прати има ефекат одвраћања од таквих покушаја јер се онда починиоци могу лако открити [4].

У новије вријеме се покушавају имплементирати рјешења за надгледање базе у реалном времену, тако да се препознају узроци нелегитимног манипулисања базом података те да се о томе одмах обавијесте администратори базе података [1].

3.1 Виртуелне приватне базе података (VPD)

Развој информационих технологија, а посебно *BI (енг. Business Intelligence)* рјешења, доводи до ситуације да је заштита података у таквом окружењу веома изазован задатак. Осим интерне заштите података пословних субјеката, пред информатичаре се стављају и захтјеви за заштиту од неовлашћеног приступа осјетљивим подацима. Једно од рјешења које *Oracle* нуди је VPD *(енг. Virtual private network)* [5].

Oracle VPD је сигурносни оквир који је први пут имплементиран у верзији базе 8i под називом FGAC *(енг. Fine Grained Access Control)*. Основна функција VPD-а је постизање нивоа сигурности на нивоу реда - RLS *(енг. Row Level Security)* [6].
Могућност заштите на нивоу колона у VPD уведена је у издању базе 10g, а своди се на скривање редова са осјетљивим колонама. Постоји и могућност приказивања свих редова са скривањем осјетљивих колона. Скривање колона може се користити само у SELECT наредбама. VPD правила на нивоу колона не могу се примјењивати над синонимима [5]. VPD механизам темељи се на динамичкој модификацији наредби за читање (SELECT) или ажурирање (INSERT, UPDATE, DELETE) података табела, погледа или синонима. Сервер базе података аутоматски модификује наредбу коју је корисник послао бази, тако да у WHERE клаузулу додаје услове повезујући их AND оператором. Додатне услове враћа функција која је имплементирана у оквиру сигурносних правила VPD-а. Уколико је истовремено активно више VPD сигурносних функција над истим објектом, резултати се повезују AND оператором у WHERE клаузули. Дакле, постоји могућност да једна сигурносна политика условљује искључење друге и резултат је тада празан скуп података [6].

4. СИГУРНОСТ АПЛИКАЦИЈА КОЈЕ РАДЕ СА БАЗАМА ПОДАТАКА

Данас многе *web* апликације захтијевају унос наших личних података, регистрацију, а затим и пријављивање на систем да би се приступило и користила апликација. Ти подаци су најчешће *mail* адреса, шифре, бројеви рачуна, адреса, телефона, а то може довести до нарушавања сигурности корисника апликације. Како се све ове активности обављају на Интернету, то захтијева да се користе базе података, велики број сервера који су у сталној интеракцији како међусобно тако и са базом података. Ово доноси проблеме у заштити *web* апликација и клијента [7].

Како би се апликације и корисници заштитили потребно је истражити *web* апликације, идентификовати критичне тачке, а затим након стварања класа потенцијалних напада понудити и рјешење, тј. заштиту, а да се притом не смањи квалитет и количина услуге.
Напади се могу разврстати у следеће класе: [8]

- Напади везани за аутентификацију
- Напади везани за ауторизацију
- Напади на клијентску страну
- Напади везани за извршавање наредби
- Откривање повјерљивих информација
- Логички напади

4.1 Напади везани за аутентификацију

Brute force је нападачка техника којом се може заобићи и нарушити аутентификациони процес. Користи методу погађања и промашаја како би открио корисничко име, шифру, криптографски кључ, број кредитне картице итд. Ова врста напада је врло честа и успјешна, а сам напад може трајати од неколико минута до неколико година.

Најбоља заштита од ове врсте напада јесте да један исти клијент на различитим системима користи различите шифре, при чему се препоручује што већа дужина шифара искомбинована од знакова, слова и бројева, што шифри даје још већу сигурност [7]. Процес аутентификације одређене *web* апликације захтијева од корисника памћење лозинке. Уколико дође до тога да корисник заборави своју лозинку, он приступа процесу за обнову лозинке. Процес углавном користи принцип „тајног питања“ *(енг. secret question)*, које корисник дефинише приликом регистрације кључа. Корисник може обновити лозинку одговором на ово питање [7].

Ови системи се могу преварити коришћењем *Brute force* напада, а и тајним питањима који се могу наслутити. На примјер, уколико нападач захтијева обнову лозинке са одређеним корисничким именом и тајно питање је „У ком граду си рођен?“, може се са великом вјероватноћом погодити тачан одговор, коришћењем *Brute force* алата, у чијем рјечнику су записани градови.

Додатна заштита од оваквих напада јесте избор тајног питања таквог да није једноставан одговор на њега, тј. да је одговор што личнији, познат малом броју људи [7].

4.2 Напади везани за ауторизацију

Ауторизација је процес којим се утврђује ниво привилегија корисника, тј. да ли одређени корисник има потребна допуштења за извођење одређене радње.

Када се корисницима или апликацијама додјели већи ниво сигурности него што им је потребан често долази до злоупотребе датих привилегија. Овај пропуст углавном је узрокован администраторовим недостатком времена да фино гранулира задатке које поједини корисник мора и може обављати, те се врло често догађа да корисник добије велика овлашћења над базом података над којом му је потребно прегледати свега пар редова [8].

4.3 Напади на клијентску страну

Ови напади су усмјерени на злоупотребу и експлоатисање корисника апликација. Када корисник приступи некој апликацији он очекује да добије исправне информације и садржај, наравно без бојазни да ће се десити напади за вријеме приступа. Много нападачких техника могу угрозити однос између апликације и корисника, а једна од најпознатијих је убацивање непостојећег садржаја *(енг. Content spoofing)* [7].

Убацивање непостојећег садржаја је врста напада којом нападач жели увјерити корисника да је садржај исправан и да не потиче са неког другог извора ван апликације коју корисник користи. Напад је обично усмјерен према *web* апликацијама које динамички генеришу URL-ове према свом HTML садржају.

На примјер, корисник жели да приступи некој *web* апликацији путем линка. Измјеном садржаја HTML документа може се промјенити извор параметра у жељени облик и уметнути садржај, а затим прослиједити овако скројени линк кориснику путем *e-mail*-а или слањем линка на форум. Корисник када посјећује страницу види домен у *web* претраживачу и увјерен је да је садржај са његове жељене локације, међутим тај садржај потиче са нападачевог извора.

Content spoofing се најчешће користи за стварање лажних *web* страница, како би се на ралативно једноставан начин украо корисников идентитет. Због тога корисник мора бити обазрив и провјеравати URL за сумњиве садржаје [8].

4.4 Напади везани за извршавање наредби

Напади везани за извршавање наредби подразумјевају нападе који су посебно дизајнирани, како би извршили штетне наредбе над *web* апликацијом. Напад уметањем SQL кода (енгл. SQL *injection*) сматра се једним од 10 најчешћих и најопаснијих напада на *web* странице. Овај облик напада искориштава рањивости на слоју базе података, а могућ је због недовољних провјера корисничких улазних података који се користе при преузимању података из базе. Рањиве могу постати све *web* странице које користе SQL базу података за свој рад. Нападе уметањем SQL кода релативно је једноставно извести и довољно је основно знање о упитима за рад с базом података [9].

Уметање помоћу поља за унос података је најпознатији облик напада уметањем SQL кода. Ако на *web* страници нису уведене неке основне провјере података које корисник уноси, напад може извести и мање стручан нападач.

Најједноставнији примјер овог облика напада је заобилажење легитимне пријаве на систем. Приликом пријаве, корисник мора уписати своје корисничко име и лозинку у два поља за упис података. Програм затим купи низове знакова које је корисник уписао и смјешта их у двије варијабле, нпр. *user* и *pass*. Добијене податке програм користи за стварање SQL упита како би провјерио постоји ли у бази података корисник с корисничким именом user и лозинком pass. Нека база података садржи табелу имена Korisnici која има атрибуте (колоне у табели) KorisnickoIme и Lozinka [9].

SQL упит се слаже на следећи начин:

```
upit = “SELECT KorisnickoIme FROM Korisnici
        WHERE KorisnickoIme = '“ + user + “'
          AND Lozinka = '“ + pass + “'“
```

Ако се корисник жели пријавити на систем, он ће уписати своје корисничко име и лозинку у одговарајућа поља. Ако корисник као корисничко име и лозинку упише, нпр, *Brankica* и *password*, SQL упит ће гласити:

```
SELECT KorisnickoIme FROM Korisnici
WHERE KorisnickoIme = 'Brankica'
AND Lozinka = 'password'
```

Упит ће резултирати претрагом табеле *KorisnickoIme* у бази података за корисником *Brankica* чија је лозинка *password*.

Међутим, ако не постоји провјера података које је корисник унио, нападач може искористити за унос за извођење било какве SQL наредбе. Нпр, ако корисник у поља унесе следећи низ знакова:

```
' OR '1'='1
```

Ствара се следећи SQL упит:

```
SELECT KorisnickoIme FROM Korisnici
WHERE KorisnickoIme = '' OR '1'='1'
AND Lozinka = '' OR '1'='1'
```

Оно што се догађа је да база података више не упоређује податке у табели с корисничким именом које је корисник унио, него провјерава истинитост тврдње '1'='1'. Како је тврдња увијек истинита, и цијели WHERE дио SQL упита ће бити истинит. То резултира враћањем првог реда у табели *Korisnici* чиме се нападач успјешно пријавио на систем као корисник који је први наведен у табели. Резултат овог напада уметањем SQL кода је могућа пријава нападача као неки други, легитимни корисник.

4.5 Откривање повјерљивих информација

Откривањем повјерљивих информација нападач покушава добити одређене системске податке о *web* апликацији. Системске информације укључују дистрибуцију програмске подршке, верзију или локације привремених *(temp)* или *backup* датотека.

Расипање информација је сигурносни пропуст којима *web* апликације откривају осјетљиве податке, као што су програмерски коментари и детаљне поруке о грешкама, које могу бити од користи нападачу. Овакви осјетљиви подаци се могу наћи у HTML коментарима, порукама о грешци, изворном коду или могу једноставно бити јавно доступни свима.

Коментари у коду и детаљне поруке о грешци представљају расипање информација којима се могу нападачу дати информације о структури мапа, структури SQL изјава и имена кључних процеса коришћених од стране *web* апликације. У развоју апликације, нпр. док програмери врше *debugging* они остављају HTML коментаре и тиме могу нарушити безбједност апликације, поготову уколико су ти коментари нека корисничка имена и шифре намјењени за тестирање апликације. Број кредитне картице, шифре и други лични подаци, као што је ЈМБГ, потребно је додатно заштитити од расипања примјеном енкрипције и додатне контроле приступа [7].

4.6 Логички напади

Логички напади су нападачке технике намјењене нарушавању или злоупотреби логичког тока *web* апликације. Под логичким током или апликационом логиком се подразумјева процедурални ток, који се користи у *web* апликацији како би се извела нека акција. Најпознатији примјери апликационе логике су: процес за обнову лозинке, регистрација рачуна или on-line куповина. *Web* апликација захтијева од корисника да се тачно и прецизно изводи одређени процес у корацима како би завршио поједину радњу. Нападачева намјера је управо заобићи или злоупотрјебити овај процес како би наштетио *web* апликацији и њеним корисницима [8].

5. СИГУРНОСТ ОПЕРАТИВНОГ СИСТЕМА

Када се говори о заштити на нивоу оперативног система, улази се у веома сложено и обимно подручје које на неки начин дотиче све слојеве оперативног система. Оперативни систем управља разним објектима који могу бити хардверски (процесор, меморија, дискови) и софтверски (датотека, програм, семафор). Заштита на нивоу оперативног система обухвата и везу оперативни систем-апликације, као и однос према мрежној архитектури тј. везама са другим системима [10].

Кључна област сигурности рачунара обухвата заштиту меморије. Ово је суштински важно у системима у којима је истовремено покренуто више процеса. Виртуелна меморија је обично опремљена одговарајућим механизмом за обављање овог задатка [11]. Једна од важнијих сигурносних техника је и контрола приступа. Сврха контроле приступа је да се обезбједи да само овлашћени корисници имају приступ до одређених система и њихових појединачних ресурса и да се приступ

и измјена одређених дијелова података ограничени овлашћеним појединцима и програмима. Чување информација у безбједном стању значајно је како за кориснике тако и за систем администраторе. Заштитом датотека од неовлашћеног коришћења, корисници штите интегритет свог рада [11].

Нарастујућа врста пријетње је она коју намећу вируси и слични софтверски механизми. Ове пријетње искоришћавају слабости у системском софтверу било да би задобили неовлашћени приступ до информација или да би деградирали системске услуге [11].

Технологија која се све више примјењује у војним и комерцијалним условима су системи од повјерења. Системи од повјерења нуде начин за управљање приступом подацима на основу тога ко је чему овлашћен да приступа. Кључна ствар је да је овај систем пројектован и имплементиран тако да корисници имају потпуно повјерење да ће одређени систем спроводити дату политику сигурности [11].

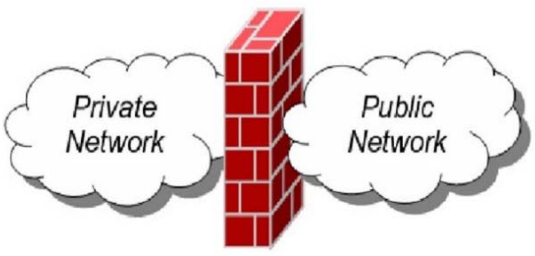
6. Сигурност мреже и WEB сервера

Када се говори о заштити на нивоу мрежне инфраструктуре, обично се мисли на следеће основне елементе: примјену мрежних баријера (*енг. firewalls*), блокирање непотребних портова (прикључака), шифровање путање, изоловање путање помоћу рутера и комутатора или помоћу посебне инфраструктуре [10].Доминантни приступ у мрежној сигурности је базиран на томе да се мрежа покуша сегментирати на више мањих дијелова, постављајући на рубним дијеловима подсегмената мрежна баријере (*енг. firewalls)* које примјењују сигурносна правила и политике како би онемогућили упадање из спољашњих мрежа [1].

Заштита на нивоу *web* сервера може да обухвати, на примјер, сљедеће елементе: софтверску заштиту апликације (рецимо, заштиту од прекорачења бафера), изоловање битних апликација на намјенским серверима и умреженим рачунарима, примјену специфичних протокола (на примјер, криптографски заштићеног протокола SSH умјесто незаштићеног протокола *Telnet*), [10].

6.1 Firewalls

Мрежне баријере су еволуирале заједно с развојем интернета те се користе за постављање контролних тачака безбједности на границама приватних мрежа. У контролним тачкама мрежна баријера испитује све пакете који пролазе између приватне мреже и интернета. У зависности од тога да ли пакети задовољавају правила дефинисана листама за контролу приступа, мрежна баријера ће дозволити или забранити проток тог пакета. Мрежна баријера је филтер на релацији локална мрежа - Интернет (Сл. 3), [11].



Слика 3. Мрежна баријера

Код заштита база података углавном се користе следеће мрежне баријере:

- Пакетно филтриране мрежне баријере - надгледају изворне и одредишне IP адресе било које везе и провјеравају их према уграђеном сету правила како би одлучили да ли би се веза требала дозволити или не. Не провјеравају садржај па их је лако заобићи.
- Апликативни посредници *(енг. application proxy)* - замјењују клијента на серверској страни односно сервера на клијентској страни. Дозвољавају и прекидају те двије везе по потреби, а сав промет пролази кроз њих.
- Инспекцијске мрежне баријере - процесори пакета који провјеравају читаве сесије. Провјеравају постоје ли протоколи те постоје ли злонамјерно обликовани пакети. Провјеравају конзистентност таблица стања промета и провјеравају стање TCP везе, адресне транслације и друге. Ове мрежне баријере подржавају кориштење VPN веза, [1].

6.2 Енкрипција

Елемент повјерљивости података најбоље се осигурава преко метода криптовања *(енг. encryption)*. Криптовање или енкрипција би се требала схватити као последња линија одбране кад све остале сигурносне мјере закажу. Енкрипција је поступак помоћу кога се изворни текст трансформише у шифрован текст. Енкрипција се користи да би се обезбједило да ниједан корисник, осим корисника коме је порука намјењена, не може да сазна садржај поруке. Ако неовлашћени корисници дођу у посјед криптованог текста и виде његов садржај не могу прочитати изворни текст [1].

6.3 SSL протокол

Размјена повјерљивих података са другим рачунарима на Интернету понекад је нужност. Када то чинимо, податке шаљемо кроз јавну мрежу и ризикујемо могућност да их неко на њиховом путу покупи. Сва размјена повјерљивих података требала би се обављати SSL *(енг. Secure Sockets Layer),* или неким другим протоколом који подржава енркипцију података у промету. SSL протокол је тренутно најчешће коришћен метод за обављање сигурних трансакција на мрежи. Првобитно је дизајниран 1995. године од стране корпорације *Netscape Communications*, тада главног произвођача *Web* претраживача.

SSL је широко прихваћен протокол чије имплементације обухватају кључне елементе сигурности размјене података: повјерљивост, интегритет, аутентичност и непоречивост. Ове карактеристике осигуране су коришћењем асиметричне енкрипције. Сертификати потписани од стране организације од повјерења гарантују нам идентитет сервера са којим комуницирамо (Сл. 4), а асиметрични алгоритам рјешава проблем размјене криптографског кључа [12].

Слика 4. Верификација сертификата од стране CA (*енг. Certification Authority*)

SSL омогућава административни приступ серверима уз коришћење сертификата умјесто лозинки. Коришћење сертификата у пракси се показује као боља метода у поређењу са лозинкама, јер је за приступ тајном кључу на локалном рачунару често потребно више труда него за откривање лозинке [12].

7. Закључак

Сигурност база података неисцрпна је и увијек актуелна тема. Базе података садрже важне информације о пословању. Подаци у њима представљају слику тренутног стања фирме и пословних процеса. Управо због тога врло је важно базе података заштитити и управљати њима на прави начин. Важно је разумјети да сигурност као готов производ не постоји. Сигурност је процес са којим настојимо да очувамо ресурсе у оном облику и на онај начин на који смо ми задовољни у нашем пословању. Гледано с аспекта базе то је заштита података, од било којег ауторизованог и неауторизованог приступа. Онај ко има приступ може податке да малициозно модификује, промјени, искористи као и онај ко нема исти приступ. Постоје начини размишљања и техничка правила која у сигурности увијек вриједе, али пријетње и доступна рјешења стално се мјењају. Због тога не можемо рећи да смо постигли потпуну сигурност и да више не морамо да бринемо о томе, јер сигурност никако није крајње стање него процес.

У овоме раду је обухваћено неколико истраживачких тема које су за посљедицу имале будући допринос који се тиче препорука везаних за сигурност база података и то:

- корисницима је потребно додјељивати само неопходна овлашћења,
- посебну пажњу потребно је посветити управљању корисничким налозима и лозинкама,
- исправно примјењене методе надзора и периодичке анализе могу увелико помоћи приликом откривања напада, а тиме и олакшати проналажење рањивости и њихово уклањање,
- потребно ја анализирати проблеме везане за *Web* апликације, као и начине за њихово спречавање и одбрану,
- коришћење енкрипције злонамјерним корисницима отежава приступ осјетљивим информацијама, како корисничким лозинкама, тако и свим осталим подацима смјештеним у бази,
- постављање сервера с базом података у унутрашњу мрежу чини га далеко сигурнијим.

Захвалница

Захваљујем се свом професору и ментору доц. др Срђану Ногу, на савјетима и помоћи приликом израде овог рада.

Литература

[1] Центар информацијске сигурности, „*Заштита база података*“, август 2012 http://www.cis.hr/files/dokumenti/CIS-DOC-2012-08-059.pdf приступљено (27.09.2014).
[2] Дамјановић С., „*Криптографска заштита база података*“, *Магистарски рад*, Универзитет Сингидунум - Београд , мај 2010.
[3] Бруно Мандић, „*Методологија и механизми реализације сигурносне политике код (не)комерцијалних RDBMS*“, *Дипломски рад*, Свеучилиште у Загребу, Факултет електротехнике и рачунарства, септембар 2012.
[4] Craig S. Mullins, „*Администрација база података*“, Компјутер Библиотека, Чачак, 2003.
[5] Наташа Дворшак, „*VPD у примјени*“, http://mypaper.mediazona.hr/Publish/Repository/hroug/0e24f5d4-8f12-4501-a3b4-430d4c7d7ad2/1044.pdf приступљено (10.10.2014.год).
[6] Oracle, "*Oracle® Database Security Guide 10g Release 2 (10.2)* " http://docs.oracle.com/cd/B19306_01/network.102/b14266.pdf приступљено (10.10.2014.год).
[7] Кристијан Ристић, „*Сигурност Web апликација*“, Електронски факултет Ниш, 2012.
[8] Web Application Security Consortium, *"Threat Classification"* http://projects.webappsec.org/f/WASC-TC-v1_0.pdf приступљено (27.10.2014).
[9] Центар информацијске сигурности, „*Напади уметањем SQL кода*“, јун 2011, http://www.cis.hr/files/dokumenti/CIS-DOC-2011-09-025.pdf приступљено (1.11.2014.год).
[10] Драган Плескоњић, Немања Мачек, Борислав Ђорђевић, Марко Царић, „*Сигурност рачунарских система и мрежа*“, Микро књига.
[11] William Stallings, „*Оперативни системи: Принципи унутрашње организације и дизајна*“, превод петог издања, Рачунарски факултет, Београд 2007.
[12] Croatian Academic and Research Network, „*Сигурније пословање на Интернету*“, http://www.cert.hr/sites/default/files/sigurnije_poslovanje_na_internetu.pdf приступљено (11.11.2014.год).

ABSTRACT

The subject of this paper is to analyse basic security approaches and database administration. In this paper mechanisms of protecting databases with a lot secure measures are taken into consideration. Moreover, these security measures ensure: protection of data stored in databases and web application protection which uses database resource. In addition to this, this paper also contains basic concept of systems and tools for database administration, hardware on which database is stored and related telecommunication lines of communication with external world. The aim of all these mechanisms is to save privacy, integrity and availability of information stored in databases.

DATABASE SECURITY AND ADMINISTRATION

Brankica Sladoje