

## SISTEM ZA UPRAVLJANJE BEZBIJEDNOŠĆU RAČUNARSKIH RESURSA ASSET MANAGEMENT CYBER SECURITY SYSTEM

Nevena Lazarević, *Fakultet tehničkih nauka, Novi Sad*

### Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

**Kratak sadržaj** – U radu je izložena implementacija sistema za upravljanje bezbijednošću računarskih resursa organizacije. Detaljno su objašnjeni model i implementacija sistema. Dodatno, ukazano je na svrhu sistema i navedene su funkcionalnosti koje pruža sistem.

**Ključne reči:** Resurs, Upravljanje, Bezbijednost, Generisanje izvještaja

**Abstract** – The paper presents the implementation of Asset management cyber security system. Detailed description about its model and implementation is given. Additionally, purpose and functionalities of system are pointed out.

**Keywords:** ITAM, Asset, Cyber security, Report generation

### 1. UVOD

Kako bi se obezbijedilo uspješno poslovanje u organizacijama velikih razmjera potrebno je zadovoljiti ogromne tehnološke zahtjeve. Sa povećanjem resursa unutar organizacija, njihovo praćenje postaje sve složenije.

Pod resursom (engl. *Asset*) se može smatrati sve što je vrijedno za organizaciju. Resurs može biti u obliku fizičkog uređaja, virtuelne mašine, podatka ili pak softverskog sistema koji se koristi u sklopu same organizacije. Dakle, sve što ima vrijednost i podržava poslovanje organizacije može se nazvati resursom.

Ogroman zahtjev je obezbijediti sigurnost organizacije. Zbog velike konkurencije na tržištu, postoje stalne bezbijedonosne prijetnje koje ugrožavaju uspješno funkcionisanje organizacije. Da bi organizacija uspjela opstati uprkos konstantnim prijetnjama, potrebno je osigurati sve resurse da bi sama organizacija bila bezbijedna.

Bezbijednost pruža mnoge prednosti kao što su povjerenje kupaca, ugled i značajna tržišna nadmoć. Da bi se ispunili navedeni zahtjevi, od velike je važnosti koristiti centralizovan sistem za kontrolu, čiji zadatak jeste da koristi obezbjeđuje i upravlja svim računarskim resursima određene organizacije.

U radu je izložena implementacija AMCSS (Asset Management Cyber Security System) [1] sistema koji je izgrađen sa ciljem da pruži platformu za upravljanje i osiguravanje računarskih resursa u organizaciji.

Cilj rada jeste da ukaže na značaj obezbjeđivanja i vođenja evidencije o računarskim resursima koje posjeduje organizacija. Korišćenjem ovakvog softverskog rješenja, u ranoj fazi se mogu detektovati neočekivana ponašanja resursa u organizaciji. Dodatno, moguće je detektovati i prijetnje koje bi mogle narušiti bezbijednost organizacije ili neki drugi način ugroziti njeno poslovanje.

### 2. UPRAVLJANJE BEZBIJEDNOŠĆU RAČUNARSKIH RESURSA

Računarski resursi obuhvataju različite tehnološke resurse kako hardverske, softverske, virtuelne tako i informacione.

S tim u vezi, sistem za upravljanje računarskim resursima obuhvata četiri ključne cjeline [2]:

- Sistem za upravljanje hardverskim resursima,
- Sistem za upravljanje softverskim resursima,
- Sistem za upravljanje virtuelnim resursima i
- Sistem za upravljanje informacionim resursima.

Sistem za upravljanje hardverskim resursima (engl. *Hardware Asset Management*) ima za cilj da upravlja i obezbjeđuje sve fizičke uređaje koje organizacija posjeduje. Hardverski resursi obuhvataju servere, radne stanice, prenosne računare, mrežne uređaje, mobilne uređaje i slično [3]. Sistem za upravljanje softverskim resursima (engl. *Software Asset Management*) ima zadatak da upravlja i osigurava sve softverske resurse.

Softverskim resursima smatraju se aplikacije, softverska rješenja, biblioteke i slično [4]. Sistem za upravljanje virtuelnim resursima (engl. *Virtual Asset Management*) upravlja i obezbjeđuje sve virtuelne mašine koje organizacija posjeduje [5].

Sistem za upravljanje informacionim resursima (engl. *Information Asset Management*) ima za cilj da upravlja i osigurava sve podatke koje su od značaja organizaciji. Informacioni resursi obuhvataju različite ugovore, standarde, reklame, datoteke i slično [6].

Da bi se bezbijedno upravljalo svim računarskim resursima u organizaciji, potrebno je pratiti sve faze kroz koje resurs prolazi u toku svog životnog ciklusa. Dodatno, bitno je shvatiti koji je cilj svake faze što će biti detaljno objašnjeno u narednom odjeljku [7].

### NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Goran Sladić, vanr. prof.

## 2.1. Životni ciklus resursa

Glavni korak u upravljanju resursima je razumijevanje njihovog životnog ciklusa [8]. Ukoliko se resursom upravlja pravilno, to jeste u skladu sa fazom u kojoj se nalazi, mogućnosti resursa će biti iskorišćene maksimalno. U skladu sa time, ukoliko organizacija iskoristi sve svoje resurse na najbolji mogući način, neosporna je činjenica da će profitirati u svakom segmentu svog poslovanja. Resurs prolazi kroz četiri ključne faze u toku svog životnog ciklusa, i to [9]:

1. Nabavka,
2. Registracija,
3. Aktivni vijek i
4. Odlaganje.

Nabavka predstavlja prvu fazu životnog ciklusa određenog resursa i ona obuhvata planiranje nabavke i samu nabavku resursa [9]. Planiranje nabavke se zasniva na utvrđivanju potreba za novom opremom u organizaciji. Utvrđivanje potrebe za resursima zasniva se na procjeni efikasnosti postojećih sredstava i utvrđivanje njihovih performansi i mogućnosti da ispune dodijeljene operativne zadatke. Drugi korak predstavlja proces nabavke novih resursa. Potrebno je donijeti nabolju odluku za kupovinu novog resursa. Odluka mora biti takva da napravi balans između traženih zahtjeva i dozvoljenih troškova.

Glavni cilj u ovom koraku jeste nabaviti resurs čije ulaganje će se isplatiti organizaciji u određenom vremenskom roku [9].

Registracija predstavlja drugu fazu životnog ciklusa resursa. Nakon što je resurs nabavljen, potrebno je registrovati ga i otpočeti sa njegovim korišćenjem. Da bi se resurs uspješno registrovao, resurs je potrebno prethodno: konfigurirati, instalirati u skladu sa definisanim specifikacijama, povezati na mrežu organizacije na siguran i pouzdan način, definisati fizičku lokaciju resursa, dodijeliti vlasniku koji će biti odgovoran za njega i obučiti radnike (ukoliko se radi o složenom resursu i postoji potreba za obučavanjem) [10].

Nakon što je resurs u eksploataciji, započinje njegov aktivni životni vijek i tada ispunjava sve dodijeljene operativne zadatke. Resurs se nalazi u fazi aktivnog vijeka sve dok se ne nađe u takvom stanju da je neophodno njegovo uklanjanje iz organizacije jer je postao neefikasan [10]. Tokom ove faze, fokus treba biti na odgovarajućem održavanju, praćenju, upravljanju i potencijalnom poboljšanju resursa. Cilj je smanjivanje poteškoća u prilagođavanju na novonastale operativne zahtjeve [10].

Kada završi aktivni vijek resursa, on se smatra viškom, neefikasnim ili neadekvatnim na bilo koji drugi način. Odlaganje treba posmatrati kao prvi korak u povećavanju vrijednosti i profita organizacije.

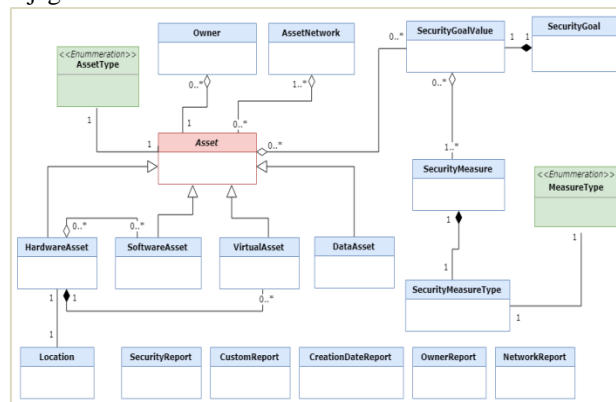
Nakon završenog odlaganja određenog resursa, započinje životni ciklus novog resursa koji će ga zamijeniti. Kao što je ranije navedeno, životni ciklus započinje nabavkom, što znači da nabavka započinje odmah nakon završenog odlaganja [9].

Kada je riječ o hardverskim resursima, prije njihovog odlaganja neophodno je nepovratno ukloniti sve podatke koji se nalaze na njima. Dodatno, recikliranje tehnoloških uređaja, odnosno hardverskih resursa, je takođe stavka o kojoj treba voditi računa jer predstavlja odličnu osnovu za održivo upravljanje računarskim resursima, a takođe utiče i na ugled same organizacije.

U skladu sa time, potrebno je voditi računa o tome da li postoje određene zakonske regulative koje je potrebno ispoštovati prilikom odlaganja.

## 3. MODEL SISTEMA

U ovom odeljku biće opisan model sistema za upravljanje bezbjednošću računarskih resursa u organizaciji. Za efikasno upravljanje bezbednosnim karakteristikama, potrebno je pratiti sve računarske resurse koje organizacije posjeduju, od fizičkih uređaja, preko softverskih aplikacija, do virtuelnih mašina i ostalih. Takođe, svaki resurs neophodno je osigurati i pratiti tokom njegovog životnog ciklusa. Slika 1 prikazuje dijagram klasa sistema.



Slika 1 - Dijagram klasa sistema

Model sistema podijeljen je u tri cjeline (modula), i to:

1. Modul resursa,
2. Bezbijedonosni modul i
3. Modul za izvještavanje.

### 3.1. Modul resursa

Osnovna komponenta sistema je resurs na kome se zasniva kompletna struktura sistema. U modelu resursa postoji samo jedan učesnik, a to je administrator organizacije. On ima pravo da registruje novi ili ukloni postojeći resurs. Prilikom registracije novog resursa (klasa *Asset*), administrator ima obavezu da resursu dodijeli vlasnika (klasa *Owner*). Resursi mogu biti grupisani. Grupa resursa, čiji jedini uslov jeste da sadrži minimalno dva resursa, se naziva mrežom i predstavljena je klasom *AssetNetwork*. Mreža može sadržati resurse istog tipa, svrhe, funkcije i slično.

Da bi se zadovoljila činjenica da resurs može biti sve što organizacija posjeduje, od velike je važnosti definisati različite kategorije kojima resurs može pripadati. U skladu sa time, resurs može biti klasifikovan kao:

1. Hardverski resurs,
2. Softverski resurs,
3. Virtuelni resurs ili
4. Informacioni resurs.

### 3.2. Bezbijedonosni modul

Resurs se smatra ranjivim ukoliko ima zadat barem jedan sigurnosni cilj. U bezbijedonosnom modelu postoji jedan učesnik, a to je vlasnik resursa. Pošto je vlasnik odgovoran za dodijeljene resurse, dužan je da prati bezbijednost resursa. Da bi resurs bio siguran, vlasnik ima obavezu da postavi sigurnosne ciljeve resursu. Ovaj korak takođe

podrazumijeva definisanje sigurnosnih mehanizama. Bez-bijedonosni model se može podijeliti u dvije grupe:

1. Sigurnosni ciljevi i
2. Sigurnosna kontrola.

Sigurnosni ciljevi (klasa *SecurityGoal*) predstavljaju ciljeve koji se moraju ispuniti tokom životnog ciklusa određenog resursa da bi resurs bio siguran i zaštićen. Sigurnosna kontrola (klasa *SecurityMeasure*) obuhvata bezbijedonosne mjere koje se vezuju za sigurnosne ciljeve. Da bi organizacija bila sigurna, obje grupe moraju biti na pravilan način implementirane. Dodatno, grupe su međuzavisne. Prvo se definišu ciljevi koje je potrebno ispuniti nad resursom, a nakon toga i mjere koje će se koristiti da bi se ispunili zadati ciljevi.

### 3.3. Modul za izvještavanje

Sa povećanim brojem resursa unutar organizacije, izazov je voditi evidenciju o svakom od njih. Zadatak postaje još teži ukoliko je potrebno pratiti resurse koji pripadaju određenim mrežama ili resurse određenih vlasnika. Da bi se ovaj problem riješio, sistem nudi mogućnost generisanja izvještaja. Generisanje izvještaja je funkcionalnost putem koje sistem vraća resurse, koji zadovoljavaju definisane kriterijume, na brz i efikasan način.

Krajnji korisnik sistema ima na raspolaganju pet šablona za generisanje izvještaja, i to:

1. Sigurnosni šablon (klasa *SecurityReport*) – koristi se kada se žele pratiti ranjivi resursi,
2. Šablon opšte namjene (klasa *CustomReport*)
3. Šablon vremena kreiranja (klasa *CreationDateReport*) – koristi se kada se žele pratiti resursi registrovani u određenom vremenskom intervalu,
4. Šablon vlasništva (klasa *OwnerReport*) – koristi se kada se žele pratiti resursi koji su dodijeljeni određenim vlasnicima i
5. Mrežni šablon (klasa *NetworkReport*) – koristi se kada se žele pratiti resursi koji pripadaju određenim mrežama resursa.

## 4. IMPLEMENTACIJA SISTEMA

Prilikom implementacije sistema, cilj je napraviti jednostavnu i intuitivnu aplikaciju koju će moći koristiti svi zaposleni u organizaciji, od administratora, preko menadžera, do operatera i ostalih zaposlenih.

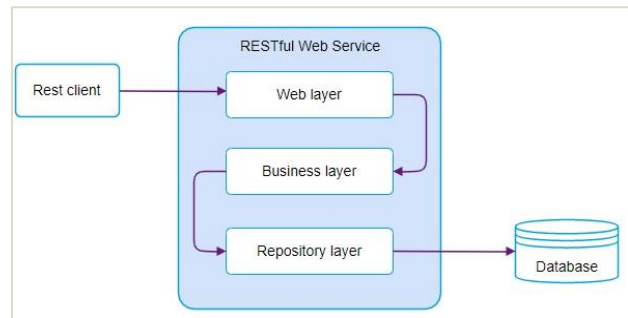
### 4.1. Arhitektura sistema

Sistem je implementiran kao standardna veb aplikacija.

Sistem se sastoji iz tri komponente (Slika 2):

1. *Klijentska aplikacija,*
2. *Serverska aplikacija i*
3. *Sistem za upravljanje bazom podataka.*

Prva komponenta sistema jeste klijentska aplikacija. Predstavlja JavaScript [11] baziranu klijentsku aplikaciju i pisana je u Angular 7 [12] radnom okviru. Jedna od najzanimljivijih korišćenih biblioteka koja je zadužena za izgled aplikacije jeste Angular Material [13]. Komunikacija između klijentske aplikacije i serverske aplikacije vrši se korišćenjem HTTP protokola [14] uz slanje JSON [15] objekata. Naredna komponenta jeste serverska aplikacija.



Slika 2 - Arhitektura sistema

Pisana je u Java 8 [16] programskom jeziku, a korišćen je SpringBoot [17] radni okvir. Ova aplikacija je implementirana kao troslojna RESTful (Representational state transfer) [18] aplikacija. Troslojna arhitektura serverske aplikacije obuhvata:

- Mrežni sloj,
- Servisni sloj i
- Repozitorijumski sloj.

Serverska aplikacija osluškuje zahtjeve koji stižu sa klijentske aplikacije. Kada pristigne novi zahtjev mrežni sloj je zadužen za obradu pristiglog zahtjeva. Nakon njegove obrade, mrežni sloj upućuje zahtjev servisnom sloju, koji kasnije upućuje zahtjev sloju repozitorijuma.

Repozitorijumski (engl. Repository) sloj je zadužen za dobavljanje podataka iz baze podataka. Posljednja komponenta sistema jeste sistem za upravljanje bazom podataka. Korišćena je MySQL [19] baza, koja predstavlja relaciju [20] bazu podataka.

Name	Description	Mac Address	IP Address	Serial number
Hp laptop	Small and portable PC	MS-00-25-SS-SS	192.168.100.25	KA010201
Network router	Wireless router	/	/	10SPSWILAN
HP WS 102-S	An old work station	00-4K-BS-76-SD-E2	100.525.50.15	H0101919SAQ00M
Lenovo Laptop	Personal computer	F0-03-8C-00-8C-0B	192.168.0.15	G5K1CX17K474852

Slika 3 - Komponenta za prikaz resursa

Sistem je izdijeljen na stranice. Za svaku vrstu resursa, navedenu u Modelu resursa, postoji posebna stranica. Svaka stranica se sastoji iz dvije komponente: komponente za prikaz resursa i komponente za modifikovanje resursa. Slika 3 prikazuje komponentu za prikaz resursa. Predstavljena je tabelom koja sadrži prethodno registrovane resurse. Slika 4 prikazuje komponentu za modifikovanje resursa. U obliku je forme koja se popunjava traženim podacima. Pored polja koja opisuju određeni resurs, postoje i akcije koje se mogu izvršiti nad resursom i nalaze se na naslovnoj traci ove komponente. Polja u komponenti će biti nepopunjena ukoliko u komponenti za prikaz resursa nije selektovana nijedna vrsta.

U ovom slučaju, moguće je registrovati novi resurs tako što se prvo popune polja a nakon toga i klikne na dugme + ili Save.

Ukoliko je selektovana jedna vrsta u tabeli registrovanih resursa, polja u formi su automatski popunjena odgovarajućim vrijednostima. Resurs je moguće ažurirati tako što se u odgovarajuća polja unesu izmjene, te se nakon toga klikne na Save ili Update dugme koje se nalazi u

sredini na naslovnoj traci (dugme u obliku olovke). Dodatno, selektovan resurs je moguće izbrisati ukoliko se klikne na Delete dugme, to jeste posljednje dugme u naslovnoj traci (dugme u obliku kante za smeće).

Nakon uspješnog ažuriranja ili brisanja resursa, tabela za prikaz registrovanih resursa je automatski ažurirana.

Slika 4 - Komponenta za modifikovanje resursa

## 5. ZAKLJUČAK

U radu je opisan sistem za upravljanje bezbjednošću računarskih resursa organizacije sa akcentom na model i implementaciju sistema. Sve tehnologije i modeli koji su korišćeni u realizaciji sistema i razlozi njihovog korišćenja su navedeni u okviru rada.

Opisan sistem nalaže da se za svaki resurs koji posjeduje organizacija vezuje zaposleni koji će predstavljati vlasnika. Vlasnik je odgovoran da prati neočekivana ponašanja svojih resursa. U cilju poboljšanja performansi, sistem može preuzeti ovaj zadatak na sebe, te čim uoči ranjivosti kontaktira vlasnika slanjem hitnog upozorenja na službeni broj telefona.

Pored toga, trenutno postoji pet šablona za generisanje izvještaja. U zavisnosti od organizacije u kojoj se koristi sistem, preporučljivo je implementirati još nekoliko šablona koji će zadovoljiti specifične potrebe organizacije.

## 6. LITERATURA

- [1] 1 NIST Special Publication 1800-5 - IT Asset Management.
- [2] Johannes W. van den Bent, „EXIN ITAMorg IT, Asset Management Foundation WorkBook“, November 2017.
- [3] <https://www.itassetmanagement.net/2015/02/19/importance-effective-ham-hardware-asset-management/>, (pristupljeno u julu 2019.)
- [4] <https://www.techopedia.com/definition/27131/software-asset-management-sam>, (pristupljeno u julu 2019.)
- [5] <http://ogabengineering.com/virtual-asset-management-system/>, (pristupljeno u julu 2019.)
- [6] <https://www.sciencedirect.com/topics/computer-science/information-asset-management>, (pristupljeno u julu 2019.)
- [7] <https://blog.assetsonar.com/it-asset-management-software-functions-benefits-and-implementation-c1d53159a535>, (pristupljeno u julu 2019.)

- [8] <https://road-asset.piarc.org/en/data-and-modeling-lifecycle-planning/lifecycle-planning>, (pristupljeno u julu 2019.)
- [9] <https://pecb.com/article/4-key-stages-of-asset-management-life-cycle->, (pristupljeno u julu 2019.)
- [10] <https://www.lce.com/Life-Cycle-Asset-Management-1112.html>, (pristupljeno u julu 2019.)
- [11] <https://www.javascript.com/>, (pristupljeno u avgustu 2019.)
- [12] <https://angular.io/docs>, (pristupljeno u avgustu 2019.)
- [13] <https://material.angular.io/>, (pristupljeno u avgustu 2019.)
- [14] David Gourley and Brian Tooty “HTTP: The Definitive Guide”, October 2002.
- [15] <https://www.json.org/>, (pristupljeno u avgustu 2019.)
- [16] James Gosling, Bill Joy, Guy Steele, Gilad Bracha, and Alex Buckley “The Java Language Specification”, March 2014.
- [17] [https://www.tutorialspoint.com/spring/spring\\_overview.html](https://www.tutorialspoint.com/spring/spring_overview.html), (pristupljeno u avgustu 2019.)
- [18] <https://www.codecademy.com/articles/what-is-rest>, (pristupljeno u avgustu 2019.)
- [19] <https://www.mysql.com/>, (pristupljeno u avgustu 2019.)
- [20] Jan L. Harrington, “Relational Database Design and Implementation”, May 2016.
- [21] [http://www.java2s.com/Book/Java/Language-Basics/Boolean\\_Logical\\_Operators.htm](http://www.java2s.com/Book/Java/Language-Basics/Boolean_Logical_Operators.htm), (pristupljeno u avgustu 2019.)

## Kratka biografija:



**Nevena Lazarević** rođena je 30. aprila 1996. godine u Zofingenu, u Švajcarskoj. Osnovnu školu „Tamara Begović“ i srednju školu gimnaziju „Vaso Pelagić“ u Brčko Distriktu završava kao nosilac Vukove diplome. Osnovne akademske studije upisuje 2014. godine na Fakultetu tehničkih nauka u okviru Univerziteta u Novom Sadu, smjer Softversko inženjerstvo i informacione tehnologije. 2018. godine završava osnovne i upisuje master akademske studije na istom fakultetu, na istom smjeru. Položila je sve ispite predviđene planom i programom.kontakt: lazarevic.nevena.43@gmail.com