UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Bojan Pirc

UVEDBA SISTEMA ZA UPRAVLJANJE Z IDENTITETAMI

DIPLOMSKO DELO NA
UNIVERZITETNEM ŠTUDIJU

Mentor: izr. prof. dr. Marjan Krisper

Ljubljana, 2010

Univerza
v Ljubljani

Fakulteta za računalništvo
in informatiko

Tržaška 25
1000 Ljubljana, Slovenija
telefon: 01 476 84 11
faks: 01 426 46 47
www.fri.uni-lj.si
e-mail: dekanat@fri.uni-lj.si

Št. naloge: 01666/2010
Datum: 05.04.2010

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Kandidat: **BOJAN PIRC**

Naslov: **UVEDBA SISTEMA ZA UPRAVLJANJE Z IDENTITETAMI**
IMPLEMENTATION OF IDENTITY MANAGEMENT SYSTEM

Vrsta naloge: Diplomsko delo univerzitetnega študija

Tematika naloge:

V okviru diplomskega dela predstavite področje upravljanja z identitetami, opredelite ključne funkcionalnosti sistemov za upravljanje z identitetami in opišite procese IT glede na smernice ITIL in CobiT, ki se navezujejo na to področje. Poleg predstavitve področja in opisa funkcionalnosti sistemov izdelajte primer uvedbe sistema za upravljanje z identitetami z Microsoft Forefront Identity Manager 2010 v izbranem podjetju.

Mentor:

prof. dr. Marjan Krisper

Fakulteta za računalništvo in informatiko • UNIVERZA V LJUBLJANI • 2

Dekan:

prof. dr. Nikolaj Zimic

Zahvala

Iskreno se zahvaljujem mentorju izr. prof. dr. Marjanu Krisperju za mentorstvo in nasvete pri izdelavi diplomske naloge. Zahvaljujem se tudi vsem svojim najbližjim, ki so mi stali ob strani skozi celoten študij, predvsem svoji ženi Tei.

Kazalo

Seznam uporabljenih kratic

AD	angl. *Active Directory*; Microsoftova rešitev imeniške storitve
ANSI	angl. *American National Standards Institute*; Ameriški državni inštitut za standarde
CobiT	angl. *Control Objectives for Information and related Technology*; zbirka dobrih praks upravljanja informatike v podjetju
DNS	angl. *Domain Name System*; domenski strežnik
DSD	angl. *Dynamic Separation of Duties*; dinamično ločevanje nalog
ERP	angl. *Enterprise Resource Planning*; celovita programska rešitev
FIM	angl. *Forefront Identity Manager*; Microsoftov sistem za upravljanje z identitetami tretje generacije
HRM	angl. *Human Resource Management*; upravljanje s človeškimi viri
IdM	angl. *Identity Management*; upravljanje z identitetami
ILM	angl. *Identity Lifecycle Manager 2007*; Microsoftov sistem za upravljanje z identitetami druge generacije
ISO	angl. *International Organization for Standardization*; Mednarodna organizacija za standardizacijo
IT	angl. *Information Technology*; informacijska tehnologija
ITIL	angl. *Information Technology Infrastructure Library*; zbirka dobrih praks upravljanja informatike v podjetju
ITU	angl. *International Telecommunication Union*; mednarodna zveza za telekomunikacije
LDAP	angl. *Lightweight Directory Access Protocol*; lahki protokol za dostop do imenikov
MA	angl. *Management Agent*; upravljavski agent – komponenta FIM Synchronization Servicea
MIIS	angl. *Microsoft Identity Integration Server 2003*; Microsoftov sistem za upravljanje prve generacije
MPR	angl. *Management Policy Rule*; upravljavsko pravilo – komponenta FIM Servicea
NIST	angl. *National Institute of Standards and Technology*; Narodni inštitut za standarde in tehnologije
RBAC	angl. *Role Based Access Control*; nadzor dostopa na podlagi vloge v organizaciji
SoD	angl. *Separation of Duties*; ločevanje nalog
SSD	angl. *Static Separation of Duties*; statično ločevanje nalog
WSS	angl. *Windows Sharepoint Services*; Microsoftova portalska rešitev

Povzetek

Večina organizacij se v današnjih časih srečuje s problemom nadzora nad dostopi do sistemov in virov organizacije. Tukaj se pojavlja ključno vprašanje kdo ima do kakšnega vira dostop in ali je do njega tudi upravičen. Pri reševanju teh vprašanj nam pomagajo sistemi za upravljanje z identitetami, kjer pa z uvedbo sistema v organizacijo ta ne pridobi le pregleda nad dostopi uporabnikov, temveč omogoča izvajanje zadane varnostne politike in procesov za zagotavljanje ustreznega nivoja varnosti. Sistemi za upravljanje z identitetami omogočajo tudi sledljivost zahtev uporabnikov in odobritev teh zahtev. Rezultat uvedbe sistema za upravljanje z identitetami je višji nivo IT varnosti v organizaciji, hkrati pa se samim uporabnikom zviša nivo zavedanja potrebe po varnosti s tem, da ima vsak svoje zadolžitve in odgovornosti pri zagotavljanju varnosti znotraj organizacije.

Namen diplomskega dela je predstavitev področja upravljanja z identitetami, opredelitev ključnih funkcionalnosti sistemov za upravljanje z identitetami in opis procesov glede na smernice ITIL in CobiT, ki se navezujejo na področje upravljanja z identitetami. Poleg predstavitve samega področja in opisa funkcionalnosti sistemov pa je glavni rezultat diplomskega dela praktičen prikaz uvedbe sistema za upravljanje z identitetami Microsoft Forefront Identity Manager 2010, s katerim so podprti osnovni procesi zaposlovanja in dodeljevanja dostopa v obravnavanem podjetju. Sama uvedba sistema vključuje vzpostavitev metaimenika med kadrovskim sistemom, imeniško storitvijo in Microsoft Forefront Identity Managerjem 2010, za katerega pa je opisan tudi postopek implementacije osnovnih procesov zaposlovanja ter zahtevanja in dodeljevanja dostopa.

Ključne besede

upravljanje z identitetami, metaimenik, imeniške storitve

Abstract

Many organizations nowadays face a problem of managing organization's system and resource access. The key question presented here is who has access to what resource and if that person is entitled to it. The identity management systems help us solve this question. By applying the identity management system into the organization, the organization doesn't only acquire overview of user access but it also allows the execution of the given security policy and processes which are required for ensuring the specified IT security level. The identity management systems provide auditing of user requests and approvals. The implementation of identity management system in the organization resolves in a higher level of IT security including increased user awareness of the need for IT security which is achieved by giving each user its responsibilities and accountabilities for ensuring IT security.

The purpose of the thesis is to present identity management scope, definition of key functionalities of the identity management systems and review of ITIL and CobiT guideline processes which refer to the identity management. Besides the presentation of the scope and the definition of key functionalities the main goal of the thesis is a practical display of Microsoft Forefront Identity Manager 2010 implementation which supports basic processes of employment and assignment of access to the company's resources. The implementation itself contains configuration and implementation of metadirectory between the human resources system, directory service and Microsoft Forefront Identity Manager 2010, for which the implementation procedure of the basic employment, request and approval processes, is also described.

Keywords

identity management, metadirectory, directory services

1 Uvod

Mnoge današnje, predvsem večje slovenske organizacije, imajo veliko število zaposlenih, pogosto tudi v več državah. Poleg tega pa je lahko informacijska struktura organizacij sestavljena iz več deset ali celo več kot sto sistemov, ki omogočajo nemoteno delovanje na različnih poslovnih področjih. To pomeni, da so informacije z vidika organizacije široko porazdeljene po različnih sistemih in dostopne večjemu številu zaposlenih. Mnoge od teh informacij so za organizacijo bistvenega pomena in vrednosti, zato je potrebno čim bolje poskrbeti, da ne uidejo iz organizacije ali celo v roke konkurence. Pri takih informacijah je seveda vedno potrebno vedeti kdo ima do njih dostop in tudi kdaj je imel kdo do njih dostop. Škoda, povzročena zaradi izgube za organizacijo pomembnih podatkov, je lahko ogromna. Problem obvladovanja nadzora nad takimi informacijam pa nastaja že pri 50 ali 100 zaposlenih, torej je pri tisoč ali več deset tisoč zaposlenih obvladovanje praktično nemogoče, vsaj ne v okviru sprejemljivih stroškov. Sistemi za upravljanje z identitetami so se ravno zaradi vrednosti informacij in vedno višjih zahtev po varnosti v zadnjem desetletju pojavili in razvili ter s tem opredelili današnje področje upravljanja z identitetami. Sistemi za upravljanje z identitetami obravnavajo zaposlene in njihove pravice kot identitete in njim dodeljene vloge.

Uvedba sistema za upravljanje z identitetami pa je za organizacijo običajno časovno in finančno zelo zahtevna. Pri tem je seveda potrebno upoštevati tudi velikost in razpršenost organizacije ter nenazadnje kakšno je obstoječe zavedanje pomembnosti varnosti in samo izvajanje akcij za zagotavljanje ustreznega nivoja varnosti. Del časa in stroškov pri uvedbi sistema se pripisuje tudi sami opredelitvi novih oz. dopolnitvi obstoječih varnostnih pravil in procesov. Pri tem je potrebno opredeliti tudi zadolžitve zaposlenih, ki se bodo izvajale preko sistema za upravljanje z identitetami, pa naj bo to ali oddaja zahteve ali pa potrjevanje zahteve za dodelitev dostopa. Prav tako je potrebno izvesti ustrezna izobraževanja, če je le možno za večino zaposlenih, saj bodo verjetno slej kot prej morali izvesti določene aktivnosti preko sistema za upravljanje z identitetami.

Namen diplomske naloge je podrobna predstavitev ključnih konceptov upravljanja z identitetami in praktična uvedba sistema za upravljanje z identitetami v podjetje. Naloga je vsebinsko razdeljena na dva dela. V prvem, teoretičnem delu, najprej na splošno predstavim sistem za upravljanje z identitetami, sledi opis splošne arhitekture, ključnih entitet in funkcionalnosti sistema za upravljanje z identitetami ter načinov razprševanja. Podrobneje obravnavam tudi procese današnjih IT smernic ITIL in CobiT, ki se navezujejo na področje

upravljanja z identitetami. Teoretični del se zaključi z opisom sistema za upravljanje z identitetami Microsoft Forefront Identity Manager 2010. V drugem delu diplomske naloge pa natančno prikažem praktičen primer uvedbe sistema za upravljanje z identitetami Forefront Identity Manager 2010. Za izvedbo praktičnega dela uvedbe sem pripravil lastno okolje, kjer sem namestil vse potrebne sisteme in pripravil obstoječe stanje zaposlenih tako v kadrovskem sistemu kot imeniški storitvi. V to okolje sem namestil Forefront Identity Manager 2010, pripravil in povezal metaimenik s kadrovskim sistemom, Microsoft Active Directoryjem in Microsoft Exchangeom 2010 ter glede na zadane poslovne procese zaposlovanja in dodeljevanja dostopa te procese tudi implementiral v sam sistem upravljanja z identitetami. Za samo praktično izvedbo implementacije sistema za upravljanje z identitetami sem izbral produkt Forefront Identity Manager 2010 predvsem zato, ker je glede na predhodnika (ILM) z novimi funkcionalnostmi postal bolj konkurenčen ostalim produktom in tudi zato, ker imam s tem produktom na svojem področju dela največ izkušenj.

Glavni cilj diplomskega dela je prikaz uspešne uvedbe sistema za upravljanje z identitetami v moje okolje, na katerega lahko impliciramo tudi marsikatero večje podjetje. Kakšne so posledice uvedbe sistema za organizacijo na kratko in daljše časovno obdobje pa predstavim v sklepnem delu naloge.

2 Sistem za upravljanje z identitetami (IdM sistem)

2.1 O IdM sistemu

Upravljanje identitet je možno opredeliti kot široko administrativno področje, ki obravnava identifikacijo posameznikov znotraj sistema in kontrolira dostop do virov tega sistema s postavitvijo omejitev na vzpostavljenih identitetah posameznikov [14].

IdM sistem tako lahko opredelimo kot sistem, ki pokriva področje upravljanja identitet. Mark Bergman in Joel Cooper sta IdM sistem označila kot »zbirko tehnologij, poslovnih procesov in temeljnih pravil, ki sistemom v omrežju omogočajo opredelitev kdo ima dostop do njih ter kdo in kdaj ima pravice do katerih virov, medtem pa je zagotovljena zasebnost posameznikov in varnost zaupnih podatkov« [2].

IdM sistemi so v zadnjih letih postali zelo razširjeni tako s strani ponudnikov programske opreme kot tudi v samih organizacijah, ki so te sisteme uvedle. Danes večina večjih ponudnikov programske opreme, kot so Oracle, Microsoft, Sum, IBM, HP in ostali, ponujajo lastne rešitve IdM sistemov. Te rešitve se med seboj zelo razlikujejo, saj je področje upravljanja identitet izjemno široko in kompleksno ter ni nobenih opredeljenih standardov, katerih bi se morali ponudniki programske opreme držati. Ponudniki se v glavnem držijo uveljavljenih smernic in dobrih praks IT, kot sta npr. ITIL ali CobiT, vsekakor pa vsi težijo k doseganju ciljev, ki jih opredeljuje področje upravljanja identitet.

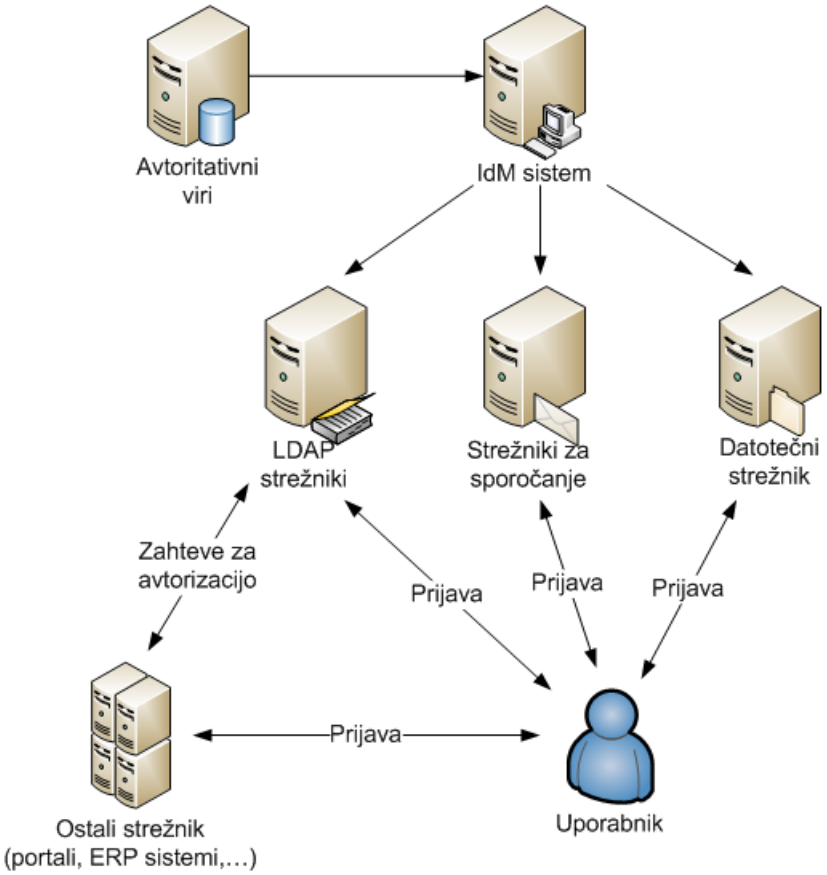
V naslednjih poglavjih bom podrobneje predstavil splošno arhitekturo IdM sistemov, ključne entitete in funkcionalnosti sistemov, načine razprševanja informacij ter opisal procese IT smernic ITIL in CobiT, ki se navezujejo na področje upravljanja identitet.

2.2 Splošna arhitektura

Čeprav je sam IdM sistem lahko avtoritativni vir podatkov, so običajno oz. v veliki večini to drugi sistemi, kot so ERP, kadrovski idr., iz katerih pa IdM periodično pridobiva podatke o identitetah in vlogah. IdM sistem dejansko poskrbi za ustvarjanje uporabnikov in dodeljevanje virov uporabnikom (angl. *provisioning*), odstranjevanje uporabnikov in njemu dodeljenih virov (angl. *deprovisioning*) ter o(ne)mogočanje uporabnikov ali posameznih dostopov.

Splošna arhitektura umestitve IdM sistema je predstavljena v spodnji sliki (Slika 2-1), ki prikazuje običajne težnje oblikovanja sistemov. Ko IdM iz avtoritativnega vira pridobi podatke o identiteti, zanjo ustvari ustrezne uporabniške račune in ji dodeli ustrezne dostope.

Ti podatki se nato prenesejo v razne ciljne sisteme, kot so imeniške storitve, strežniki za sporočanje oz. strežniki za katere IdM sistem neposredno skrbi za dostope. Ostali sistemi, ki niso neposredno vezani na IdM sistem, pa preverjajo avtorizacije preko imeniških storitev. Uporabniki se tako lahko prijavijo na katerikoli sistem – sistem, ki je neposredno povezan z IdM sistemom ali pa posredno povezan sistem [9].



Slika 2-1: Splošna arhitektura IdM sistemov [9]

2.2.1 Imeniške storitve

Imeniške storitve (angl. *directory services*) so programski sistem, ki hrani, organizira in nudi dostop do informacij v imeniku. Pri razvoju programske opreme imenik predstavlja seznam razlikovanj nazivov in njihovih vrednosti oz. ponuja vpogled v imenovane vrednosti, podobno kot slovar. Imeniki imajo lahko zelo majhen doseg, kjer podpirajo majhno število vrst vozlišč in podatkovnih tipov, lahko pa podpirajo tudi zelo veliko število tipov. Imeniške storitve dejansko definirajo imenski prostor za omrežje. V tem kontekstu je imenski prostor termin, ki se uporablja za hrambo enega ali več objektov kot imenovane zapise [13].

Skozi zgodovino je bilo izdelanih in uporabljenih veliko imeniških storitev. V naslednjih podpoglavjih bom opisal imeniške storitve, ki so bile v zgodovini in so tudi še danes najbolj razširjene.

2.2.1.1 X.500

X.500 ali težki protokol za dostop do imenikov, se pogosto imenuje tudi pradedek imeniških storitev, saj je njegova opredelitev osnova za številne imenike, ki so danes v uporabi v organizacijah. X.500 je celovita specifikacija imeniških storitev, ki je v osnovi nastala kot distribuirana, omrežnoneodvisna imeniška storitev za sporočilne storitve, katerih specifikacija se je imenovala X.400. Družino specifikacij X.500 so v 80-ih letih razvili v »International Standards Organization« ali ISO in v »International Telecommunication Union« ali ITU. Najbolj poznani član te družine je avtentifikacijsko ogrodje X.509, ki se uporablja za infrastrukturo javnih ključev, kar pa je osnova za digitalne certifikate [10, 13].

2.2.1.2 LDAP

LDAP ali lahki protokol za dostop do imenikov, je bil ustvarjen za poenostavitev določenih funkcionalnosti X.500. LDAP je bil prvotno tudi ustvarjen kot protokol, ki bi se ga implementiralo kot prehod do X.500. Zasnovan je na TCP/IP mrežnem protokolu in vsebuje programski vmesnik za kliente, kar je X.500 tudi manjkalo. Namesto prehoda do X.500 storitev, se je LDAP tako razvil v celotno imeniško storitev.

LDAP vsebuje tudi metode za povezovanje na druge LDAP strežnike, ki lahko skupaj sodelujejo ter ustvarijo enoten in celovit virtualni imenski prostor iz imenskih prostorov individualnih strežnikov. Čeprav večina LDAP strežnikov nudi načine replikacije, pa za to ni definiranega nobenega standarda in zaradi različnih implementacij posameznih proizvajalcev, strežniki ne morejo delovati povezano skupaj [10].

Posledica razširjenosti LDAP protokola je tudi ponudba sistemov zasnovanih na LDAP protokolu. Spodnja tabela (Tabela 2-1) prikazuje nekaj največjih ponudnikov programskih rešitev in njihovih LDAP produktov.

Ponudnik programskih rešitev	LDAP produkt
Microsoft	Active Directory
Apple	Open Directory
Oracle	Oracle Internet Directory
Novell	eDirectory
IBM	Tivoli Directory Server
Sun Microsystems	OpenDS

Tabela 2-1: Največji ponudniki LDAP rešitev

Ena izmed alternativ zgoraj prikazanim LDAP produktom je tudi OpenLDAP, ki je brezplačna odprtokodna rešitev.

2.2.1.3 DNS

Tipični predstavnik imeniških storitev, ki ne temelji na standardih X.500 in LDAP, je sistem domenskih imen ali DNS. Ta sistem je namenjen preslikavi domenskih imen in IP naslovov, uporabljal pa se je še pred predstavitvijo X.500 in LDAP standardov. DNS je porazdeljen imenik, ki je namenjen omogočanju infrastrukture za enovit, globalen imenik domenskih imen. Imenik je sestavljen iz tisočih strežnikov v lasti tisočih svetovnih organizacij. Arhitektura DNS tako omogoča strežnikom učinkovito in kooperativno odgovarjanje na poizvedbe glede preslikav domenskih imen, hkrati pa strežniki nudijo delegirano kontrolo nad preslikavami za katerikoli imenski naslov [10].

2.2.2 Imeniške storitve in IdM sistemi

Enotna prijava ali SSO (Single sign-on) dandanes predstavlja osnovo za številne poslovne in druge organizacije, saj je za uporabnike precej neprijetno, da morajo hraniti in upravljati z več digitalnimi poverilnicami hkrati oz. si morajo zapomniti ter vnašati uporabniška imena in gesla za vsak sistem. Tak način prijavljanja pa seveda predstavlja težave tudi za samo organizacijo, saj so zaposleni manj efektivni in posledično povzročajo višje stroške zaradi obračanja se na center za pomoč uporabnikom. Raziskave so pokazale, da se z uvedbo enotne prijave število klicev v center za pomoč uporabnikom zmanjša za 33 %, prav tako pa se varnost v splošnem dvigne za 32 %. Poleg statistike, ki jo prikazujejo raziskave, pa je ključen rezultat tekoča uporabniška izkušnja [6, 10].

Problem enotne prijave so organizacije delno ali v celoti rešile z uvedbo imeniških storitev. Te namreč omogočajo prijavo v sistem ter dostop do datotečnih sistemov, portalov ali poštnih strežnikov z enim uporabniškim imenom in geslom. Kljub temu pa še vedno obstajajo sistemi,

ki ne uporabljajo avtentifikacije preko imeniške strukture ali pa uporabljajo imeniške strukture drugih proizvajalcev. Probleme s takimi sistemi se rešuje z različnimi načini agregacije podatkov identitet [10].

2.2.3 Načini agregacije podatkov identitet

Načini agregacije podatkov identitet za sisteme, ki ne uporabljajo avtentifikacije preko imeniške strukture oz. uporabljajo imeniške strukture drugih proizvajalcev, so:

- izdelava centralne hrambe identitet;
- izdelava metaimenika za sinhronizacijo podatkov identitet med imeniki;
- izdelava virtualnega imenika, ki služi kot integriran pregled nad podatki identitet vseh imenikov v organizaciji;
- federativni imeniki, ki povezujejo hrambe podatkov identitet [10].

2.2.3.1 Centralna hramba identitet

Izdelava centralne hrambe identitet je izvedljiva le za manjše organizacije. Ključna naloga centralne hrambe identitet je opredelitev oz. izdelava globalnega enoličnega identifikatorja, na katerega se lahko povežejo različne identitete preostalih sistemov. Preslikovanje povezav na en identifikator je konceptualno lahka naloga, vendar je težka za implementacijo, saj pomeni spremembo v implementaciji sistemov, kar pa lahko močno dvigne stroške organizacije. Namreč kakršnakoli sprememba pri preslikovanju posledično pomeni, da se morajo nanjo odzvati vsi sistemi, ki uporabljajo centralno hrambo identitet, kar pa lahko predstavlja izjemno visoke stroške [10].

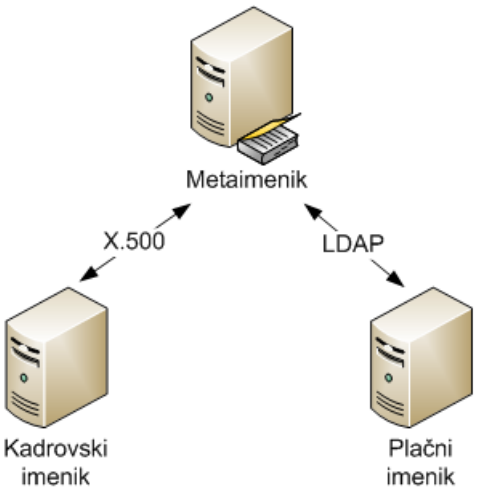
2.2.3.2 Metaimeniki

Metaimeniki so zbirke informacij o imenikih iz različnih virov imenikov, te informacije pa so agregirane na tak način, da je zagotovljen enoten pogled na vse podatke.

Današnje organizacije hranijo informacije v večjem številu imenikov vseh vrst, vsi imeniki pa ne vsebujejo informacij, ki jih je potrebno agregirati, kljub temu pa je večino imenikov možno agregirati brez poseganja v delovanje imenika ali aplikacije, ki jo ta uporablja. Metaimeniki omogočajo organizaciji zbiranje podatkov iz obstoječih imenikov v eno celovito hrambo informacij, po kateri je možno izvajati poizvedbe, kot če bi bile vse informacije shranjene v enem samem imeniku.

Metaimeniki delujejo preko programskih agentov, katerih naloga je pridobivanje podmnožice informacij imenika iz skupine imenikov, kar pa ni tako enostavno kot samo agregiranje nepovezanih zapisov. Uporaba metaimenikov zelo pogosto vključuje agregacijo atributov enega subjekta iz več imenikov v en skupen zapis. Metaimeniki lahko tudi dvosmerno sinhronizirajo podatke in se jih tako lahko uporablja za prenos podatkov iz enega imenika v drugega.

Primer metaimenika, ki je podan na spodnji sliki (Slika 2-1Slika 2-2), predstavlja agregacijo zapisov iz dveh imenikov – kadrovskega in plačnega. Metaimenik pridobiva podatke o delovnem mestu in statusu iz kadrovskega imenika preko imeniške storitve X.500, podatke o plačilnem razredu pa iz plačnega imenika preko imeniške storitve LDAP. Podatki se agregirajo v en zapis, ki predstavlja enoten pogled na zaposlenega – metaimenik v tem primeru ignorira ostale atribute iz kadrovskega in plačnega imenika, saj so za ta primer nepomembni [10].



Slika 2-2: Primer metaimenika [10]

Glavne prednosti, ki jih nudi tehnologija metaimenikov so:

- Enotna točka reference nudi abstraktno mejo med aplikacijo in dejansko implementacijo, zaradi katere lahko organizacija zamenja proizvajalca imenika, spremeni sistemske implementacije ali reorganizira podatke, aplikacije pa izvajajo poizvedbe samo na enem viru.
- Enotna točka administracije, katera zmanjšuje breme dostopanja do več imenikov z več vmesniki za vzdrževanje podatkov.

- Zmanjšano administrativno breme pri upravljanju s podvojenimi podatki, saj se lahko odvečne informacije v imenikih odstrani.

Problemi, ki se pojavljajo pri metaimenikih so:

- Težave pri ugotavljanju iz katerih imenikov je sestavljen zapis v metaimeniku.
- Izogibanje konfliktom imenskega prostora med različnimi agregiranimi imeniki.
- Pri implementaciji metaimenika je potrebno določiti kje se lahko podatke spreminja. Sinhronizacije lahko zapisujejo na nivoju metaimenika, na nivoju imenikov ali pa na obeh nivojih. Več kot je nivojev, na katerih je možno spreminjaje podatkov, bolj kompleksne so sinhronizacije [10].

2.2.3.3 Virtualni imeniki

Virtualni imeniki uporabljajo precej podoben koncept kot metaimeniki, saj prav tako ustvarijo enoten imenik z enotnim pogledom na različne neodvisne imenike, razlika pa je v načinu pridobivanja podatkov. Metaimeniki namreč tipično uporabljajo programske agente za replikacijo in sinhronizacijo podatkov iz različnih imenikov, virtualni imeniki pa naredijo enovit pogled na več imenikov z uporabo poizvedb v realnem času, ki so zasnovane na preslikavi iz polj virtualne sheme v polja fizične sheme imenika. Druga razlika med virtualnimi imeniki in metaimeniki je tudi ta, da imajo metaimeniki svojo hrambo podatkov in informacije v imenikih se tudi ohranjajo, medtem ko virtualni imeniki nimajo svoje, ločene hrambe podatkov. Virtualni imeniki, ki se običajno uporabljajo v primerih, ko je pomemben dostop do pogosto se spreminjajočih podatkov v realnem času, namreč delujejo tako, da eno poizvedbo spremenijo v več poizvedb na različne fizične imenike ter nato agregirajo rezultate poizvedbe v realnem času in vrnejo podatke uporabniku. Na ta način virtualni imeniki rešujejo težave povezane s sinhronizacijo podatkov in replikacijo [10].

2.2.3.4 Federativne identitete

Tako kot pri metaimenikih in virtualnih imenikih, tudi pri federativnih identitetah podatki o identitetah ostajajo na distribuiranih lokacijah. Ključna razlika med federativnimi identitetami in preostalima dvema tehnologijama pa je ta, da se pri federativnih identitetah ne naredi enotnega pogleda na infrastrukturo identitet, ampak federacija definira procese in podporno tehnologijo, tako da lahko različne hrambe identitet kooperativno rešujejo naloge identitet. Federacija pomeni to, da lokalne identitete in njihovi podatki ostajajo na mestu, vendar so

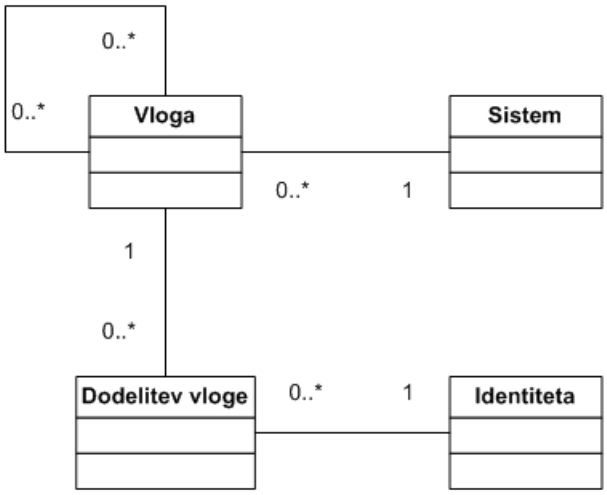
povezani skupaj z mehanizmom na višjem nivoju [10]. Tipično za IdM sisteme je ravno uporaba federativnih identitet, za sinhronizacijo med sistemi pa se uporablja metaimenike.

2.3 Ključne entitete IdM sistema

Sistemi za upravljanje z identitetami uporabljajo štiri ključne entitete:

- identiteta,
- vloga,
- sistem in
- dodelitev vloge.

Spodnja slika (Slika 2-3) prikazuje osnovni metamodel entitet sistemov IdM. Kot lahko razberemo iz slike, sistemom pripadajo vloge, vsaki vlogi pa lahko pripada več identitet oz. vsaka identiteta ima lahko dodeljenih več vlog. Povezava med vlogami in identitetami se v metamodelu prikazuje kot entiteta dodelitev vloge, ki predstavlja vezno entiteto med vlogami in identitetami. Model pa seveda dovoljuje tudi hierarhično strukturo vlog.



Slika 2-3: Metamodel entitet IdM sistemov

Seveda pa ima vsak IdM sistem svojo implementacijo in lahko za nazive entitet uporablja druge izraze. Prav tako nekatere entitete niso implementirane v takem smislu kot so tukaj prikazane – primer tega je dodelitev vloge, ki je lahko implementirana kot članstvo v vlogi, ta podatek pa je shranjen kot večvrednosti atribut vloge.

2.3.1 Identiteta

Beseda »identiteta« se nanaša na velik razpon možnih samostalnikov – od množice lastnosti, ki določajo osebnost, do zgoščene vrednosti gesla [3].

Digitalna identiteta vsebuje podatke, ki enolično opisujejo osebo ali stvar, imenovano tudi subjekt ali entiteta, in prav tako vsebuje informacije o povezavah subjekta do drugih entitet. Subjekt ali entiteta je oseba, organizacija, programska oprema, računalnik ali katerakoli stvar, ki zahteva dostop do vira [10]. *Digitalna identiteta je osebna identifikacijska informacija, ki je selektivno izpostavljena omrežju* [14].

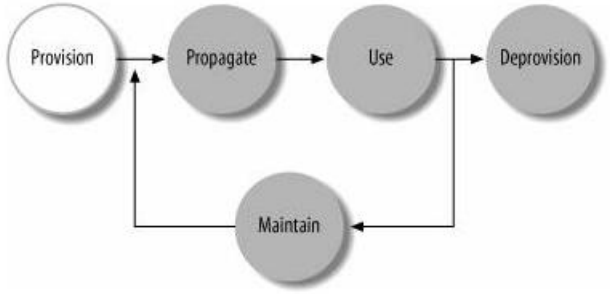
Identiteta oz. digitalna identiteta je digitalna reprezentacija množice zahtev enega digitalnega subjekta zase ali za nekoga drugega. Digitalni subjekt je entiteta obstoječa v digitalnem svetu, katerega se obravnava oz. opisuje. Vsak digitalni subjekt ima končno, vendar neomejeno število atributov, ki opisujejo identiteto. Digitalni subjekt je lahko človeški ali pa tudi ne. Primeri nečloveških digitalnih subjektov so:

- naprave in računalniki,
- digitalni viri ter
- pravila in povezave med ostalimi digitalnimi subjekti [12].

2.3.1.1 Življenjski cikel identitete

Življenjski cikel identitete obstaja tako v kompleksnem sistemu za upravljanje z identitetami za veliko organizacijo, kot tudi pri uporabniških računih na domačem računalniku. Razumevanje posameznih korakov oz. kako se odvija življenjski cikel na vsakem sistemu v organizaciji in v organizaciji kot celoti, pa je ključno pri oblikovanju strategije za upravljanje digitalnih identitet.

Slika 2-4 prikazuje življenjski cikel digitalne identitete, ki je sestavljen iz ustvarjanja (angl. *provisioning*), razprševanja (angl. *propagation*), uporabe (angl. *usage*), vzdrževanja (angl. *maintaining*) in odstranjevanja identitete (angl. *deprovisioning*). Življenjski cikel se torej začne s tem, ko je digitalna identiteta ustvarjena. V naslednjem koraku se digitalna identiteta prenese oz. razprši v katerikoli sistem, ki jo bo uporabil. Sledi sama uporaba identitete. Občasno se na identiteti izvaja tudi določeno vzdrževanje, kot je npr. zamenjava gesla ali sprememba atributa. Po vzdrževanju se identiteta nato ponovno prenese v sisteme. Življenjski cikel digitalne identitete se zaključi na točki, ko se identitete ne potrebuje več, zato se jo odstrani oz. uniči [10]. Podroben opis posamezne faze je opisan v nadaljnjem besedilu.



Slika 2-4: Življenjski cikel identitete [10]

2.3.1.1.1 Ustvarjanje identitete

Prvi korak življenjskega cikla digitalne identitete (ustvarjanje identitete) zajema proces pripravljanja informacijskega sistema, da zagotovi storitev klientu, stranki ali drugemu uporabniku. Iz vidika digitalne identitete ta faza pomeni izdelavo samega zapisa identitete in napolnitev atributov z ustreznimi podatki. Atributi so lahko standardni podatki (ime, lokacija, elektronski naslov, telefon …) ali pa podatki specifični za točno določen sistem. Začetek življenjskega cikla digitalne identitete oz. njeno ustvarjanje se lahko izvede z akcijo administratorja, samopostrežno ali samodejno. Ob zaposlitvi nove osebe običajno administratorji na več sistemih ustvarijo identiteto, da lahko novozaposleni dostopa do pisarne, dobiva plačo, uporablja delovno postajo itd. V organizacijah pa je prisoten trend imenovan »štart nulti-dan«, ki pomeni, da lahko novozaposleni že isti dan začne dejansko delati in ima ob tem dostop do vseh potrebnih virov [10]. Z ustreznimi poslovnimi procesi in avtomatizacijo dostopov pa IdM sistemi ta trend zmanjšujejo iz povprečno 10 ur na zaposlenega na 1 uro [6]. S tem se tudi prelaga delo in odgovornost iz administratorjev na neposredno odgovorne osebe zaposlenega – vodje zaposlenih in lastnike vlog oz. virov.

2.3.1.1.2 Razprševanje identitete

Faza razprševanja identitete v druge sisteme oz. potreba po razprševanju je odvisna od oblike sistema, v katerem se identiteta ustvari. Pri enostavnih sistemih zadostuje samo zapis informacij o identiteti na datotečni sistem ali zapis v podatkovno bazo, kompleksnejši sistemi pa lahko nudijo neke vrste deljen imenik identitet, kjer je identiteta kreirana na enem mestu, uporablja pa se v več sistemih. Primer take vrste deljenega imenika je LDAP imenik. IdM sistemi pa te sisteme povezujejo in skrbijo za ustrezen prenos podatkov, ki se mora izvesti po vsaki spremembi zapisa identitete. Prenos podatkov se mora izvesti zanesljivo. Ker pa vsi sistemi ne ponujajo možnosti prenosa podatkov v transakcijah, to postane izziv. Večji problem predstavlja predvsem večje število izvedenih akcij v enem prenosu. Zato je potrebno

pri takih sistemih previdno razmisliti o potrebnih nadomestnih akcijah, preden se jih uvede [10].

2.3.1.1.3 Uporaba identitete

Ko je enkrat identiteta kreirana in prenesena v ustrezne sisteme, se jo lahko poljubno uporablja. To je s stališča sistema za upravljanje z identitetami najbolj preprosta faza, saj za samo uporabo identitet skrbijo ravno sistemi, kamor se je identiteta zapisala.

2.3.1.1.4 Vzdrževanje identitete

Skozi celoten življenjski cikel identitete je potrebno njeno vzdrževanje. Ne glede na naravo identitete, se atributi skozi čas spreminjajo, kar pa se običajno zgodi ali zaradi spremembe atributov identitete (npr. nov domači naslov), ali zaradi spremembe vlog in dostopov. Spreminjanje ali dodajanje novih atributov pa je lahko tudi posledica novih poslovnih priložnosti ali uvedbe popolnoma novih informacijskih sistemov [10]. Najbolj pomembno pa je to, da se po vsaki spremembi podatkov identitete, te spremembe prenesejo tudi v preostale sisteme, na katere sprememba vpliva.

Vzdrževanje identitet je postalo tudi ena izmed dražjih aktivnosti, saj se s tem dnevno ukvarja center za pomoč uporabnikom, kar pa predstavlja veliko časovno in stroškovno porabo. Pogosto se namreč dogaja, da uporabniki pozabijo svoje geslo, zamenjajo vlogo v organizaciji ali pa zamenjajo lokacijo delovnega mesta. In več aktivnosti, ki jih lahko uporabnik opravi sam, posledično pomeni nižje stroške za organizacijo, saj se na ta način uporabnik ne obrača v tolikšni meri na center za pomoč uporabnikom [10]. Ta problem do določene mere zmanjšujejo sistemi za upravljanje z identitetami z uvedbo samopostrežnih aktivnosti in sistemi za upravljanje z gesli, kjer si lahko uporabniki sami ponastavijo geslo.

2.3.1.1.5 Odstranjevanje identitete

Zadnji korak življenjskega cikla identitete predstavlja odstranjevanje identitete, ki je enako pomembno kot njeno ustvarjanje. Dobri sistemi za upravljanje z identitetami omogočajo, da zaposleni lahko še isti dan začnejo učinkovito delati, poskrbeti pa morajo tudi za odstranitev identitete in dostopov do virov. Za organizacijo je zelo pomembno, da novozaposleni lahko še isti dan začne delati in da ob njegovem odhodu nima več nobenega dostopa. Organizacija mora spremljati uporabniške račune. Uporabniški računi in dostopi namreč predstavljajo dve večji tveganji za organizacijo, saj ima uporabnik oz. oseba, ki ni več zaposlena, dostop do virov organizacije, če se identitete ne odstrani pravočasno, kar lahko v veliki meri zlorabljajo

hekerji. Če uporabniku torej ostane dostop do določenih virov sistema ali celo do zaupnih podatkov, je to za organizacijo zelo veliko tveganje [10].

2.3.2 Vloga

Vloga predstavlja dostop do sistema, dostop do storitve sistema ali pa samo storitev sistema, ki je lahko pravica, dostop, programska oprema, poštni predal in varnostna ali distribucijska skupina. Vloga lahko torej predstavlja veliko število stvari. Prednost širokega pomena vloge pa je v tem, da lahko v IdM sistem vključimo veliko število različnih sistemov in njihove lastnosti oz. dostope opredelimo kot vloge. Z večjim številom sistemov, ki so povezani s sistemom IdM, pa dobimo bolj celovit pregled nad dodeljenimi dostopi in pravicami oz. storitvami.

Tabela 2-2 prikazuje primere vlog za različne sisteme.

Identiteta	**Vrsta vloge**	**Sistem**	**Izvedba vloge v sistemu**
Bojan Pirc (oseba)	Dostop	Microsoft Active Directory	Uporabniški račun
Bojan Pirc (oseba)	Pravica	Datotečni sistem	Uporabnik Bojan ima pravico branja v mapi Dokumenti
NB-bojan (prenosni računalnik)	Programska oprema Microsoft Office	Microsoft SMS	Nameščen Microsoft Office na prenosnem računalniku NB-bojan
Bojan Pirc (oseba)	Poštni predal	Microsoft Exchange	Poštni predal bojan.pirc
Bojan Pirc (oseba)	Varnostna ali distribucijska skupina	Microsoft Active Directory	Članstvo v varnostni ali distribucijski skupini.

Tabela 2-2: Primeri vlog za posamezni sistem

2.3.3 Dodelitev vloge

Dodelitev vloge predstavlja povezavo med identiteto in vlogo. Dodelitev vloge je v splošnem preprosta entiteta, ki vsebuje referenco na identiteto in vlogo. Poleg tega pa običajno vsebuje tudi datum veljavnosti dodelitve vloge in dodatne atribute (npr. naziv uporabniškega imena, naziv poštnega predala itd.), ki so specifični za to dodelitev.

Če pogledamo zgornjo tabelo (Tabela 2-2), ki prikazuje identitete, vrste vloge in izvedbe vloge v posameznem sistemu, lahko izvedbo vloge v sistemu opredelimo kot posledico realizacije dodelitve vloge v posameznem sistemu – znotraj IdM sistema dodelitev vloge predstavlja uporabniški račun, pravico do branja v mapi, poštni predal, članstvo v skupini ... Seveda gre tukaj ponovno opozoriti, da so IdM sistemi različno implementirani in je tako lahko entiteta dodelitev vloge uporabljena na drugačen način ali pa je uporabnikom skrita oz.

je implementirana tako, da je uporabnik ne vidi oz. je ne more neposredno uporabljati. Primer tega je implementacija dodelitve vloge preko specifičnih atributov identitet ali vlog.

2.3.4 Sistem

Sistem predstavlja informacijski sistem ali storitev, ki je del organizacije, lahko pa je tudi zunanji sistem. Tako je sistem lahko imeniška storitev, spletni portal, ERP ali HRM sistem, sistem za nadzor proizvodnje, sistem za upravljanje s programsko opremo, sistem za upravljanje z elektronsko pošto itd. Večje organizacije imajo lahko več deset ali celo sto in več različnih sistemov, ki so lahko podporni ali pa tudi ključni za samo delovanje organizacije.

S stališča IdM postane nek sistem pomemben za IdM sistem takrat, kadar se ga poveže s samim IdM sistemom. To pomeni, da sistem postane vir ali ponor informacij, s katerimi IdM sistem upravlja. Cilj organizacij je, da z IdM sistemom poveže čim večji nabor sistemov in storitev, katere se v organizaciji uporablja, saj se le tako lahko zagotovi širok nadzor ter pregled nad dostopi in pravicami identitet.

2.4 Ključne funkcionalnosti

V poglavju bom opisal ključne funkcionalnosti, ki jih ponujajo današnji IdM sistemi. Predstavil bom samopostrežbo, nadzor dostopa na podlagi vloge, ločevanje nalog, poročila skladnosti, revizijo dostopa, eskalacijo, delegacijo in podporo poslovnim procesom. Tukaj seveda niso naštete vse funkcionalnosti IdM sistemov, saj mnogi ponudniki ponujajo dodatne, bolj napredne ali razširjene funkcionalnosti oz. module.

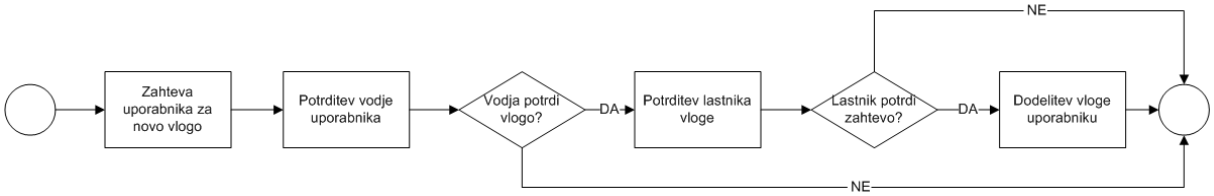
2.4.1 Samopostrežba

Samopostrežba (angl. *self-service*) je ena izmed ključnih funkcionalnosti, ki jih ponujajo IdM sistemi. Namen samopostrežbe je, da se do določene mere delegira vlogo upravljanja končnim uporabnikom in s tem razbremeni center za pomoč uporabnikom. Posledica razbremenitve centra za pomoč uporabnikom pa je seveda tudi zmanjšanje operativnih stroškov podjetja. Z uvedbo IdM sistema se samopostrežbo običajno uporablja za:

- zahteve za nove vloge ali dostope,
- ponastavitev gesla in
- upravljanje lastnih podatkov.

2.4.1.1 Zahteve za nove vloge in dostope

Pri zahtevanju nove vloge ali dostopa gre za proces, kjer končni uporabnik sam izbere oz. določi katere vloge ali dostope potrebuje. Da pa se zagotovi upravičenost zahteve uporabnika za dostop, se v procesu uporablja potrjevanje zahteve. Običajno se uporablja eno- ali dvonivojsko potrjevanje, redkeje pa se uporablja trinivojsko potrjevanje ali pa avtomatska odobritev brez potrjevanja. V primeru enonivojskega potrjevanja potrjuje vodja identitete, ko gre za dvonivojsko potrjevanje, pa je najprej potrebna potrditev vodje identitete in nato še lastnika vloge. Primer procesa za zahtevo dostopa z dvonivojskim potrjevanjem je prikazan v spodnji sliki (Slika 2-5).



Slika 2-5: Standardni proces za zahtevo dostopa z dvonivojskim potrjevanjem

Več kot je nivojev potrjevanja, manj je možnosti za zlorabe, vendar pa se s tem povečuje tudi čas odobritve zahteve in kompleksnost samega procesa.

2.4.1.2 Ponastavitev gesla

Samopostrežba za ponastavitev gesla, ki spada v koncept upravljanja gesel (angl. *password management*), je proces, s katerim lahko uporabniki sami restavrirajo pozabljeno geslo. Raziskave so pokazale, da ima center za pomoč uporabnikom približno 45 % klicev povezanih s ponastavitvijo gesel in da 15 % uporabnikov pokliče center za pomoč uporabnikom za ponastavitev gesla [6]. Z uvedbo sistema za upravljanje z gesli in samopostrežbo za ponastavitev gesel, se število klicev v center za pomoč uporabnikom drastično zmanjša, hkrati pa je sistem na voljo 24 ur na dan.

Samopostrežba za ponastavitev gesla temelji na temu, da mora uporabnik v primeru pozabljenega gesla sistemu dokazati, da je to res on in da si lahko ponastavi geslo. Najbolj razširjen način dokazovanja avtentičnosti uporabnika deluje tako, da mora uporabnik ob prvi prijavi v sistem za upravljanje z gesli odgovoriti na več vnaprej določenih vprašanj, ki temeljijo na osebnih podatkih uporabnika. Primeri vprašanj so: dekliški priimek matere, športni klub za katerega navijaš, ime prve domače živali itd. S tem ko uporabnik odgovori na ta vprašanja, se uspešno registrira v sistem za upravljanje z gesli. Če torej uporabnik pozabi

svoje geslo, se prijavi v sistem za upravljanje z gesli tako, da odgovori na vprašanja in si nato ponastavi geslo.

2.4.1.3 Upravljanje lastnih podatkov

Z uvedbo procesa za upravljanje lastnih podatkov, lahko uporabniki sami popravljajo podatke o svoji identiteti. Seveda gre tukaj le za točno določene podatke, običajno za tiste, katerih avtoritativni vir je IdM sistem. Najbolj pogosto se ta proces uporablja pri zunanjih sodelavcih, kjer se jim omogoči, da sami spreminjajo kontaktne podatke, ki so zapisani na identiteti. V večini primerov je ob spremembi podatkov potrebna potrditev vodje, kar tudi zmanjšuje možnost zlorab podatkov.

2.4.2 Nadzor dostopa na podlagi vloge

Nadzor dostopa na podlagi vloge ali RBAC (angl. *Role Based Access Control*) je sistem namenjen avtomatizaciji dodeljevanja dostopa identitetam, na podlagi vloge v podjetju. Proces uvajanja RBAC strukture v organizacijo pa se imenuje inženirstvo vlog (angl. *Role engineering*).

Področje RBAC je zelo razširjeno. Poleg RBAC standardov za splošne namene, so bili opredeljeni tudi RBAC standardi za uporabo v vojski, zdravstvu, industriji in biometriki. IdM sistemi, ki ponujajo RBAC, uporabljajo RBAC za splošne namene, sama uporaba RBAC pa je namenjena večjim organizacijam. RBAC je danes razširjen v podatkovnih bazah, upravljavskih sistemih in operacijskih sistemih.

Lahko rečemo, da je RBAC tako stara kot nova tehnologija. V zgodovini je prišlo zaradi pomanjkanja standardov do različnih implementacij RBAC-ja. Leta 2004 pa je ANSI predstavil standard ANSI INCITS 359-2004, ki je osnovni RBAC standard za splošne namene. ANSI standard temelji na NIST modelu, ki je bil opredeljen leta 2000 in od takrat dalje še dodatno izpopolnjen. Ker pa je RBAC amorfni koncept, je bilo predstavljenih in implementiranih še veliko drugih modelov, katerih ne bom opisoval [1].

Osnovni koncept RBAC je vzpostaviti pravice in dostope na podlagi funkcionalnih vlog v podjetju in nato ustrezno dodeliti uporabnikom vlogo ali skupino vlog. Funkcionalne vloge v podjetju so lahko definirane na podlagi delovnega mesta, organizacijske strukture, nalog, odgovornosti itd. [4]

NIST model definira standardni referenčni model RBAC, ki je razdeljen na štiri nivoje. Temelji na uporabi pozitivnih pravic[1], pri čemer negativne pravice niso izključene in je neodvisen od same narave vlog. Poleg tega pa ne opredeljuje načina avtentifikacije in zahtev po skalabilnosti števila uporabnikov, vlog ali pravic.

Nivoji RBAC po NIST modelu so:

- linearni RBAC,
- hierarhični RBAC,
- omejen RBAC in
- simetrični RBAC.

Nivoji so med seboj kumulativni in vsak nivo doda eno novo zahtevo. Tabela 2-3 prikazuje kratek povzetek osnovnih funkcionalnosti posameznega nivoja in kako se ti nivoji med seboj povezujejo. Podrobnejši opis posameznega nivoja pa sledi v naslednjih podpoglavjih.

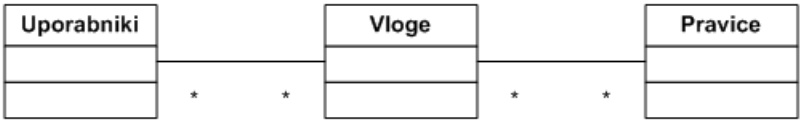
Nivo	**Naziv**	**Funkcionalnosti RBAC**
1.	Linearni RBAC	- uporabnik pridobi pravice preko vloge; - mora podpirati mnogo proti mnogo relacijo med uporabnikom in vlogo; - mora podpirati mnogo proti mnogo relacijo med pravico in vlogo; - mora podpirati pregled relacije med uporabniki in vlogami; - uporabniki lahko hkrati uporabljajo pravice več vlog.
2.	Hierarhični RBAC	Poleg vseh funkcionalnosti linearnega RBAC: - mora podpirati hierarhijo vlog; - Nivo 2a: podpira neomejene hierarhije; - Nivo 2b: podpira omejene hierarhije.
3.	Omejen RBAC	Poleg vseh funkcionalnosti hierarhičnega RBAC: - mora uveljavljati razdelitev nalog (SoD); - Nivo 3a: podpira neomejene hierarhije; - Nivo 3b: podpira omejene hierarhije.
4.	Simetrični RBAC	Poleg vseh funkcionalnosti omejenega RBAC: - mora podpirati pregled relacij med vlogami in pravicami, performančno primerljivo s pregledom relacij med uporabniki in vlogami; - Nivo 4a: podpira neomejene hierarhije; - Nivo 4b: podpira omejene hierarhije.

Tabela 2-3: Funkcionalnosti posameznih nivojev RBAC po modelu NIST [4]

[1] Pozitivne pravice so tiste pravice, ki uporabnikom dovoljujejo oz. omogočajo dostop, negativne pa tiste, ki uporabnikom onemogočajo dostop.

2.4.2.1 Linearni RBAC

Linearni RBAC predstavlja osnovni pogled na RBAC. Osnovna zahteva linearnega modela je, da je številnost relacij uporabnik-vloga in pravica-vloga zastavljena kot mnogo proti mnogo, kakor prikazuje Slika 2-6. Posledica tega je, da imajo lahko uporabniki več vlog, vloga pa ima lahko več pravic hkrati, pri tem pa se ne izključuje drugih načinov, s katerimi lahko uporabnik pridobi pravice, kot so npr. neposredne dodelitve vlog. Druga zahteva modela pa je, da mora sistem, ki implementira RBAC, nuditi pregled nad relacijami uporabnik-vloga, torej pregled kateri uporabniki imajo katere vloge.



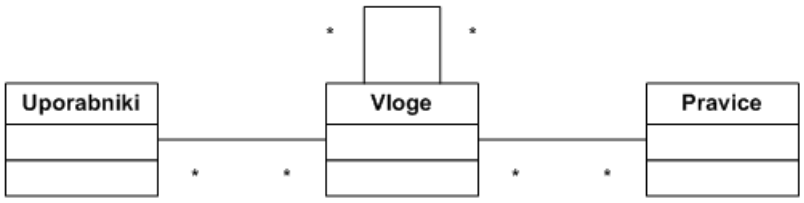
Slika 2-6: Linearni model RBAC

V primeru iz resničnega sveta si lahko predstavljamo uporabnika kot zaposleno osebo, vlogo kot delovno mesto, pravico pa kot dostop do nekega sistema. V tem primeru RBAC poskrbi, da zaposleni pridobi ustrezne dostope do sistemov, ki so opredeljeni za njegovo delovno mesto.

2.4.2.2 Hierarhični RBAC

Hierarhični RBAC, ki ga prikazuje Slika 2-7, se glede na linearni RBAC razlikuje le po uvedbi hierarhije vlog. Hierarhije vlog so naravni način za strukturiranje vlog, glede na organizacijsko strukturo avtoritet in odgovornosti. Matematično gledano so hierarhije delno urejene množice, kar pomeni, da vsebujejo refleksivne, tranzitivne in antisimetrične relacije. Veliko produktov pa nudi podporo le omejenim hierarhijam, ki pa vseeno nudijo bistveno boljše zmogljivosti glede na linearni model. Hierarhični model zato pozna dva podnivoja:

- **Splošni hierarhični RBAC**: podpira neomejene delno urejene množice, ki opisujejo hierarhijo vlog.
- **Omejeni hierarhični RBAC**: glede na splošno hierarhijo vsebuje omejitve v strukturi hierarhije vlog. V večini primerov so hierarhije omejene na preprosta drevesa [4].



Slika 2-7: Hierarhični model RBAC

2.4.2.3 Omejen RBAC

Omejen RBAC hierarhičnemu modelu doda omejitve. Omejitve so lahko statične in se navezujejo na relacije med uporabniki in vlogami, ali dinamične, ki se navezujejo na aktivacijo vlog znotraj uporabnikove seje [4]. Več o omejitvah in razdelitvah nalog pa je napisano v poglavju 2.4.3.

2.4.2.4 Simetrični RBAC

Simetrični RBAC se navezuje na reševanje problema nadzora nad relacijami med vlogami in pravicami. Ključni problem je identifikacija dodeljenih pravic vlogam (in posledično uporabnikom), ki so postale neprimerne ali zastarele. Simetrični RBAC tako nadgrajuje omejen RBAC s pogledom, ki služi pregledovanju pravic za določeno vlogo ali uporabnika. S tem pogledom lahko organizacija identificira vse potekle vloge uporabnika ali pa vse pravice, ki jih je potrebno odstraniti ob odhodu uporabnika iz organizacije.

Simetrični RBAC pa lahko vsebuje tako dodatne poglede, ki prikazujejo direktno in indirektno[1] dodeljene pravice, kot tudi preglede z možnostjo izbora posameznega sistema [4].

2.4.3 Ločevanje nalog

Za ločevanje nalog se pogosto uporablja kratica SoD (angl. *Separation of Duties* ali *Segregation of Duties*). SoD se nanaša na razdelitev opravil in povezanih pravic med različne vloge, z namenom preprečevanja, da bi uporabnik dobil preveliko avtoriteto. Cilj SoD je zmanjševanje možnosti prevar in večjih napak, to pa dosežemo z uvedbo ustrezne razporeditve nalog različnim uporabnikom. SoD je tako za mnoga podjetja in informacijske sisteme ključna varnostna zahteva.

NIST model RBAC opredeljuje dva tipa SoD:

- **Statični SoD ali SSD**: definira omejitve za relacije med uporabnikom in vlogo na nivoju dodelitve vloge uporabniku. To pomeni, da če je uporabniku dodeljena ena

[1] Direktno dodeljene pravice so pravice, ki so neposredno dodeljene uporabnikom, indirektne pa tiste pravice, ki jih uporabnik dobi preko hierarhije vlog.

vloga, se mu ne sme dodeliti druge vloge, če imata vlogi med seboj definirano omejitev. Omejitve med vlogami se ohranjajo skozi celotno hierarhijo oz. se dedujejo.

- **Dinamični SoD ali DSD**: definira omejitve med vlogami, katere lahko uporabnik uporablja simultano, na nivoju aktivacije vlog. Za razliko od statičnega SoD, sta uporabniku lahko dodeljeni obe vlogi, vendar mu dinamični SoD preprečuje, da bi obe vlogi lahko uporabil hkrati [4].

V domeno IdM sistemov lahko postavimo statični SoD, saj praviloma IdM sistemi nimajo nadzora nad aktivacijami vlog (kdaj kakšen uporabnik uporablja vlogo). Ko oz. če je to potrebno, je to v domeni ciljnega sistema.

Mnogi IdM sistemi pa poleg osnovne logike ločevanja nalog nudijo tudi proces razreševanja konfliktov vlog, ki so nastali na podlagi pravil ločevanja nalog. Konflikt vlog nastane v primeru dodelitve dveh ali več vlog uporabniku, s katerimi se ustvari problem pri razdelitvi nalog ali pa pride do konflikta interesov. Koncept konfliktov znotraj SoD je sestavljen iz dveh postopkov:

- **Zaznavanje konfliktov na podlagi pravil**: je tipično implementirano kot razširitev RBAC, lahko pa je implementirano kot samostojen modul. Pravila SoD se praviloma določajo preko uporabniškega vmesnika sistema in lahko temeljijo na podlagi vnaprej definiranih kombinacij pravic ali na podlagi vlog uporabnika v organizaciji. Slednji način je uporabnikom bolj intuitiven, vendar pa je njegova implementacija odvisna od samih definicij vlog v organizaciji. Problem pri vnaprej definiranih kombinacijah pravic pa nastane z velikim številom pravic, saj je teh kombinacij lahko tudi več tisoč, kar je skoraj nemogoče nadzorovati oz. vzdrževati. IdM sistemi lahko ponujajo tudi druge načine definiranja pravil konfliktov med vlogami, mnogi pa omogočajo tudi programsko razširitev pravil in s tem omogočijo organizacijam definicijo lastnih pravil in algoritmov za določanje konfliktov.
- **Reševanje konfliktov**: je običajno proces, ki se sproži ob zaznavanju konflikta. Način reševanja konfliktov je odvisen od samega IdM sistema, saj se v nekaterih sistemih zgolj pošljejo obvestila o konfliktu, drugi pa izvedejo zahtevo za razrešitev konflikta, kjer mora odgovorna oseba za uporabnika izbrati eno ali nobene od konfliktnih vlog.

Tipični primeri konfliktov vlog:

- Ena vloga zahteva podroben dostop in vsebuje pozitivne pravice, druga pa preprečuje ta dostop in vsebuje negativne pravice.
- Dve vlogi omogočata uporabniku izvajanje nalog, katere ne smejo biti združene npr. uporabnik lahko zahteva in potrjuje dostop.

2.4.4 Poročila skladnosti

Poročila skladnosti (angl. *compliance reports*) so običajno centralizirana znotraj enega sistema ter omogočajo celovit pregled nad identitetami in njihovimi dostopi. S temi poročili se lahko v organizaciji ustrezno preverja, ali so vsi dostopi skladni z zahtevano varnostno politiko. Naprednejša poročila, poleg prikaza dostopov identitet, omogočajo tudi prikaz oseb, ki so te dostope odobrile, zgodovinske poglede (spremembe dostopov, stanje dostopov na določen dan), prikaz glede na sisteme oz. vloge itd. Vedno večje težnje po uvedbi poročil skladnosti v organizaciji so posledica vedno višjih varnostnih zahtev in zahtev po skladnosti z zakonskimi iniciativami oz. standardi (npr. Sarbanes-Oxley, EuroSox, HIPAA, Basel II), katerim podjetja sledijo.

2.4.5 Revizija dostopa

Revizija dostopa (angl. *attestation*) je periodičen proces revizije dostopov identitet. Revizija dostopa je tudi naslednji varnostni korak, ki sledi poročilom skladnosti. Poročila skladnosti nam namreč omogočajo prikaze trenutnih dostopov identitet, revizija dostopa pa ta poročila dopolnjuje s tem, da mora ustrezna avtoritativna oseba te dostope tudi potrditi oz. zavrniti – izvesti revizijo dostopov. Pri samem izvajanju procesa se pojavljajo ključna vprašanja:

- **Kaj se pregleduje**: dostope identitet, podatke identitet, SoD pravila …
- **Kdo pregleduje**: vodje identitet, interni revizorji, vodje oddelkov ...
- **Kdaj in kako pogosto**: sam interval se določi na nivoju organizacije in je običajno določen znotraj standardov, ki jih podjetja izvajajo. Intervali so lahko mesečni, kvartalni, polletni ali letni.

Proces revizije dostopa je lahko avtomatičen ali ročni, vendar pa se periodične ročne revizije dostopov v organizacijah običajno ne izvajajo, saj so časovno in stroškovno zelo potratne, ker je potrebno pridobiti in preveriti podatke iz vseh povezanih sistemov v celotni organizaciji, te podatke pa nato še revidirati.

Uporaba avtomatskega procesa revizije dostopov pa zagotavlja hitrejši in natančnejši pregled dostopov z nižjimi stroški. Proces tudi pomaga pri bolj efektivnem upravljanju celovitih

informacijskih sistemov, pri čemer se minimizira tveganja in dosega uspešne varnostne preglede. Izvedba avtomatskega procesa revizije dostopov poteka tako, da po izteku določenega obdobja revizor dobi nalogo oz. aktivnost procesa, v katerem IdM sistem avtomatsko pripravi podatke za revizijo, ki jo mora revizor opraviti. Običajno se proces izvaja na način, da revizor dobi aktivnost, kjer ima prikazane vse dostope identitet. Naloga revizorja pa je, da za vsak dostop določi, ali je identiteta upravičena do dostopa ali ne. Po izvedbi revizije, se identiteti vse neustrezne vloge takoj odvzame, revizor pa dobi pripravljeno poročilo oz. poročilo opravljene revizije.

2.4.6 Eskalacija

Eskalacija je varnostni mehanizem, ki zagotavlja, da se zahteve uporabnikov ne izvajajo predolgo oz. se ne »izgubljajo«. Namen eskalacije je, da se zahteva, ki čaka na potrditev vodje (ali katere druge osebe), po določen času avtomatsko dodeli drugemu vodji oz. drugi osebi. Eskalacija je lahko eno- ali večnivojska. Nivoji eskalacije določajo, kolikokrat se lahko aktivnost eskalira. Najbolj pogost primer uporabe eskalacije so neplanirane odsotnosti zaposlenih.

Običajno je eskalacija že vgrajena v ogrodje delovnih tokov procesov in jo je za uporabo potrebno le omogočiti, pri tem pa definirati maksimalen čas izvajanja aktivnosti in uporabnika, kateremu se ob eskalaciji aktivnost dodeli. Pogosto je to nadrejena oseba osebi, kateri je bila aktivnost dodeljena.

2.4.7 Delegacija

Delegacija je komplementarna funkcionalnost eskalaciji – če je eskalacija uporabljena za primer neplaniranih odsotnosti zaposlenih, se delegacija uporablja za planirane odsotnosti zaposlenih. Seveda pa se delegacija ne uporablja le za planirane odsotnosti zaposlenih, ampak tudi v drugih primerih, kot je npr. začasna delitev nalog zaradi preobremenjenosti uporabnika.

Namen delegacije je, da lahko uporabnik svoje dostope ali pravice delegira drugemu uporabniku za določen čas. S tem ko uporabnik delegira vse ali samo del svojih vlog drugemu uporabniku, lahko ta nato izvaja ustrezne aktivnosti uporabnika, ki mu je delegiral vloge. Z delegacijo se ohranja časovno odvisnost sistema (npr. potrjevanje zahtev), hkrati pa dopušča, da se aktivnosti preverjanja skladnosti in revizij neprekinjeno izvajajo v skladu z varnostno politiko.

2.4.8 Podpora poslovnim procesom

Ena izmed funkcionalnosti, ki je za uspešnost IdM sistema ključnega pomena, je podpora poslovnim procesom organizacije. Ker se organizacije med seboj močno razlikujejo, je z enim ali nekaj standardnimi procesi nemogoče pokriti vse potrebe posamezne organizacije. Z uvedbo različnih mednarodnih standardov in predpisov se lahko določeni procesi standardizirajo, vendar pa se ne morejo pokriti vse zahteve organizacije, saj se marsikje pojavljajo bolj kompleksni procesi ali izjeme v procesih, ki jih morajo IdM sistemi podpreti. Tako običajno IdM sistemi nudijo možnost implementacije novih procesov, brez posegov v sam IdM sistem. Te procese se običajno realizira preko grafičnega vmesnika za urejanje procesov in na ta način se lahko dopolnjuje standardne, preddefinirane procese, lahko pa gre za čisto samostojne procese. Z možnostjo izdelave lastnih procesov se tako zagotovi, da se lahko pokrije številne zahteve organizacij, brez potrebe po spreminjanju samega IdM sistema.

2.5 Načini razprševanja

Načini sinhronizacije predstavljajo načine razprševanja podatkov. Tukaj je torej zajeta tako izdelava kot vzdrževanje ter odstranjevanje identitet in ostalih podatkov. Poznamo dva načina razpršitve:

- ročna razpršitev,
- avtomatska razpršitev.

2.5.1 Ročna razpršitev

Ročna razpršitev je značilna za sisteme, ki niso neposredno povezani z IdM sistemom. V tem primeru IdM sistem vsebuje podatke o identitetah, vlogah in dodelitvah vlog sistema, vendar se ti podatki med samima sistemoma ne sinhronizirajo oz. prenašajo samodejno. Tipičen primer ročne razpršitve je, da se po potrditvi zahteve dodeli nalogo skrbniku sistema, da ustrezne spremembe ročno izvede tudi na ciljnem sistemu. Pri tem je seveda velika odgovornost skrbnikov sistema, da dosledno in pravilno izvajajo spremembe, tako v IdM kot v ciljnem sistemu.

Namen upravljanja dostopa do nepovezanega sistema z ročno razpršitvijo podatkov je ta, da IdM sistem nudi uporabo procesov izdelave zahtev, sledljivost izdelave in potrjevanja zahtev ter pregled nad dostopi identitet.

2.5.2 Avtomatska razpršitev

Avtomatsko razpršitev se uporablja za sisteme, ki so neposredno povezani z IdM sistemom. To pomeni, da se spremembe narejene v IdM sistemu samodejno izvedejo tudi na ciljnem sistemu. Izvedba se lahko izvede neposredno iz IdM sistema ali pa z avtomatsko sinhronizacijo preko metaimenikov. Ta način razprševanja glede na ročno razprševanje nudi:

- manjšo verjetnost napak pri izvedbi sprememb na ciljnem sistemu,
- hitrejši prenos spremembe v ali iz ciljnih sistemov,
- večjo skladnost podatkov med IdM in ciljnim sistemom, saj lahko sčasoma pri ročni razpršitvi pride do razlik med IdM in ciljnim sistemom, če skrbniki ne izvajajo sprememb dosledno.

Kljub vsem zgoraj naštetim prednostim, pa se ročna razpršitev še vedno uporablja, ker ima cenejšo in hitrejšo implementacijo. Seveda pa je ročna razpršitev smiselna le za sisteme, kjer se ne izvaja veliko sprememb, saj se z veliko količino sprememb preveč obremeni skrbnike sistema.

2.6 ITIL

Information Technology Infrastructure Library oz. ITIL predstavlja zbirko napotkov in smernic za upravljanje in uvajanje storitev informacijskih tehnologij. Gre tudi v svetovnem merilu za danes najbolj razširjen pristop k upravljanju IT storitev.

ITIL je konec 80-ih let razvila britanska vladna agencija CCTA (*Central Computing and Telecommunications Agency*). Leta 2001 je izšla druga verzija, imenovana ITIL v2, zadnja različica pa se imenuje ITIL v3 in je bila izdana leta 2007. Zadnja verzija ITIL, za razliko od prejšnjih, predstavlja smernice skozi življenjski cikel IT storitev, ta pa vsebuje pet faz. Vsaka izmed faz pa je opisana v posamezni knjigi, ki so:

1. Strategija razvoja (angl. *Service Strategy*),
2. Načrtovanje storitev (angl. *Service Design*),
3. Tranzicija storitev (angl. *Service Transition*),
4. Izvajanje storitev (angl. *Service Operation*),
5. Nenehno izboljševanje storitev (angl. *Continual Service Improvement*) [15].

V nadaljnjih podpoglavjih se bom osredotočil na proces upravljanja dostopa (angl. *Access Management*), ki je opisan v knjigi »ITIL Service Operation«, saj se ta proces navezuje na

obravnavano tematiko – sisteme za upravljanje z identitetami. Knjiga spada v četrto fazo življenjskega cikla ITIL storitev in polega procesa upravljanja dostopa vsebuje tudi naslednje procese:

- upravljanje z dogodki (angl. *Event Management*),
- upravljanje z incidenti (angl. *Incident Management*),
- upravljanje s problemi (angl. *Problem Management*),
- izpolnjevanje zahtev (angl. *Request Fulfillment*) [8].

Namen teh procesov je učinkovito in uspešno zagotavljanje IT storitev.

2.6.1 Proces upravljanja dostopov

Pojem proces upravljanja dostopov je bil prvič predstavljen v ITIL v3 in je opredeljen kot »proces odobritve avtoriziranim uporabnikom pravice do uporabe storitve, medtem ko neavtoriziranim uporabnikom preprečuje dostop« [8]. Pogosto pa se v različnih organizacijah namesto izraza upravljanje dostopov uporabljata izraza upravljanje s pravicami (*Rights Management*) in upravljanje identitet (*Identity Management*).

Kot lahko vidimo iz zgoraj navedene definicije, se ta ujema z opredelitvijo IdM po M. Bergmanu in J. Cooperju. Tudi poslovna vrednost, osnovni koncepti in aktivnosti procesa upravljanja dostopov, ki jih ITIL opisuje, lahko najdemo v večini današnjih IdM sistemov, katere nudijo različni proizvajalci programske opreme.

Za lažje zagotavljanje ustreznih dostopov in pravic, proces upravljanja dostopa uporablja oz. vključuje katalog vseh vlog v organizaciji in storitve, ki jih te vloge podpirajo. Ta katalog izdela in vzdržuje sistem za upravljanje dostopa skupaj s preostalimi sistemi. Pogosto pa se to avtomatizira z uporabo imeniških storitev.

Proces upravljanja dostopov je tesno povezan s procesoma upravljanja varnosti in razpoložljivosti. Procesa upravljanja varnosti in razpoložljivosti sta v primerjavi s procesom upravljanja dostopov namenjena planiranju – vzpostavljanju pravil, postopkov in navodil – proces upravljanja dostopov pa izvajanju teh pravil in postopkov.

Ključni gonilnik procesa upravljanja dostopov je upravljanje informacijske varnosti, ki nudi varnostna pravila, pravila varovanja podatkov in orodja za uspešno izvajanje procesa upravljanja dostopov [8].

2.6.1.1 Osnovni koncepti

Upravljanje dostopa je proces, ki omogoča uporabnikom uporabo storitev opredeljenih v katalogu storitev. Sestavljen je iz naslednjih osnovnih konceptov:

- **Dostop** se navezuje na nivo in obseg funkcionalnosti ali podatkov storitve, do katerih je uporabnik upravičen.
- **Identiteta** se nanaša na informacije o osebah ali napravah, s katerimi se medsebojno razlikujejo kot individualne in s katerimi se preverja njihov status v organizaciji. Po definiciji je identiteta uporabnika enolična za tega uporabnika.
- **Pravica** se nanaša na dejansko nastavitev, pri čemer je uporabniku dodeljen dostop do storitve ali skupine storitev. Tipične pravice ali nivoji dostopa vsebujejo branje, pisanje, zaganjanje, spreminjanje in brisanje.
- **Storitve ali skupine storitev**: za lažje upravljanje z dostopi in pravicami se lahko uvede skupine storitev, kjer se namesto določanja uporabniku dostopa do posamezne storitve, uporabi skupine storitev in se tako uporabniku z enim korakom dodeli dostop ali pravice več storitev hkrati.
- **Imeniške storitve** se nanašajo na specifičen tip orodja, ki upravlja z dostopom in pravicami [8].

Zgoraj navedene osnovne koncepte, uporabljene v procesu za upravljanje dostopov, lahko povežemo s ključnimi entitetami IdM sistemov, katere prikazuje Slika 2-3. V koncept vloge lahko vključimo dostope, pravice in storitve, v koncept sistema pa lahko vključimo storitve ali skupine storitev in imeniške storitve. Kot vidimo lahko storitve ali skupine storitev obravnavamo kot sistem ali vlogo. To je odvisno od same implementacije storitve, saj je storitev lahko samostojen sistem, ki ima svoje dostope in pravice, lahko pa je uporaba storitev implementirana preko članstva v skupinah imeniških storitev.

2.6.1.2 Aktivnosti procesa

Proces upravljanja dostopov svetuje uporabo naslednjih aktivnosti:

- **Zahtevanje dostopa**: sistem mora nuditi možnost za zahtevanje dostopa. Dostop se običajno zahteva z zahtevami, ki so generirane preko kadrovskega sistema, s samopostrežbo, z zahtevo za spremembo itd. Pravila za zahtevanje dostopa so običajno dokumentirana v katalogu storitev.

- **Preverjanje zahteve za dostop** zagotavlja, da je uporabnik, ki je zahteval dostop, res tista oseba za katero se predstavlja in da je uporabnik upravičen do zahtevane storitve. Prva kategorija preverjanja se običajno izvrši s tem, da mora uporabnik vnesti uporabniško ime in geslo ali pa se predstaviti na kakšen drug način (npr. biometrični podatki, elektronski ključi, šifrirne naprave …). Druga kategorija pa se zagotavlja s preverjanjem, ki je neodvisno od uporabnikove zahteve. Ta preverjanja so lahko odobritev ustreznega vodje, oddaja zahteve preko storitvenega centra ali avtomatsko preverjanje preko pravila, ki določa, da ima uporabnik lahko dostop do dodatnih storitev, če jih potrebuje.
- **Zagotavljanje pravic**: proces upravljanja dostopa ne določa kdo ima pravice do katerih IT storitev, ampak le uveljavlja sprejete odločitve glede zagotavljanja ali omejevanja dostopa. Torej, ko je uporabnikova zahteva preverjena, se sproži nova zahteva za izvedbo spremembe dostopa pri vsaki ekipi oz. oddelku, ki je vključen v podporo storitve (ročna razpršitev), ali pa je celoten postopek avtomatiziran (avtomatska razpršitev). Pri zagotavljanju pravic pa lahko pride tudi do konfliktov vlog. Konflikti vlog so podrobneje opisani v poglavju 2.4.3. Poleg tega bi moralo upravljanje dostopa vključevati tudi redne preglede vlog in skupin, da se zagotovi ustreznost IT skupinam. To preverjanje je podrobnejše opisano v poglavju 2.4.5.
- **Spremljanje statusa identitet**: aktivnost se nanaša na življenjski cikel identitete. Aktivnost narekuje, da se mora sistem ustrezno odzvati na spremembe statusa ali podatkov identitet (npr. sprememba delovnega mesta, napredovanja oz. nazadovanja, upokojitve …) in če je le možno avtomatizirati proces za vsak tip identitete.
- **Pisanje dnevnika in spremljanje dostopa** je aktivnost, ki zagotavlja, da so zagotovljene pravice tudi ustrezno uporabljene. Spremljanje in nadzor dostopa morata biti vključena v vse funkcije operativnih storitev, torej tudi v proces upravljanja z dostopi.
- **Odstranjevanje oz. omejevanje dostopa**: upravljanje dostopa poleg zagotavljanja pravic vključuje tudi preklic teh pravic. Podobno kot pri zagotavljanju pravic, sistem tukaj ne odloča, temveč le izvršuje odločitve in pravila.

2.6.1.3 Metrike

Metrike, ki se jih uporablja za merjenje učinkovitosti upravljanja dostopov, so lahko:

- število zahtev za dostop;

- število odobrenih zahtev;
- število incidentov, ki zahtevajo ponastavitev pravic za dostop;
- število incidentov, ki so posledica neustreznih nastavitev za dostope.

2.6.1.4 Kritični dejavniki uspeha

ITIL za proces upravljanja dostopov opredeljuje tudi kritične dejavnike uspeha, ki so:

- Možnost preverjanja identitete uporabnika (da je oseba res tista za katero se predstavlja).
- Možnost preverjanja identitete odobritelja.
- Možnost preverjanja, ali je uporabnik upravičen do dostopa do specifične storitve.
- Možnost dodeljevanja več pravic in dostopov individualnim uporabnikom.
- Možnost določanja statusa uporabnika v določenem trenutku.
- Možnost upravljanja sprememb glede na uporabniške zahteve po dostopih.
- Možnost omejevanja dostopa neavtoriziranim uporabnikom.
- Podatkovna zbirka vseh uporabnikov in pravic, ki so jim bile odobrene [8].

2.7 CobiT

CobiT ali *Control Objectives for Information and related Technology*, podobno kot ITIL, predstavlja množico dobrih praks in procesno ogrodje za upravljanje z IT-jem. CobiT so razvili v IT Governance Institute, do danes pa so bile izdane štiri glavne verzije. Prva izdaja je bila leta 1996, zadnja oz. četrta pa leta 2005, pri čemer je bila ta izdaja leta 2007 dopolnjena in objavljena kot verzija 4.1. Trenutno je v pripravi CobiT verzija 5, vendar datum objave do danes še ni znan [11].

Komponente CobiT-a verzije 4.1 so razdeljene na štiri IT področja in 34 IT procesov. Ta področja so:

- planiranje in organizacija ali PO (angl. *Planning and Organization*),
- pridobitev in uvedba ali AI (angl. *Acquisition and Implementation*),
- predaja in podpora ali DS (angl. *Delivery and Support*) ter
- spremljanje in vrednotenje ali ME (angl. *Monitoring and Evaluation*).

Posamezni procesi znotraj področij pa so sestavljeni iz opisa procesa, kontrolnih ciljev, smernic upravljanja in zrelostnega modela [5].

V diplomski nalogi sem se osredotočil na proces, ki se nanaša na področje IdM sistemov in spada v IT področje predaja in podpora oz. DS. Proces se imenuje zagotavljanje varnosti z oznako DS5 in je podrobneje opisan v naslednjem poglavju.

2.7.1 Proces zagotavljanja varnosti sistemov

Proces pripada sklopu upravljanja s tveganji. Vsebuje vzpostavitev ter vzdrževanje IT varnostnih vlog in zadolžitev, pravil, standardov in postopkov. Poleg tega pa vključuje tudi izvajanje nadzora varnosti, periodično testiranje in implementacijo popravkov za identificirane varnostne pomanjkljivosti ali incidente [5].

Proces opredeljuje ključne varnostne aktivnosti za izvajanje in zagotavljanje varnosti v celotni organizaciji. To nam pove, da proces pokriva veliko širše področje od področja, katerega pokriva IdM, vendar pa naloge IdM sistema lahko povežemo z več aktivnostmi procesa, ki so opredeljene v naslednjem poglavju.

2.7.1.1 Aktivnosti procesa

Proces zagotavljanja varnosti sistemov opredeljuje naslednje aktivnosti:

- **Upravljanje IT varnosti**: upravljanje IT varnosti naj se izvaja na čim višjem organizacijskem nivoju, tako da je upravljanje z varnostjo v koraku s poslovnimi zahtevami.
- **Planiranje IT varnosti**: v plan IT varnosti naj se vključi področje poslovanja, tveganja in zahteve po skladnosti, pri čemer naj se upošteva tudi IT infrastrukturo in varnostno kulturo. Plan naj bo tudi ustrezno implementiran v varnostno politiko.
- **Upravljanje z identitetami**: zagotoviti je potrebno, da so vsi uporabniki in njihove aktivnosti v IT sistemih enolično definirane. Poskrbi naj se, da so pravice uporabnikov do sistemov in podatkov usklajene z opredeljenimi in dokumentiranimi poslovnimi zahtevami, hkrati pa naj bodo zahteve delovnih mest vezane na identitete uporabnikov. Poleg tega naj se zagotovi, da se pravice dostopa uporabnika zahtevajo pri vodstvu uporabnika, potrjujejo na nivoju skrbnikov sistemov in da te zahteve implementira odgovorna oseba za varnost.
- **Upravljanje z uporabniškimi računi**: postopki upravljanja z uporabniškimi računi naj se uporabljajo za vse uporabniške račune, tudi administratorske in tiste z višjim nivojem pravic. Postopki naj vsebujejo potrditev sprememb s strani uporabnika ali

potrditev s strani lastnika sistema. Poleg tega pa naj se izvajajo tudi redni pregledi vseh uporabniških računov in z njimi povezanih pravic.

- **Testiranje, spremljanje in nadzor varnosti**: implementacijo IT varnosti naj se ustrezno testira in nadzira. Funkcije pisanja dnevnika in nadziranja bodo namreč omogočile preventivo ali zaznavanje nenavadnih in neustreznih aktivnosti.
- **Definicija varnostnih incidentov**: ustrezno naj se definira potencialne varnostne incidente, da bo možno pred njimi obravnavati preko procesa upravljanja z incidenti in problemi.
- **Varovanje varnostnih tehnologij**: tehnologija povezana z varnostjo naj bo odporna na prirejanje, prav tako naj se ne razkriva dokumentov o varnosti po nepotrebnem.
- **Upravljanje s šifrirnimi ključi**: opredeli naj se pravila ter postopke za izdelavo, spremembe, odvzeme, distribucijo, uporabo, hrambo in arhiviranje kriptografskih ključev pred spremembami in razkritjem.
- **Preventiva in zaznavanje zlonamernih programov ter popravki**: v celotni organizaciji naj se izvaja ustrezno politiko varnosti, s katero se zagotavlja varnost pred zlonamernimi programi.
- **Omrežna varnost**: izvaja naj se ustrezne upravljavske postopke, s katerimi se zagotavlja ustrezne dostope in nadzor nad podatki med različnimi omrežji, npr. požarni zidovi, razdelitev omrežij itd.
- **Izmenjava občutljivih podatkov**: izmenjava občutljivih podatkov naj se izvaja samo preko varnih povezav ali pa preko medija, ki zagotavlja avtentičnost podatka [5].

2.7.1.2 Cilji

CobiT opredeljuje cilje za zagotavljanje varnosti na treh nivojih:

- Na nivoju celotnega IT-ja:
    - zagotavljanje, da imajo samo prave osebe dostop do kritičnih in tajnih podatkov,
    - zagotavljanje zaupanja v avtomatizirane transakcije in izmenjavo informacij,
    - vzdrževanje integritete informacij in procesne infrastrukture,
    - odgovornost in varovanje vseh IT sredstev ter
    - zagotavljanje, da se IT storitve in infrastruktura lahko ubranijo in hitro okrevajo v primeru napak, napadov in nesreč.

- Na nivoju procesa:
    - dovoljevanje dostopanja do občutljivih podatkov zgolj avtoriziranim uporabnikom,
    - identifikacija, nadzor in poročanje o varnostnih pomanjkljivostih in incidentih,
    - zaznavanje in razreševanje neavtoriziranega dostopa do informacij in aplikacij ter
    - minimizacija vpliva varnostnih pomanjkljivosti in incidentov.
- Na nivoju aktivnosti:
    - razumevanje varnostnih zahtev, pomanjkljivosti in nevarnosti,
    - standardizirano upravljanje uporabniških identitet in avtorizacij,
    - definiranje varnostnih incidentov in
    - redno testiranje varnosti [5].

Če primerjamo prednosti in cilje IdM sistemov, lahko naslednje cilje za zagotavljanje varnosti, ki jih opredeljuje CobiT, neposredno povežemo z IdM sistemi:

- zagotavljanje, da imajo samo prave osebe dostop do kritičnih in tajnih podatkov,
- dovoljevanje dostopanja do občutljivih podatkov zgolj avtoriziranim uporabnikom,
- zaznavanje in razreševanje neavtoriziranega dostopa do informacij in aplikacij ter
- standardizirano upravljanje uporabniških identitet in avtorizacij.

Čeprav se lahko z IdM sistemi neposredno poveže le manjši del ciljev, pa to še ne pomeni, da IdM sistemi ostalih ciljev ne dosegajo. Določeni IdM sistemi lahko nekatere preostale cilje dosegajo z dodatnimi funkcionalnostmi ali pa jih dosegajo posredno oz. s pomočjo drugih sistemov.

2.7.1.3 Metrike

Podobno kot cilje, CobiT opredeljuje tudi metrike na treh nivojih:

- Na nivoju IT:
    - število incidentov z vplivom na poslovanje,
    - število sistemov, ki ne dosegajo varnostnih zahtev ter
    - čas za dodeljevanje, vzdrževanje ter odstranjevanje dostopa in pravic.
- Na nivoju procesa:
    - število in vrste osumljenih in dejanskih kršitev dostopa,

- število kršitev pri ločevanju nalog,
- število uporabnikov, katerih gesla ne ustrezajo standardom ter
- število in vrste preprečenih zlonamernih kod.

- Na nivoju aktivnosti:
  - pogostost pregledov nadziranih vrst varnostnih dogodkov,
  - število in vrste odvečnih uporabniških računov,
  - število zavrnjenih neavtoriziranih IP naslovov, vrat in vrst prometa,
  - odstotek ogroženih in razveljavljenih kriptografskih ključev ter
  - število avtoriziranih, odvzetih, ponastavljenih ali spremenjenih pravic oz. dostopov [5].

2.7.1.4 Zrelostni model

Poleg lastnosti, aktivnosti, ciljev in metrik procesa zagotavljanja varnosti, pa CobiT opredeljuje tudi zrelostni model zagotavljanja varnosti. Zrelostni model obsega šest nivojev, ki so oštevilčeni od 0 do 5. Ti nivoji so:

0. **Neobstoječe**: organizacija ne prepoznava potrebe po IT varnosti. Odgovornosti in zadolženosti za zagotavljanje varnosti niso razdeljene ter ukrepi za podpiranje IT varnosti niso implementirani. Poročil IT varnosti ni, prav tako pa ni predvidenega procesa za odzive na kršitve IT varnosti.
1. **Začetno**: organizacija prepozna potrebo po IT varnosti, vendar pa je zavedanje pomembnosti varnosti odvisno od posameznika. IT varnost ni merjena, prav tako se ob zaznavanju kršitev IT varnosti prelaga odgovornost, saj zadolženosti niso opredeljene. Odzivi na kršitve IT varnosti so nepredvidljivi.
2. **Ponovljivo, vendar intuitivno**: odgovornosti in zadolženosti za IT varnost so dodeljene varnostnim koordinatorjem, pri čemer je avtoriteta vodstva koordinatorja omejena. Zavedanje pomembnosti varnosti je porazdeljeno in omejeno. Čeprav sistemi ponujajo informacije povezane z varnostjo, se teh ne analizira. Varnostna pravila se razvijajo, vendar pa so znanja in orodja neustrezna. Poročila varnosti so nepopolna, zavajajoča ali neprimerna. Varnostna usposabljanja so na voljo, vendar se izvajajo le na podlagi samoiniciative posameznika. IT varnost se obravnava kot odgovornost, ki je v domeni IT-ja, medtem ko poslovno področje ne vidi IT varnosti kot del svoje domene.

3. **Opredeljeno**: zavedanje pomembnosti varnosti obstaja in ga vodstvo spodbuja. Varnostni postopki so opredeljeni in usklajeni z varnostno politiko. Odgovornosti za IT varnost so dodeljene in razumljene, vendar se ne izvršujejo konsistentno. Varnostni plani in rešitve obstajajo ter so vodeni na podlagi analize tveganja. Poročila o varnosti niso v celoti osredotočena na poslovanje. Varnostno usposabljanje je na voljo za IT in poslovno področje, vendar se ga izvaja in upravlja neformalno.
4. **Vodeno in merljivo**: odgovornosti za IT varnost so jasno dodeljene, vodene in izvajane. Analiza varnostnega tveganja in vpliva se konsistentno izvaja. Varnostne politike in postopki se izvajajo s specifičnimi varnostnimi izhodišči. Identifikacija, avtentifikacija in avtorizacija uporabnikov je standardizirana. Testiranje varnosti se izvaja z uporabo standardnih in formaliziranih procesov, ki vodijo k izpopolnjevanju varnostnega nivoja. Poročila IT varnosti so povezana s poslovnimi cilji. Varnostno usposabljanje se izvaja tako na poslovnem kot IT področju. Cilji in metrike upravljanja varnosti so definirane, vendar se ne merijo.
5. **Optimizirano**: IT varnost je skupna odgovornost tako IT kot poslovnega področja. Zahteve IT varnosti so jasno opredeljene, optimizirane in vključene v odobren varnostni plan. Varnostni incidenti so takoj obravnavani s postopkom formaliziranih incidentov, ki je podprt z avtomatskimi orodji. Izvajajo se periodične varnostne ocene, s katerimi se ocenjuje efektivnost implementacije varnostnega plana. Informacije o grožnjah in ranljivostih se sistematično zbirajo in analizirajo. Varnostni procesi in tehnologije so integrirane v celotno organizacijo. Metrike upravljanja varnosti se merijo, zbirajo in ustrezno sporočajo, vodstvo pa ta merila uporablja za prilagajanje varnostnega plana in nenehno izpopolnjevanje procesa [5].

Glede na dane zrelostne modele lahko v današnjem času večino organizacij uvrstimo nekje med drugi in tretji zrelostni nivo. Z uvedbo IdM sistema v organizacijo pa lahko ta preide v tretji ali četrti zrelostni nivo, vendar je IdM sistem samo orodje, ki omogoča, da se v organizaciji dvigne varnostni nivo. Za dejansko umestitev v višji nivo pa je potrebno v sami organizaciji izvesti ustrezne korake za doseganje višjega nivoja. Tukaj gre predvsem za zavedanje pomembnosti varnosti ter uveljavljanje varnostne politike s strani vodstva in poslovnega področja organizacije.

2.8 Microsoft Forefront Identity Manager 2010

Microsoft Forefront Identity Manager 2010 ali FIM je Microsoftov sistem tretje generacije produktov za upravljanje z identitetami. Pred uradno objavo sistema se je imenoval ILM 2. Poleg upravljanja z identitetami omogoča tudi upravljanje z digitalnimi certifikati, vendar se bom v diplomski nalogi osredotočil le na del za upravljanje z identitetami.

2.8.1 Zgodovina

Kot rečeno je FIM tretja generacija Microsoftovih sistemov za upravljanje z identitetami, njegova predhodnika, ki ju predstavim v nadaljevanju, pa sta Microsoft Identity Integration Server 2003 ali MIIS 2003 in Identity Lifecycle Management 2007 ali ILM 2007.

2.8.1.1 Microsoft Identity Integration Server 2003

Microsoft Identity Integration Server 2003 ali MIIS predstavlja prvo generacijo Microsoftovih sistemov za upravljanje z identitetami in je implementiran kot metaimenik. Kot omenjeno v poglavju 2.2.3.2, MIIS predstavlja enovit pogled nad identitetami in preostalimi podatki. Identitete in ostale podatke med sistemi pridobiva in ustvarja s sinhronizacijami, ki jih izvajajo upravljavski agenti oz. MA (angl. *Management Agent*). Podrobnejši opis delovanja metaimenikov in upravljavskih agentov vsebuje poglavje 2.8.2.3, saj je celoten koncept in način sinhronizacije ostal enak vse do tretje generacije sistemov.

2.8.1.2 Identity Lifecycle Management 2007

Identity Lifecycle Management 2007 predstavlja drugo generacijo Microsoftovih sistemov za upravljanje z identitetami. Celoten sistem temelji na MIIS 2003, h kateremu dodaja funkcionalnost upravljanja s certifikati. Celoten koncept metaimenika in sinhronizacij pa ostaja enak kot v MIIS 2003.

2.8.2 Opis sistema

FIM temelji na ILM 2007 (posledično tudi na MIIS 2003), kjer jima je skupno upravljanje s certifikati in pametnimi karticami, metaimenik ter kreiranje in vzdrževanje uporabnikov. Poleg skupnih funkcionalnosti z ILM 2007, pa FIM uvaja naslednja področja oz. funkcionalnosti [7]:

- upravljanje s pravili dostopov,
- upravljanje s poverilnicami,
- upravljanje z uporabniki in
- upravljanje s skupinami.

Izmed ključnih funkcionalnosti IdM sistemov, naštetih v poglavju 2.4, FIM vsebuje:

- samopostrežbo,
- upravljanje dostopov,
- nadzor dostopa na podlagi vloge (linearni RBAC),
- eskalacijo in
- podporo poslovnim procesom.

Izmed obravnavanih funkcionalnosti IdM sistemov pa FIM ne podpira poročil skladnosti, ločevanja nalog, revizije dostopov in delegacije. Vsako od teh funkcionalnosti se v FIM lahko dodatno implementira, saj je celoten sistem razširljiv. Lahko pa se namesto lastne implementacije izbere tudi tretjega ponudnika, ki nudi module za razširitev funkcionalnosti, katerih FIM privzeto ne podpira. Med te module sodi tudi zmogljivejša različica RBAC, ki podpira ločevanje nalog, in drugi napredni moduli.

Celoten sistem FIM sestavljajo štirje produkti, vsak izmed njih pa je podrobneje opisan v naslednjih podpoglavjih:

- FIM Portal,
- FIM Service,
- FIM Synchronization Service in
- FIM Certificate Management.

Poleg zgoraj navedenih štirih produktov pa FIM vsebuje tudi dodatka namenjena odjemalcem. To sta *Password Reset Add-In* in *Office Outlook FIM Add-In*. Prvi omogoča anonimnim uporabnikom, da si ponastavijo geslo. Dodatek za ponastavitev gesla je integriran z Windows operacijskimi sistemi in je tako na voljo kot dodatna možnost na zaslonu za prijavo v sistem. Drugi dodatek pa je namenjen vsem uporabnikom aplikacije Office Outlook (poštni odjemalec orodja Microsoft Office), da lahko v samem poštnem odjemalcu potrjujejo ali zavračajo njim dodeljene zahteve za odobritev.

2.8.2.1 FIM Portal

FIM Portal je spletni vmesnik FIM Servicea, kar pomeni da se vse akcije sprožene preko FIM Portala izvedejo v FIM Serviceu. Spletni vmesnik temelji na Windows Sharepoint Services 3.0 ali WSS 3.0. Tako kot sam WSS 3.0 je tudi FIM Portal razširljiv uporabniški vmesnik,

kjer se lahko dodaja nove procese, poglede, spletne gradnike, prav tako pa se lahko spremeni celostno grafično podobo portala in se jo tako priredi potrebam organizacije.

Že ob namestitvi FIM Portala uporabniški vmesnik vsebuje naslednje funkcionalnosti:

- Samopostrežba za članstvo v varnostnih in distribucijskih skupinah. Pri kreiranju zahtev za dostop se izvede tudi proces enonivojskega potrjevanja s strani lastnika skupine.
- Upravljanje varnostnih in distribucijskih skupin.
- Upravljanje z identitetami.
- Upravljanje z zahtevami samopostrežbe.
- Prijava na funkcionalnost ponastavitve gesla.

2.8.2.2 FIM Service

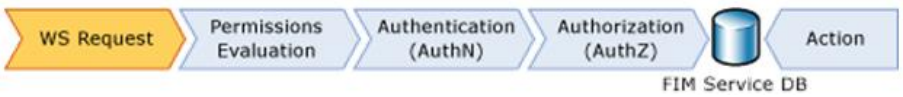
FIM Service je spletna storitev, ki skrbi za procesiranje zahtev in izvajanje delovnih tokov. Vsaka sprememba se v sistemu FIM Service obravnava kot zahteva. Zahteve se tako lahko kreirajo na podlagi akcij uporabnika preko FIM Portala, sinhronizacije podatkov s FIM Synchronization Service ali pa preko spletnih storitev, ki jih uporabljajo druge aplikacije. S tem, ko se vsaka sprememba v FIM Serviceu obravnava kot zahteva in se vsako zahtevo tudi shrani, sistem ponuja nadzor in zgodovino sprememb vseh podatkov v FIM Serviceu.

Vsaka oddana zahteva za spremembo je v FIM Serviceu obravnavana in se zanjo preveri, ali ima sprožitelj pravice za izvedbo zahteve. Pravice za izvedbo zahteve se odobrijo na podlagi upravljavskih pravil ali MPR-ja (angl. *Management Policy Rule*). Vsi dostopi do objektov v FIM Service se nastavljajo preko upravljavskih pravil, kajti brez ustrezno nastavljenega upravljavskega pravila, ki opredeljuje dostop, uporabnik nima pravic za dostop do objekta. Zato FIM Service privzeto vsebuje kar nekaj upravljavskih pravil, s katerimi je že vzpostavljen osnovni varnostni model. Poleg namena določanja dostopov, pa se upravljavska pravila lahko uporabi tudi za sprožanje delovnih tokov. Za izvedbo delovnih tokov skrbi ogrodje Windows Workflow Fundation [7].

Procesiranje zahtev se izvaja v naslednjem zaporedju, ki ga prikazuje tudi Slika 2-8:

1. Zahteva se odda preko spletne storitve (*WS Request*).
2. Na podlagi definiranih upravljavskih pravil se preveri, ali ima uporabnik ustrezne pravice za izvedbo zahteve (*Permission Evaluation*).

3. Upravljavsko pravilo sproža opredeljene delovne tokove v danem zaporedju (glede na vrsto delovnega toka):
    a. Aventifikacijske delovne tokove (*Authentication Workflow*): avtentifikacijski delovni tokovi se uporabljajo za zagotavljanje, da je uporabnik tista oseba za katero se predstavlja. Običajno se te delovne tokove uporablja pri ponastavljanju gesla, kjer mora uporabnik z odgovori na različna osebna vprašanja dokazati, da je res prava oseba.
    b. Avtorizacijske delovne tokove (*Authorization Workflow*): avtorizacijski delovni tokovi se uporabljajo za naprednejše odobritve ali zavrnitve glede na vsebino zahteve. Ročne odobritve vodij oz. lastnikov skupin so implementirane z avtorizacijskimi delovnimi tokovi.
    c. Akcijske delovne tokove (*Action Workflow*): ti delovni tokovi pa so namenjeni spreminjanju vrednosti atributov v podatkovni bazi FIM Servicea. Pred izvedbo akcijskih delovnih tokov, se morajo uspešno izvesti vsi definirani aventifikacijski in avtorizacijski delovni tokovi.



Slika 2-8: Procesiranje zahtev v FIM Serviceu [7]

FIM Service, poleg entitet s katerimi upravlja (identitete, varnostne in distribucijske skupine …), uporablja tudi objekte s katerimi je mogoče samo upravljanje teh entitet. Tukaj bi izpostavil predvsem tri med seboj tesno povezane objekte – prej omenjene delovne tokove in upravljavska pravila ter množice (angl. *Set*). Množica je nabor entitet oz. virov znotraj FIM Servicea na podlagi pravil (npr. vse zaposlene osebe, neaktivni uporabniki itd.). Vsi trije objekti pa so med seboj povezani s tem, da upravljavsko pravilo:

- določeni skupini uporabnikov dovoljuje ustrezne akcije[1] nad entitetami, ki so del opredeljene množice;
- ali na podlagi določenih akcij[1] nad entitetami, ki so del opredeljene množice, izvaja delovne tokove.

Poleg teh objektov pa se v diplomski nalogi pojavljata predvsem še dva tipa objektov, ostalih, ki pa jih v okviru diplomske naloge nisem uporabljal, pa ne bom podrobneje opisoval:

[1] Možne akcije so branje, pisanje, spreminjanje ali brisanje entitet ali posameznih atributov entitet.

- Iskalna področja (angl. *Search Scope*): iskalna področja so namenjena prikazovanju ustreznega nabora podatkov v pogledih objektov preko FIM Portala. Iskalno področje se opredeli z določitvijo objektov, ki naj jih iskalno področje prikazuje, poleg tega pa se lahko določi tudi filter s pomočjo XPath sintakse.
- Sinhronizacijska pravila (angl. *Synchronization Rule*): sinhronizacijska pravila so namenjena opredeljevanju preslikav atributov pri sinhronizaciji podatkov v FIM Synchronization Service. Sinhronizacijska pravila nadomeščajo opredeljevanje preslikav podatkov v upravljavskih agentih FIM Synchronization Servicea in s tem uvajajo funkcionalnost imenovano *Codeless Provisioning*. Ta funkcionalnost omogoča, da se objekte pri sinhronizacijah med sistemi lahko prenaša in kreira brez dodatne programske kode [7].

2.8.2.3 FIM Synchronization Service

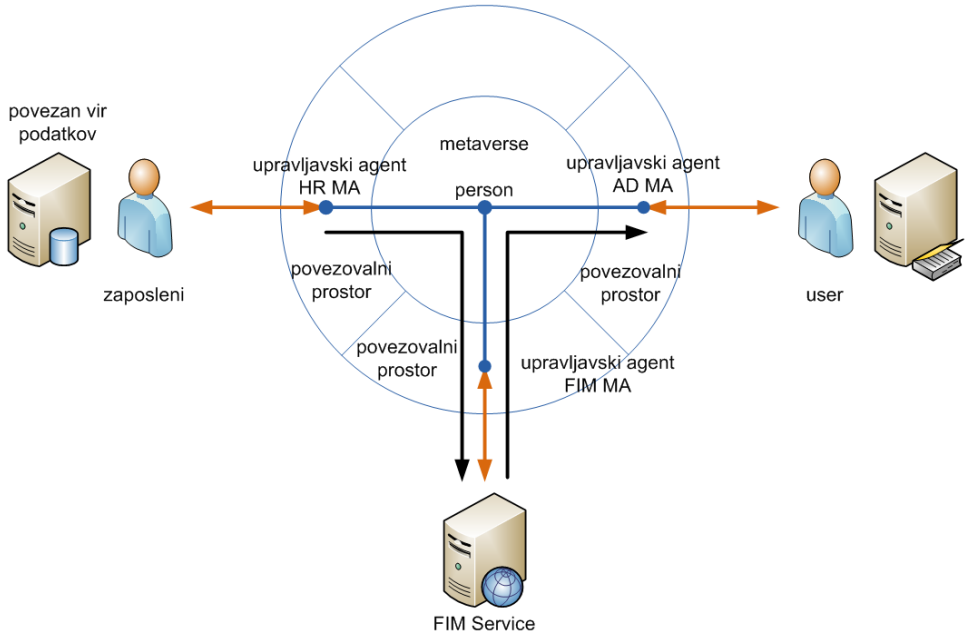
FIM Synchronization Service je metaimenik, ki komunicira in izmenjuje podatke med različnimi hrambami identitet oz. povezanimi viri podatkov preko adapterjev, ki se imenujejo upravljavski agenti. FIM Synchronization Service je poleg FIM Certificate Managerja edini podsistem FIM-a, ki je bil prisoten že v ILM [7].

FIM Synchronization Service sestavljajo naslednje ključne komponente, katere prikazuje tudi Slika 2-9:

- **Upravljavski agent** je orodje s katerim se prenaša podatke iz povezanih podatkovnih virov v povezovalni prostor. Upravljavski agenti so del FIM Synchronization Servicea in jih ni potrebno nameščati na sisteme podatkovnih virov. Upravljavski agenti prenašajo podatke v dveh korakih – najprej se podatki prenesejo v povezovalni prostor upravljavskega agenta, v drugem koraku pa se podatki iz povezovalnega prostora sinhronizirajo s podatki v metaversu. Poleg samega uvoza podatkov in sinhronizacije pa upravljavski agenti omogočajo tudi izvoz podatkov v povezan podatkovni vir. Upravljavski agent je na sliki prikazan v zunanjem kolobarju kot modra pika.
- **Povezovalni prostor** (angl. *Connector Space*) je lokalna kopija podatkov pridobljenih iz povezanih podatkovnih virov. Namen prenosa podatkov iz povezanih sistemov v povezovalni prostor je ta, da lahko sinhronizacijska storitev primerja trenutno stanje podatkov s prejšnjim in pri izvedbi sinhronizacije ne obremenjuje podatkovnega vira. Vsak upravljavski agent ima svoj povezovalni prostor. Na sliki posamezen del

zunanjega kolobarja predstavlja povezovalni prostor posameznega povezanega vira podatkov.

- **Metaverse** je osrednja hramba metaimenika in hrani skupna stanja identitet vseh povezanih podatkovnih virov. Metaverse je na sliki prikazan kot osrednji krog in je stičišče vseh povezanih podatkovnih virov [7].



Slika 2-9: Komponente in tok podatkov FIM Synchronization Service [7]

Poleg samih komponent Slika 2-9 prikazuje tudi tok podatkov in preslikave entitet. Prikazano je kako se entiteta zaposlenega preko upravljavskega agenta HR MA prenese v povezovalni prostor upravljavskega agenta in nato sinhronizira z entiteto *person*. Upravljavski agent FIM MA prenese entiteto *person* v FIM Service. FIM Service nad to entiteto izvede vsa pravila in delovne tokove, FIM Synchronization Service pa nato novo stanje entitete prenese preko upravljavskega agenta FIM MA v povezovalni prostor upravljavskega agenta in sinhronizira z obstoječo entiteto *person* v metaversu. V zadnjem koraku upravljavski agent AD MA poskrbi, da se entiteta *person* ustrezno preslika v entiteto *user*, katera pa predstavlja uporabniški račun v imeniški hrambi.

2.8.2.4 FIM Certificate Management

FIM Certificate Management je sistem namenjen upravljanju s digitalnimi certifikati in pametnimi karticami. Sistem sestavljajo lastna podatkovna zbirka, ki hrani delovne tokove in informacije o digitalnih certifikatih, portal, ki služi kot uporabniški vmesnik, ter dodatni moduli, katere se namesti na sistem digitalnega overitelja.

2.8.3 Skladnost FIM z IT smernicami

V poglavju bom opisal skladnost FIM z današnjimi standardi oz. smernicami ITIL-a in CobiT-a. FIM je v celoti razširljiv sistem in se lahko vanj naknadno implementira dodatne funkcionalnosti, s katerimi se lahko realizira določene aktivnosti in funkcionalnosti, ki jih narekujejo standardi, vendar se bom osredotočil le na funkcionalnosti, ki jih FIM privzeto nudi po namestitvi sistema.

2.8.3.1 Skladnost s smernicami ITIL-a

Aktivnosti procesa upravljanja z dostopi, ki jih opredeljuje ITIL in so v sistemu FIM podprte, so:

- **Zahtevanje dostopa**: sistem FIM v celoti podpira aktivnost zahtevanja dostopa, saj uporabnikom omogoča samopostrežbo za zahtevanje članstva tako v varnostnih kot v distribucijskih skupinah.
- **Preverjanje zahteve za dostop**: tudi to aktivnost FIM v celoti podpira, saj se mora vsak uporabnik prijaviti s svojim domenskim uporabniškim imenom in geslom, prav tako pa mora vodja potrditi vsako zahtevo za dostop, ki jo zahteva uporabnik.
- **Zagotavljanje pravic**: FIM vsebuje podsistem FIM Synchronization Service, ki omogoča avtomatsko izvedbo potrjenih zahtev za dostop. Ker pa ne vsebuje funkcionalnosti ločevanja nalog in revizije dostopov, pa to aktivnost le delno podpira.
- **Spremljanje statusa identitet**: FIM se odziva na vse spremembe identitet, vendar le v omejenem obsegu. Spremembo statusa oz. veljavnosti polno podpira, saj posledično omogoči ali onemogoči dostop do dodeljenih vlog. Spremembe delovnega mesta, napredovanja ali nazadovanja pa običajno IdM sistemi rešujejo z uvedbo RBAC, katerega pa FIM nudi le v najenostavnejši obliki.
- **Pisanje dnevnika in spremljanje dostopa**: FIM spremlja in hrani vsako spremembo podatkov izvedeno znotraj FIM in s tem nudi sledljivost nad spremembami podatkov. Samo spremljanje dostopa s posameznimi vlogami pa je realizirano v imeniških storitvah oz. v ciljnih sistemih.
- **Odstranjevanje oz. omejevanje dostopa**: FIM po preteku veljavnosti ali spremembi statusa (neaktiven, upokojen itd.) ustrezno odstrani uporabniško ime in z njim vse povezane dostope. FIM Synchronization Service nato poskrbi, da se te spremembe avtomatsko prenesejo v ciljne sisteme.

2.8.3.2 Skladnost s smernicami CobiT-a

Aktivnosti procesa zagotavljanja varnosti, ki jih opredeljuje CobiT in so v sistemu FIM podprte v celoti, so:

- **Upravljanje z identitetami**: FIM s podsistemom FIM Synchronization Service skrbi, da imajo uporabniki ustrezen dostop do vseh povezanih sistemov. Podpora poslovnim procesom pa zagotavlja, da so ti dostopi skladni z dokumentiranimi poslovnimi zahtevami. FIM omogoča tudi vezavo zahtev delovnih mest na identitete uporabnika z uporabo RBAC. Sam proces za zahtevo dostopa poteka tako, da uporabniki zahtevajo dostop preko samopostrežbe, njihovo zahtevo odobri vodja in potrjena zahteva se avtomatsko prenese v ciljne sisteme. Potrditev zahteve s strani lastnika sistema pa se lahko dodatno konfigurira.
- **Upravljanje z uporabniškimi računi**: FIM vsebuje vse postopke za upravljanje z uporabniškimi računi (kreiranje, spreminjanje, odstranjevanje itd.). Pri tem je kreiranje uporabniških računov preko kadrovskega sistema običajno avtomatizirano, kreiranje neposredno v FIM pa omogočeno samo določenim uporabnikom z ustreznimi pravicami.

Aktivnosti, ki jih FIM delno podpira:

- **Testiranje, spremljanje in nadzor varnosti**: FIM omogoča spremljanje varnosti s tem, da hrani identitete in njihove dostope na enem mestu. Samo spremljanje varnosti pa se mora izvesti ročno, saj ne vsebuje poročil skladnosti.
- **Izmenjava občutljivih podatkov**: FIM neposredno ne implementira te funkcionalnosti, vendar pa omogoča komunikacijo preko varnih povezav (https).
- **Upravljanje s šifrirnimi ključi**: sistem FIM vsebuje tudi funkcionalnost upravljanja s certifikati, ki pa se v celoti uveljavlja le skupaj z overiteljsko storitvijo (angl. *Certification Authority Service*).

FIM pa ne podpira naslednji funkcionalnosti, kateri se običajno izvaja s specializiranimi programi:

- **Preventiva in zaznavanje zlonamernih programov ter popravki**.
- **Omrežna varnost**.

Poleg zgoraj omenjenih aktivnosti FIM omogoča **upravljanje IT varnosti** na višjem organizacijskem nivoju, saj vsebuje preprost uporabniški vmesnik za upravljanje z identitetami in dostopi. FIM podpira tudi rezultate aktivnosti **planiranja IT varnosti**, ker izvaja opredeljeno varnostno politiko in pravila.

Aktivnosti **definicija varnostnih incidentov** in **varovanje varnostnih tehnologij** pa nista v veliki meri povezani z implementacijo IdM sistemov in se tako ne nanašata na področje, ki ga FIM pokriva.

Ključna pomanjkljivost FIM-a pri upravljanju z identitetami, uporabniškimi računi in pri upravljanju IT varnosti, so manjkajoča poročila skladnosti, saj ta omogočajo prikazovanje metrik, lažje merjenje uspešnosti sistema ter lažji nadzor nad identitetami in njihovimi vlogami.

3 Uvedba sistema v podjetju

V poglavju bom podrobno predstavil in opisal postopek uvedbe IdM sistema v večjem fiktivnem podjetju. Procesi temeljijo na resničnem primeru oz. dejanski uvedbi IdM sistema v večjem slovenskem podjetju, pri čemer sem podatke pripravil sam in celotno uvedbo sistema izvedel v lastno postavljenem okolju, namenjenemu diplomski nalogi. Okolje za diplomsko nalogo je sestavljeno iz dveh Windows Server 2008 R2 64-bitnih strežnikov:

- sistemi prvega strežnika: Microsoft Active Directory, Microsoft Exchange 2010,
- sistemi drugega strežnika: Microsoft SQL Sever 2008 R2, FIM 2010.

Za potrebe diplomske naloge sem izdelal enostavno podatkovno zbirko, ki predstavlja kadrovski sistem, v katerega sem vnesel približno 2800 zaposlenih. Hkrati sem vsem zaposlenim izdelal tudi uporabniške račune v AD-ju in poštne predale v Microsoft Exchangeu 2010. Poleg zaposlenih pa se izdelal tudi organizacijsko strukturo podjetja s 40-imi organizacijskimi enotami in približno 120-imi vlogami v AD-ju. Namen izdelave teh podatkov je bila vzpostavitev realnega okolja podjetja.

Nadaljnja podpoglavja opisujejo celoten postopek uvedbe IdM sistema – pregled sistemov, analizo poslovnih procesov in navsezadnje dejansko vpeljavo sistema, ki je prikazana od sinhronizacije podatkov v metaimeniku pa do implementacije poslovnih procesov, ki so zahtevani v podjetju.

3.1 Pregled sistemov v podjetju

Podjetje uporablja veliko število sistemov, kot so kadrovski sistem, sistemi za nadzorovanje proizvodnje, ERP sistemi, sistemi za nadzor nad dostopi v podjetju itd. Izmed vseh teh sistemov, se bo v prvi fazi v IdM sistem povezalo le tri ključne:

- kadrovski sistem,
- Microsoft Active Directory in
- Microsoft Exchange 2010.

S povezavo teh sistemov se bo v podjetju avtomatiziralo in standardiziralo večino postopkov, ki so se do sedaj izvajali ročno.

3.1.1 Kadrovski sistem

Celoten kadrovski sistem ali HRM (angl. *Human Resource Management*) je sestavljen iz dveh podsistemov – kadrovsko evidenco tujih zaposlenih obravnavano podjetje vodi v sistemu SAP/R3, evidenco zaposlenih v Sloveniji pa v Microsoft SQL Server podatkovni zbirki, na podlagi katere temelji celoten sistem za upravljanje z zaposlenimi in obračun plač. Poenostavljen entitetni model podatkovne zbirke kadrovskega sistema prikazuje Priloga A.

Evidence zunanjih sodelavcev se ne vodi v nobenem sistemu. Zunanji sodelavci imajo samo uporabniške račune v centralnem imeniku, za te račune pa skrbijo njihovi skrbniki (zaposleni v podjetju).

3.1.2 Microsoft Active Directory

Microsoft Active Directory ali AD je centralna imeniška storitev, kjer se hranijo uporabniški računi zaposlenih in zunanjih sodelavcev. Večina sistemov v podjetju uporablja uporabniške račune in avtentifikacijo preko AD-ja, so pa v podjetju tudi sistemi, ki imajo lastno infrastrukturo uporabnikov in njihovih dostopov.

3.1.3 Microsoft Exchange 2010

Obravnavano podjetje uporablja Microsoft Exchange 2010 kot centralni sistem za upravljanje in delo z elektronsko pošto.

3.2 Analiza poslovnih procesov

V podjetju izvedena analiza trenutnega stanja pri procesih kadrovanja in določanja dostopa je pokazala, da vsi zahtevki za dodeljevanje ali ukinjanje uporabniških računov potekajo v obliki papirnatih zahtevkov. Za namen uvedbe sistema za upravljanje z identitetami pa so bili z analizo določeni ključni procesi zaposlovanja in dodeljevanja dostopa, s katerimi se bodo avtomatizirali in sistematizirali postopki za vse zaposlene v podjetju. Ti procesi so opisani v sledečih podpoglavjih.

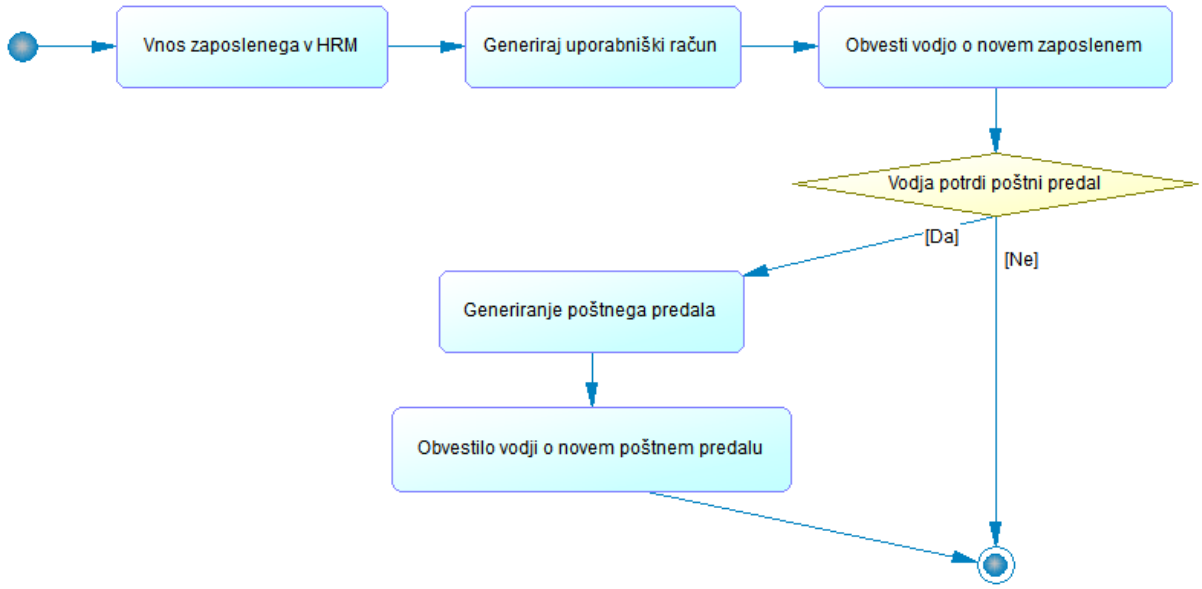
3.2.1 Zaposlitev nove osebe

Zaposlitev nove osebe poteka tako, da zaposleni v kadrovski službi vnesejo podatke o novi osebi v kadrovski sistem. FIM poskrbi za avtomatski prenos podatkov iz kadrovskega sistema v FIM, ob izdelavi nove identitete pa se morajo izvesti naslednje aktivnosti:

1. Določiti je potrebno unikatno uporabniško ime in generirati uporabniški račun.

2. Vodjo je potrebno obvestiti o novozaposlenem in mu posredovati podatke o novem uporabniškem imenu.
3. Vodji je treba dodeliti aktivnost, s katero določi, ali delavec potrebuje elektronski poštni predal.

Pri 3. aktivnosti je potrebno, v primeru potrditve elektronskega poštnega predala, zagotoviti, da se e-poštni naslov zapiše tudi v kadrovski sistem. Razen potrjevanja elektronskega poštnega predala, je celoten proces zaposlovanja avtomatiziran. Poleg samega procesa zaposlovanja pa je avtomatiziran tudi prenos sprememb podatkov o zaposlenem v kadrovski sistem. Proces zaposlovanja rednega delavca je prikazan v spodnji sliki (Slika 3-1).

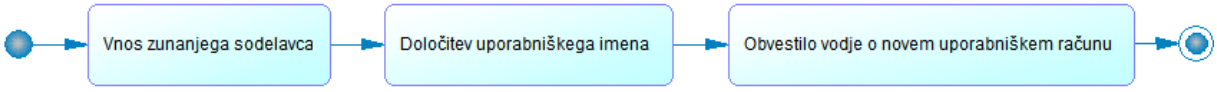


Slika 3-1: Proces zaposlitve nove osebe

3.2.2 Zaposlitev zunanjega sodelavca

Novega zunanjega sodelavca vodja ali zaposleni v kadrovski službi vnese v sistem za upravljanje z identitetami. Poleg podatkov o zunanjem sodelavcu, je obvezno potrebno vnesti tudi časovno obdobje v katerem bo lahko zunanji sodelavec imel dostop. Sistem nato poskrbi za določitev unikatnega uporabniškega imena in izdelavo uporabniškega računa. V zadnjem koraku procesa, sistem obvesti tistega, ki je podal zahtevo za nov uporabniški račun.

Proces zaposlitve zunanjega sodelavca prikazuje Slika 3-2.



Slika 3-2: Proces vnosa zunanjega sodelavca

3.2.3 Upravljanje s podatki zunanjega sodelavca

Proces upravljanja s podatki zunanjega sodelavca je zelo podoben procesu zaposlovanja zunanjega sodelavca, le da se tukaj spreminja njegove podatke in veljavnost dostopa. Podatkov o uporabniškem računu ni možno spreminjati. Upravljanje s podatki zunanjega zaposlenega lahko izvaja samo vodja identitete.

Proces upravljanja s podatki zunanjega zaposlenega prikazuje Slika 3-3.

Slika 3-3: Proces upravljanja s podatki zunanjega zaposlenega

Posebna pravila pa veljajo pri spremembah statusa zunanjega sodelavca . Razlog za posebna pravila je, da se podatke o zunanjih zaposlenih ročno vnaša in vzdržuje znotraj FIM ter v obravnavanem podjetju ne želijo, da bi se po preteku veljavnosti identitete zunanjega zaposlenega vsi podatki tudi odstranili. Za veliko večino zunanjih sodelavcev namreč velja, da imajo dostop do podjetja samo za omejena obdobja večkrat letno – torej samo takrat, kadar potrebujejo dostop. Če bi tako za zunanje sodelavce veljala enaka pravila kot za redne, bi morali ob vsakem ponovnem prihodu zunanjega sodelavca podatke tudi ročno vnesti v FIM. Da bi se izognili večkratnemu vnosu zunanjih sodelavcev, sta zanje definirani naslednji pravili:

- Uporabniški račun zunanjega zaposlenega je lahko aktiven samo v času veljavnosti identitet.
- Izven tega časa mora biti uporabniški račun onemogočen, vendar članstva v skupinah ostanejo nespremenjena.

3.2.4 Redni izstop zaposlenega

Redni izstop delavca predstavlja prenehanje delovnega razmerja zaposlenega v podjetju ob datumu poteka delovnega razmerja. V tem primeru se mora zaposlenemu onemogočiti uporabniški račun in odstraniti vsa članstva v vlogah. Proces rednega izstopa se mora izvesti ob datumu poteka delovnega razmerja, ta podatek pa je določen v HRM sistemu.

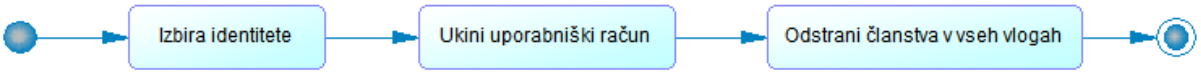
Proces prikazuje Slika 3-4.

Slika 3-4: Proces rednega izstopa zaposlenega

3.2.5 Takojšnji izstop zaposlenega

Takojšnji izstop delavca pomeni, da delavec preneha z delom v podjetju pred potekom pogodbe oz. datumom poteka delovnega razmerja. V tem primeru gre običajno za kršenje pogodbe in pravilnikov v podjetju ter je delavcu potrebno čim prej ukiniti dostop do sistema. Proces lahko izvedejo samo skrbniki IdM sistema.

Proces prikazuje Slika 3-5.



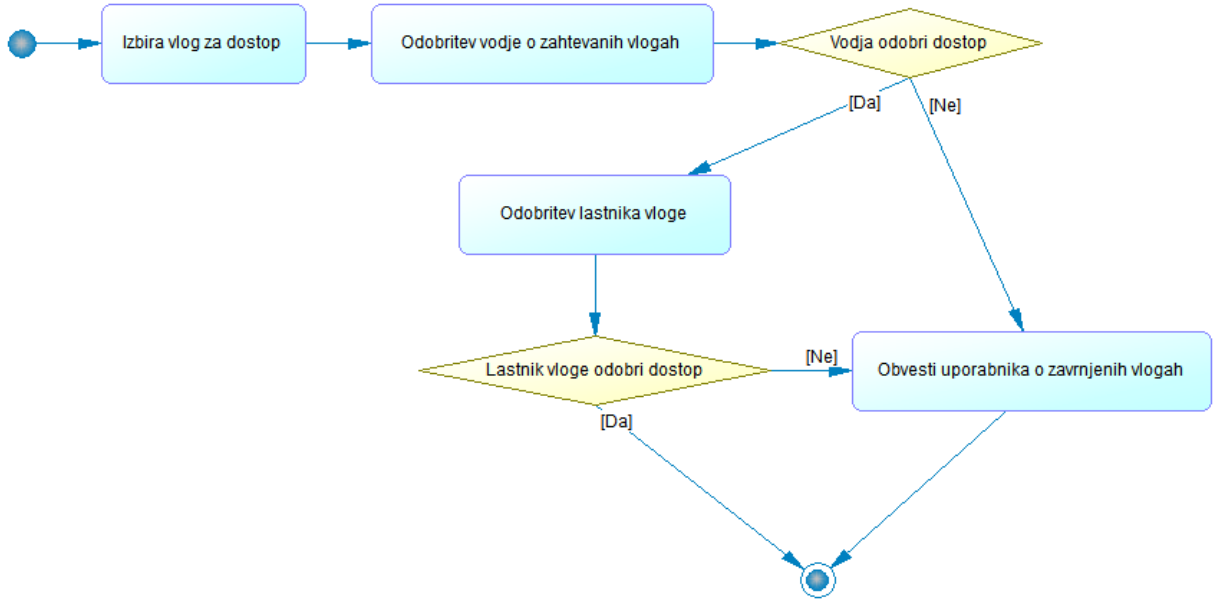
Slika 3-5: Proces takojšnjega izstopa zaposlenega

3.2.6 Zahteva za dodelitev dostopa

Proces za dodelitev dostopa se v sistemu za upravljanje z identitetami uvede kot samopostrežni zahtevek uporabnika za dostope do sistemov, ki jih potrebuje. Vsak zahtevek mora potrditi vodja zaposlenega in lastnik vloge. V tem primeru gre za dvonivojsko potrjevanje dostopov. Zahteve za dostop lahko izvajajo vsi zaposleni v podjetju.

Sam proces je namenjen določanju članstva v varnostnih skupinah AD-ja. V podjetju uporabljajo tudi distribucijske skupine AD-ja, proces za članstvo v teh skupinah se tudi izvaja s samopostrežbo, a le z enonivojskim potrjevanjem tj. potrjevanjem vodje.

Proces zahteve za dodelitev dostopa prikazuje Slika 3-6.



Slika 3-6: Proces zahteve za dostop

3.3 Vpeljava sistema Forefront Identity Manager 2010

Po opredelitvi procesov, sistemov in entitet, s katerimi bo FIM operiral, se lahko lotimo dejanske vpeljave sistema. Ključni koraki vpeljave so analiza obstoječih podatkov v kadrovskem sistemu in AD-ju, umestitev FIM v arhitekturno zasnovo podjetja, priprava razvojnega, testnega in produkcijskega okolja ter sama namestitev in konfiguracija FIM. Vsak od prej omenjenih korakov je opisan v nadaljnjih podpoglavjih, pri čemer pa sem izpustil namestitev FIM, ki vsebuje kar nekaj korakov, vendar je še vedno relativno enostavna.

3.3.1 Analiza podatkov

Z analizo podatkov se preveri dejansko stanje podatkov v sistemu, glede na opredeljene entitete in njihove atribute. Ključni izvedeni koraki pri analizi podatkov so:

- določitev enoličnega identifikatorja identitet in ostalih entitet,
- identifikacija manjkajočih in nepopolnih podatkov ter
- določitev avtoritativnega vira vsakega atributa entitet, ki se bodo sinhronizirale med sistemi.

Določitev enoličnega identifikatorja

Pri analizi podatkov je potrebno določiti enolični identifikator za vsako entiteto, katero se bo povezovalo med različnimi sistemu. Tabela 3-1 prikazuje opredeljene enolične identifikatorje za te entitete. Tukaj bi izpostavil predvsem entiteto zaposlenih, za katero je potrebno poskrbeti, da je enolični identifikator ustrezno napolnjen tako v kadrovskem sistemu kot v

AD-ju, saj se bodo na podlagi tega atributa združili podatki o zaposlenem iz kadrovskega sistema in podatki o uporabniškem računu iz AD-ja.

Entiteta	**Enolični identifikator**
Zaposleni	Kadrovska številka zaposlenega
Zunanji sodelavec	Uporabniško ime
Organizacijska enota	Identifikator org. enote
Varnostna skupina	Naziv varnostne skupine
Distribucijska skupina	Naziv varnostne skupine

Tabela 3-1: Enolični identifikatorji entitet

Identifikacija manjkajočih in nepopolnih podatkov

Ključna stvar pri identifikaciji manjkajočih in nepopolnih podatkov je opredelitev kateri atributi so za posamezno entiteto obvezni in kateri ne. Potrebno je poskrbeti, da so pred uvedbo FIM vsi obvezni atributi ustrezno napolnjeni, prav tako pa morajo biti napolnjeni vsi atributi, ki se jih bo uporabilo kot enolične identifikatorje. Pri preverjanju, ali so atributi ustrezno napolnjeni, je najbolje preveriti že v samem sistemu, saj če je že tam atribut obvezen, bo ta tudi ustrezno napolnjen ob sinhronizaciji. V nasprotnem primeru pa je potrebno te podatke ročno preveriti, npr. s poizvedbo po podatkovni zbirki, če gre za sistem relacijske podatkovne zbirke. Pri preverjanju enoličnega identifikatorja je potrebno paziti predvsem pri entiteti zaposlenih, če je kadrovska številka zaposlenega vnesena v vse uporabniške račune v AD-ju, saj lahko ob nepravilnih ali manjkajočih podatkih pride do napačnega uvoza podatkov v FIM ob inicialnem uvozu.

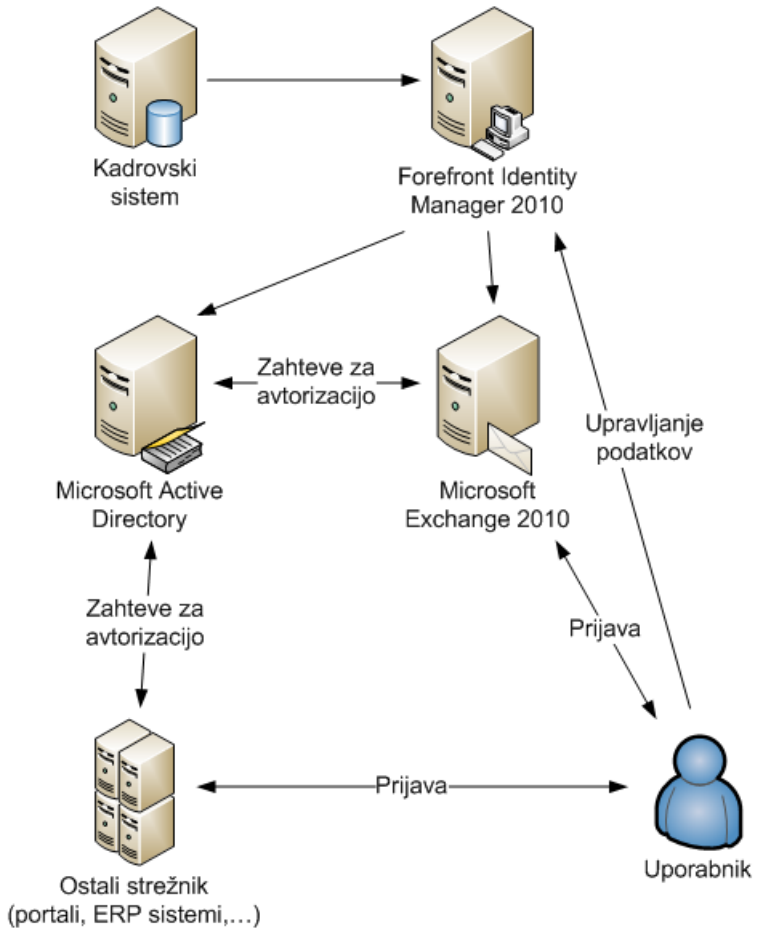
Določitev avtoritativnega vira atributov entitet

Za vsako entiteto in njene atribute, ki se bodo uporabljali preko sistema FIM, je potrebno določiti avtoritativni vir podatkov. Avtoritativni vir atributov entitet je potrebno določiti dvakrat – prvič za izvedbo inicialnega uvoza podatkov v FIM in drugič za izvedbo redne sinhronizacije. Tabela 3-4 prikazuje avtoritativne vire atributov za primer inicialnega uvoza, Tabela 3-5 pa prikazuje avtoritativne vire atributov za primer redne sinrhonizacije.

3.3.2 Arhitektura sistemov

Slika 3-7 prikazuje planirano arhitekturo sistemov v podjetju in interakcijo med sistemi po uvedbi FIM. Na sliki so izpostavljeni le tisti sistemi, ki neposredno komunicirajo ali vplivajo na FIM. Kot lahko vidimo, je FIM v arhitekturi sistemov umeščen med kadrovski sistem in

AD, kjer je poteka glavni tok podatkov iz kadrovskega sistema v FIM ter nato iz FIM v AD in posledično tudi v Microsoft Exchange 2010. Ostali strežniki in sistemi izvajajo avtentifikacijo in avtorizacijo preko AD-ja, uporabniki pa preko FIM upravljajo dostope in pravice povezane z AD-jem ter posledično tudi z ostalimi sistemi in Microsoft Exchangeom 2010.



Slika 3-7: Arhitektura sistemov

3.3.3 Priprava okolij za uvedbo sistema

Pri uvedbi IdM sistema je pomemben korak tudi vzpostavitev ustreznih okolij za uvedbo sistema. Minimalno je potrebno vzpostaviti vsaj tri okolja – razvojno, testno in produkcijsko:

- **Razvojno okolje** je namenjeno konfiguraciji ter razvoju procesov in funkcionalnosti, ki so predvidene v okviru uvedbe sistema. Okolje mora vsebovati vse ključne sisteme, s katerimi FIM komunicira, vendar pa so lahko vsi ti nameščeni na en sistem in lahko vsebujejo le minimalno množico podatkov. Na ta način se zagotovi enostavno in hitro okolje za razvoj, kjer je možno uporabljati vse povezane sisteme. Preden se vse spremembe prenese na testno okolje, je potrebno na razvojnem okolju izvesti tudi osnovno testiranje funkcionalnosti.

- **Testno okolje** je namenjeno testiranju funkcionalnosti in procesov s strani naročnika. Testno okolje mora biti za razliko od razvojnega čim bolj podobno produkcijskemu. To pomeni, da naj bodo sistemi postavljeni na enako konfiguriranih strežnikih kot v produkcijskem okolju, hkrati pa naj vsebujejo, če je le možno, tudi kopijo produkcijskih podatkov. S tem, ko je testni sistem bolj podoben produkcijskemu, je večja verjetnost, da se na produkciji ne bodo pojavljale napake zaradi različnih podatkov oz. različnih konfiguracij sistemov.
- **Produkcijsko okolje**: v produkcijskem okolju je potrebno namestiti le FIM. Po uspešnem izvedenem testu na testnem okolju je potrebno vse spremembe prenesti v produkcijsko okolje, izvesti inicialni uvoz in po uspešno izvedenem inicialnem uvozu izvesti nekaj ciklov redne sinhronizacije.

3.3.4 Konfiguracija FIM Service

Po izvedbi namestitve FIM Service in FIM Portal je potrebno sistem tudi ustrezno konfigurirati. Konfiguracija FIM Service in izdelava procesov se izvede preko FIM Portala ter vključuje pripravo entitetnega modela, izdelavo zahtevanih procesov (zaposlitev nove osebe, zaposlitev zunanjega sodelavca, upravljanje s podatki zunanjega sodelavca, redni izstop zaposlenega, takojšnji izstop zaposlenega, dodelitev dostopa) in ostalih funkcionalnosti. Posamezni koraki konfiguracije in izdelave procesov so podrobneje opisani v naslednjih podpoglavjih.

3.3.4.1 Priprava entitetnega modela

Z analizo poslovnih procesov in podatkov v obravnavanem podjetju so se opredelile entitete, ki se bodo uporabljale v IdM sistemu. Preden pa se prične implementacija procesov, je potrebno dopolniti entitetni model, katerega uporablja FIM Service. Tabela 3-2 prikazuje kako se splošne entitete preslikajo v FIM Service entitete in katere od teh entitet so privzeto že na voljo v FIM Service.

Splošna entiteta	**FIM Service entiteta**	**Obstaja v FIM Service**
Zaposleni	User	Da
Zunanji sodelavec	User	Da
Organizacijska enota	Organization unit	Ne
Distribucijska skupina	Group	Da
Varnostna skupina	Group	Da

Tabela 3-2: Preslikava splošnih entitet v FIM Service entitete

Kot je razvidno iz tabele se za zaposlene in zunanje sodelavce uporablja ista entiteta v FIM Service (*user*), prav tako se ena entiteta uporablja za varnostne in distribucijske skupine (*group*). Vendar pa se zaposleni od zunanjih sodelavcev ločijo po atributu *EmployeeType*, varnostne skupine od distribucijskih pa po atributu *Type*. Entiteti *user* in *group* privzeto vsebujeta že zelo velik nabor atributov in tako dodatnih atributov ni potrebno dodajati. FIM Service pa privzeto nima entitete za organizacijske enote, zato je potrebno entiteto in njene atribute dodati. Entiteto organizacijskih enot se v FIM Service doda z imenom *Organization unit*.

3.3.4.2 Izdelava procesa za zaposlitev nove osebe

Proces zaposlovanja nove osebe se začne z vnosom osebe v kadrovski sistem. FIM Synchronization Service nato z dnevnimi sinhronizacijami poskrbi, da se nova oseba doda v FIM Service. FIM Service pa ob dodajanju nove osebe na podlagi upravljavskega pravila:

1. Naredi zahtevo za potrditev poštnega predala in jo dodeli vodji.
2. V primeru potrditve zahteve generira nov enoličen poštni vzdevek za osebo.
3. Generira enolično ime za uporabniški račun osebe.

Izdelava zahteve za potrditev poštnega predala

FIM Service privzeto že vsebuje avtorizacijski delovni tok potrjevanja zahtev, vendar ga v tem primeru ne moremo uporabiti. Razlog je ta, da se ob zavrnitvi zahteve prekine celoten delovni tok – v primeru procesa zaposlitve nove osebe, bi se tako celoten postopek zaposlitve osebe prekinil. Posledično se nova oseba ne bi zapisala v FIM Service in tudi ne bi dobila ustreznih dostopov in računov. Tako je za potrebe procesa potrebno narediti nov avtorizacijski delovni tok potrjevanja zahteve, ki se ob zavrnitvi ne bo prekinil, ampak le ne bo generiral poštnega vzdevka. Delovni tok za potrjevanje poštnega predala, imenovan *Email Approval Workflow*, prikazuje Priloga E.

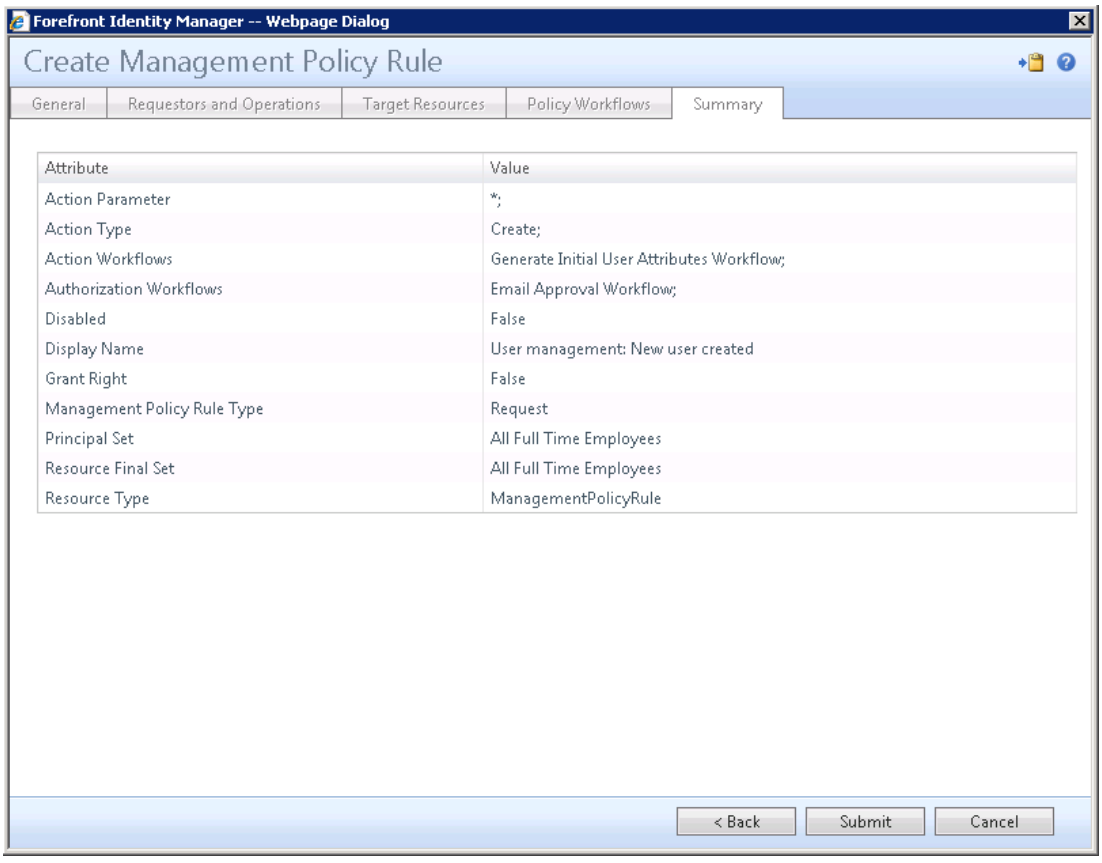
Generiranje enoličnega poštnega vzdevka in imena za uporabniški račun osebe

Za generiranje enoličnega poštnega vzdevka in imena za uporabniški račun osebe, je potrebno narediti nov akcijski delovni tok. Delovni tok mora pri generiranju poštnega vzdevka upoštevati odobritev oz. zavrnitev zahteve za poštni vzdevek. Torej, če je bila zahteva za poštni vzdevek odobrena, se poštni vzdevek generira, drugače pa se aktivnost generiranja poštnega vzdevka preskoči. Delovni tok, imenovan *Generate Initial Approval Workflow*,

prikazuje Priloga F, primer programske kode za izdelavo enoličnega poštnega vzdevka pa prikazuje Priloga G.

Opredelitev upravljavskega pravila za izvedbo procesa zaposlovanja nove osebe

Poleg zgoraj opisanih delovnih tokov pa se potrebuje še upravljavsko pravilo, ki bo ta delovna tokova tudi zagnalo. Upravljavsko pravilo opredelimo tako, da dodamo nov objekt tipa *Management Policy Rule*, v katerem nastavimo, da se za vsako novo identiteto vsebovano v množici *All Full Time Employees* (vsi redno zaposleni), zaženeta avtorizacijski delovni tok *Email Approval Workflow* za potrjevanje poštnega vzdevka ter akcijski delovni tok *Generate Initial User Attributes Workflow* za generiranje enoličnega poštnega vzdevka in imena za uporabniški račun. Primer upravljavskega pravila prikazuje Slika 3-8.

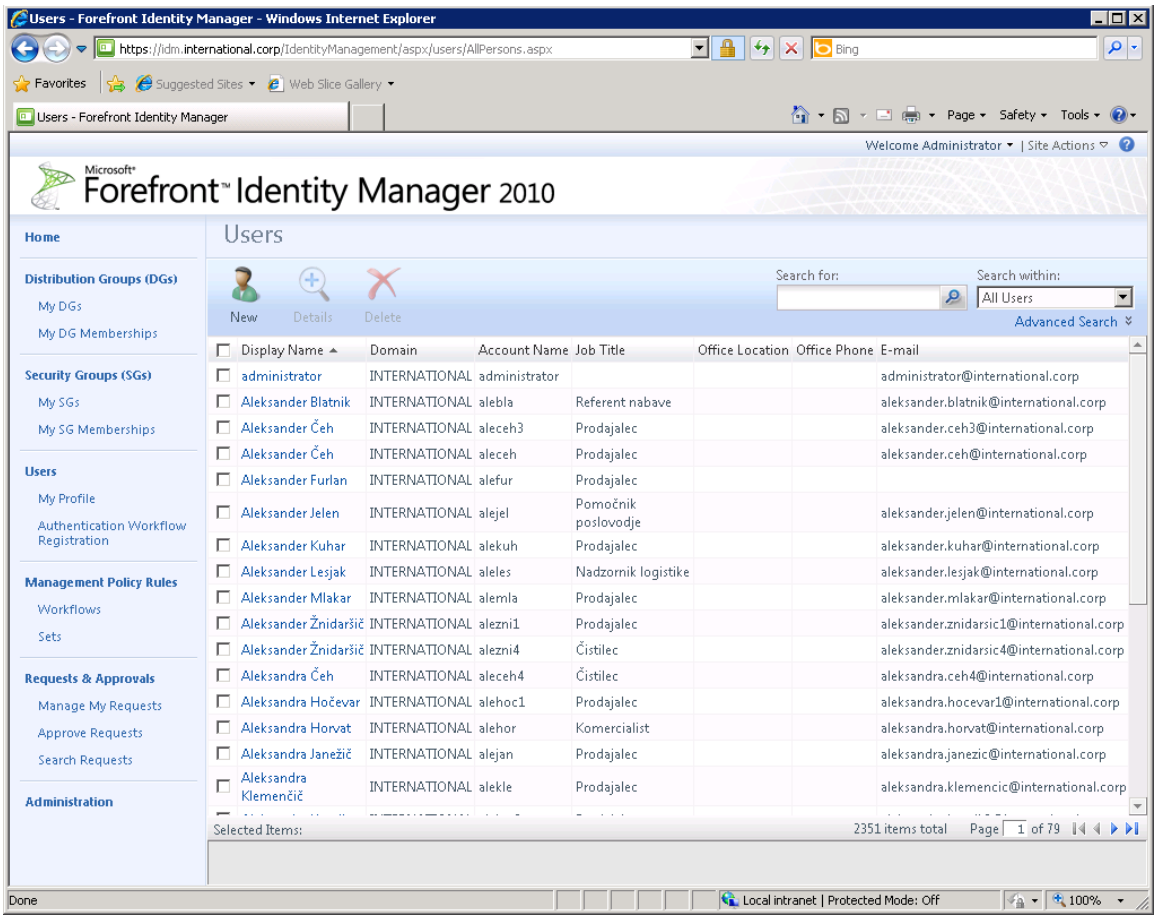


Slika 3-8: Upravljavsko pravilo za izvedbo procesa zaposlovanja nove osebe

3.3.4.3 Izdelava procesa za zaposlitev zunanjega sodelavca

FIM Portal privzeto vsebuje pogled pregleda identitet (pogled prikazuje Slika 3-9), na njem pa je možno izvajati dodajanje, urejanje ali brisanje identitet. Privzeta nastavitev je taka, da lahko identitete dodajajo le administratorji, hkrati pa jim je omogočeno tako spreminjanje podatkov vseh identitet kot tudi brisanje katerekoli identitete. Navadni uporabniki pa imajo

dosti bolj omejene pravice, saj v seznamu vidijo samo svojo identiteto in tudi to lahko le spreminjajo. Navadni uporabniki torej nimajo pravic dodajanja in brisanja identitet.



Slika 3-9: Pogled identitet v FIM Portalu

Poslovna pravila opredeljena v obravnavanem podjetju pa navajajo, da se zaposlene lahko dodaja samo preko kadrovskega sistema, zunanje sodelavce pa le preko FIM Portala. Izdelavo procesa za zaposlovanje zunanjega sodelavca je tako najlažje izvesti, da se uporabi obstoječe funkcionalnosti pogleda identitet, le da se uporabnikom in administratorjem v tem pogledu nastavi ustrezne pravice. Nastavitev pravic se izvede z naslednjim postopkom:

1. Omogoči se obstoječe upravljavsko pravilo *User management: Users can read selected attributes of other users*, ki dovoljuje, da vsi uporabniki vidijo vse identitete. Ta korak omogoči, da katerikoli uporabnik v pogledu vidi vse identitete.
2. Doda se novo področje iskanja ali *Search Scope*, ki omeji pogled nad identitetami tako, da vodje vidijo samo svoje zaposlene. Obstoječe področje iskanja *All Users*, ki dovoljuje iskanje po vseh identitetah, se nato dodeli samo administratorjem. Ta korak

omogoči, da se v pogledu identitet lahko išče in posledično tudi prikaže samo identitete podrejene vodjem.

3. Doda se novo množico podatkov, ki vsebuje le vodje identitet.
4. Doda se novo upravljavsko pravilo, ki množicama vodje in administratorji dovoljuje dodajanje in spreminjanje novih identitet, vendar le tipa zunanji sodelavec ali *Contractor*. Ta korak določa, da se lahko dodajajo le identitete za zunanje sodelavce.

3.3.4.4 Izdelava procesa za upravljanje s podatki zunanjega sodelavca

Z izdelavo procesa za zaposlitev zunanjega sodelavca je podprt tudi proces upravljanja s podatki zunanjega sodelavca. Ta je podprt z upravljavskim pravilom, katerega se je izdelalo v 4. koraku nastavitve pravic procesa zaposlovanja zunanjega sodelavca. Gre za pravilo, ki vodjem in administratorjem omogoča dodajanje in spreminjanje zunanjih sodelavcev. Tako lahko vodja ali administrator preko pogleda identitet izbere ustreznega zunanjega sodelavca in mu spremeni podatke.

K upravljanju s podatki zunanjega sodelavca pa spada tudi proces avtomatiziranega omogočanja in onemogočanja uporabniškega računa zunanjega sodelavca. Za razliko od izstopa zaposlenega, se pri tem procesu onemogoči samo uporabniški račun, vsa članstva v skupinah pa zunanjemu sodelavcu ostanejo. Za izvedbo avtomatiziranega upravljanja z aktivnostjo uporabniškega računa se:

1. Izdela akcijska delovna tokova za omogočanje oz. onemogočanje uporabniškega računa, imenovana *Enable Account Workflow* in *Disable Account Workflow*.
2. Opredeli se množico imenovano *Active Contractors*, katera vsebuje vse zunanje sodelavce, ki so aktivni glede na datum veljavnosti.
3. Izdela se dve upravljavski pravili; eno pravilo sproži akcijski delovni tok *Enable Account Workflow* takrat, ko postane identiteta zunanjega sodelavca del množice *Active Contractors*, drugo pa sproži akcijski delovni tok *Disable Account Workflow* takrat, ko identiteta zunanjega sodelavca ni več del množice *Active Contractors*.

3.3.4.5 Izdelava procesa za redni izstop zaposlenega

Redni izstop zaposlenega nastopi po poteku delovnega razmerja. Ta podatek je v FIM Service zapisan v entiteti *user*, v datumskem polju *Employee End Date*. Za pravilnost tega podatka skrbi kadrovski sistem.

FIM Service privzeto ne briše nobenih podatkov in je zato potrebno poskrbeti le za onemogočanje uporabniškega računa in umik članstva iz vseh skupin. Sama realizacija procesa pa se izvede z naslednjimi koraki:

1. Izdela se akcijski delovni tok, ki bo uporabniku onemogočil uporabniški račun in odstranil članstva iz vseh skupin.
2. Opredelimo množico podatkov, ki vsebuje zaposlene, katerim jim je poteklo delovno razmerje.
3. Opredelimo upravljavsko pravilo, ki bo za vsako identiteto, katera postane član zgoraj omenjene množice, sprožilo delovni tok brisanja podatkov.

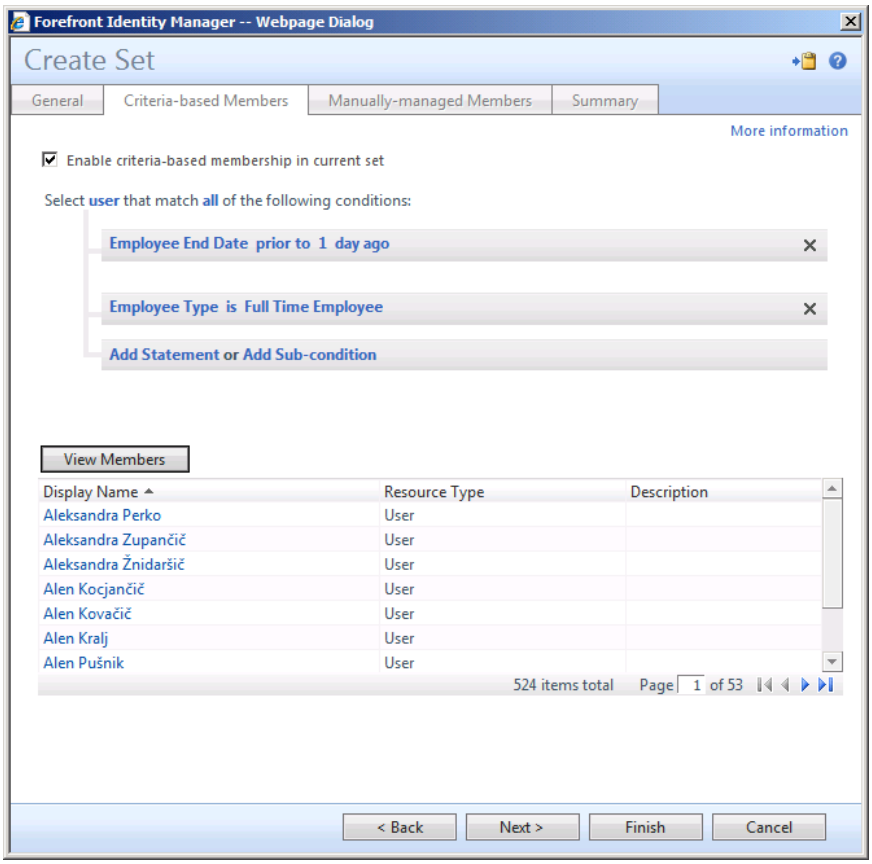
Izdelava akcijskega delovnega toka za onemogočanje uporabniškega računa

Akcijski delovni tok za onemogočanje uporabniškega računa in odstranjevanje članstva v skupinah, imenovan *ExpireIdentityWorkflow*, prikazuje Priloga H. Kot je razvidno iz priloge se celoten delovni tok razdeli na dva dela – v prvem delu nastavi uporabniku atribut *AccountDisabled* na vrednost *true*, v drugem pa v zanki odstrani članstva v vseh skupinah.

Opredelitev množice zaposlenih s poteklim delovnim razmerjem

Množico se opredeli tako, da se v administraciji doda nov objekt tipa *Set*, imenovan *Expired Employees*. V množici se nato nastavi kriterije, ki določajo kateri objekti postanejo del množice. Kriterija sta sledeča (primer kriterijev prikazuje Slika 3-10):

- uporabnik je redno zaposleni,
- današnji dan je 1 dan za datumom poteka delovnega razmerja.



Slika 3-10: Kriteriji za določanje zaposlenih s poteklim delovnim razmerjem

Opredelitev upravljavskega pravila za brisanje zaposlenih s poteklim delovnim razmerjem

Za opredelitev upravljavskega pravila dodamo nov objekt tipa *Management Policy Rule*, v katerem nastavimo, da se v trenutku, ko identiteta postane del množice *Expired Employees*, nad njo izvede delovni tok za onemogočanje računov in odstranjevanje dostopa.

3.3.4.6 Izdelava procesa za takojšnji izstop zaposlenega

Za razliko od procesa za redni izstop zaposlenega, kjer se avtomatsko ukinja dostope na podlagi podatkov o veljavnosti pogodbenega razmerja pridobljenih iz kadrovskega sistema, je pri procesu za takojšnji izstop to potrebno narediti ročno, ne glede na podatke iz kadrovskega sistema. Običajno se v kadrovskem sistemu že ustrezno uredi, da oseba nima več veljavnega pogodbenega razmerja, vendar ta sprememba pride v FIM Service šele ob nočni sinhronizaciji. Proces za takojšnji izstop zaposlenega pa poskrbi, da uporabnik v najkrajšem možnem času nima več dostopa do sistema.

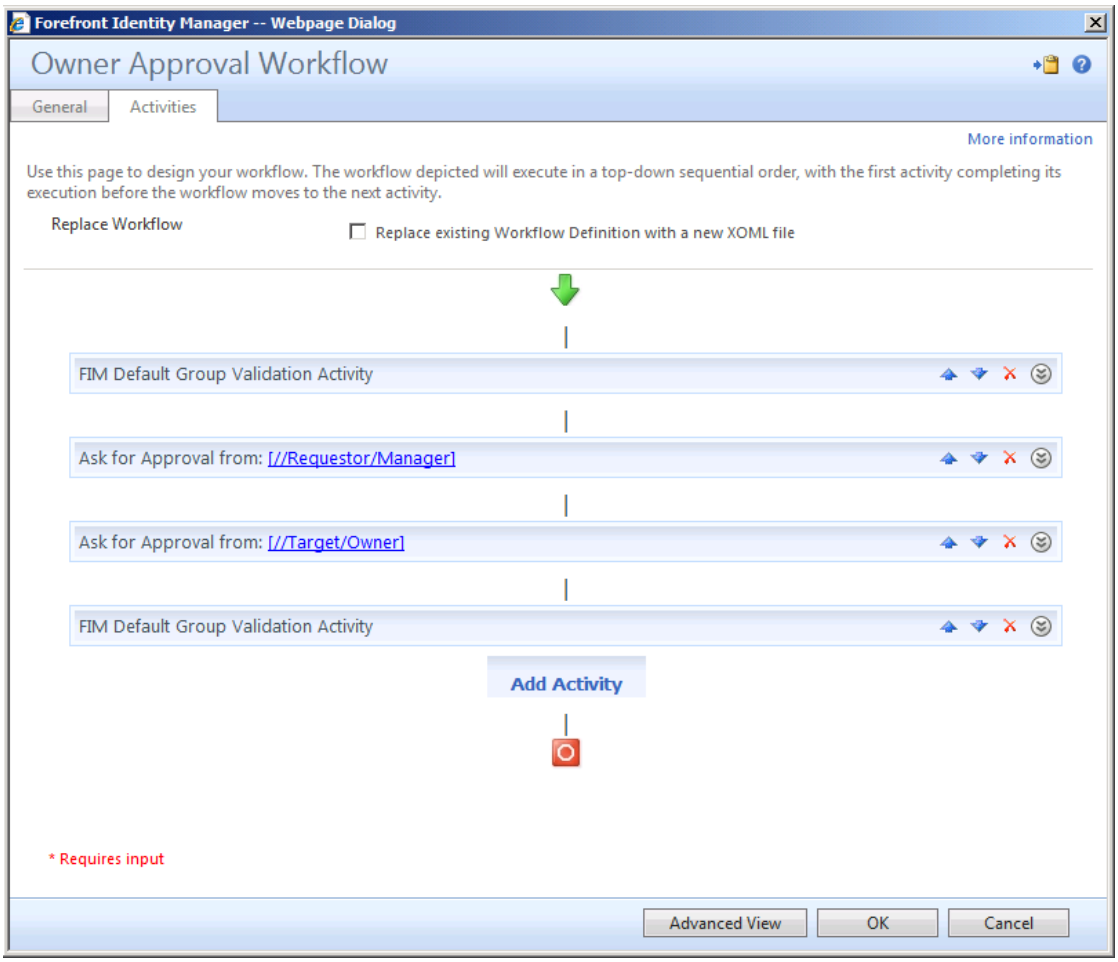
Proces je realiziran na tak način, da lahko administrator v pogledu identitet izbere ustreznega zaposlenega in mu nastavi atribut *Account Disabled* na vrednost *true*. Na podlagi tega atributa

FIM Synchronization Service poskrbi za onemogočanje uporabniškega računa pripadajočega uporabnika. Ukinitev računa in dostopov pa se nato izvaja s procesom rednega izstopa zaposlenega.

Ker se je z uvedbo procesa zaposlovanja zunanjih sodelavcev onemogočilo upravljanje z redno zaposlenimi preko FIM Portala, je za realizacijo procesa potrebno opredeliti novo upravljavsko pravilo, ki skrbnikom sistema FIM dovoljuje spreminjanje atributa *Account Disabled* za vse identitete, ki so del množice *Full Time Employees*.

3.3.4.7 Izdelava procesa za dodelitev dostopa

FIM Service privzeto že vsebuje delovni tok *Owner Approval Workflow*, ki je namenjen potrjevanju zahtev s strani lastnikov varnostnih in distribucijskih skupin. Za zagotovitev skladnosti delovnega toka s procesom dodeljevanja dostopa v obravnavanem podjetju, ki zahteva dvonivojsko potrjevanje, je potrebno v ta delovni tok dodati novo aktivnost *Ask for Approval from*, ki je namenjena potrjevanju zahtev. Pri opredelitvi te aktivnosti je potrebno določiti kdo bo zahtevo za dostop najprej potrjeval. To se nastavi tako, da se na aktivnosti oz. prvem nivoju potrjevanja opredeli potrjevalca z referenčno potjo *//Requestor/Manager*, kar pomeni, da se zahteva za dostop pošlje v potrjevanje vodji (*Manager*) tistega, ki je sprožil zahtevo (*Requestor*). Kot je razvidno iz spodnje slike (Slika 3-11), se po potrditvi vodje izvede naslednja (obstoječa) aktivnost *Ask for approval from* oz. drugi nivo potrditve, kjer pa zahtevo potrdi lastnik skupine oz. vloge. Ta je določen preko referenčne poti *//Target/Owner*, kar pomeni, da se aktivnost potrjevanja pošlje lastniku (*Owner*) skupine (Target).



Slika 3-11: Delovni tok aktivnosti potrjevanja.

3.3.4.8 Ostale funkcionalnosti FIM

Poleg izdelave zahtevanih procesov obravnavanega podjetja, pa FIM privzeto nudi tudi funkcionalnost upravljanja varnostnih in distribucijskih skupin. S temi skupinami lahko upravljajo lastniki skupin in administratorji sistema FIM.

3.3.5 Vzpostavitev FIM Synchronization Service

Pri postavitvi metaimenika je najprej potrebno za vsako entiteto iz entitetnega modela opredeliti avtoritativni vir tako same entitete kot tudi njenih atributov. Naslednji korak je vzpostavitev centralne hrambe in sinhronizacija podatkov med sistemi.

Z vpeljavo FIM Synchronization Servicea se v podjetju vzpostavi metaimenik. Kot sem že omenil v poglavju 2.8.2.3, FIM Synchronization Service uporablja upravljavske agente za pridobivanje in razprševanje podatkov med sistemi.

Sama vzpostavitev FIM Synchronization Service poteka v treh korakih:

1. Priprava entitetnega modela.
2. Konfiguracija in izvedba inicialnega uvoza podatkov v metaimenik in IdM sistem.
3. Konfiguracija in izvedba rednih sinhronizacij za prenos podatkov (sprememb) med sistemi.

Izvedba vsakega od zgoraj navedenih korakov je opisana v naslednjih podpoglavjih.

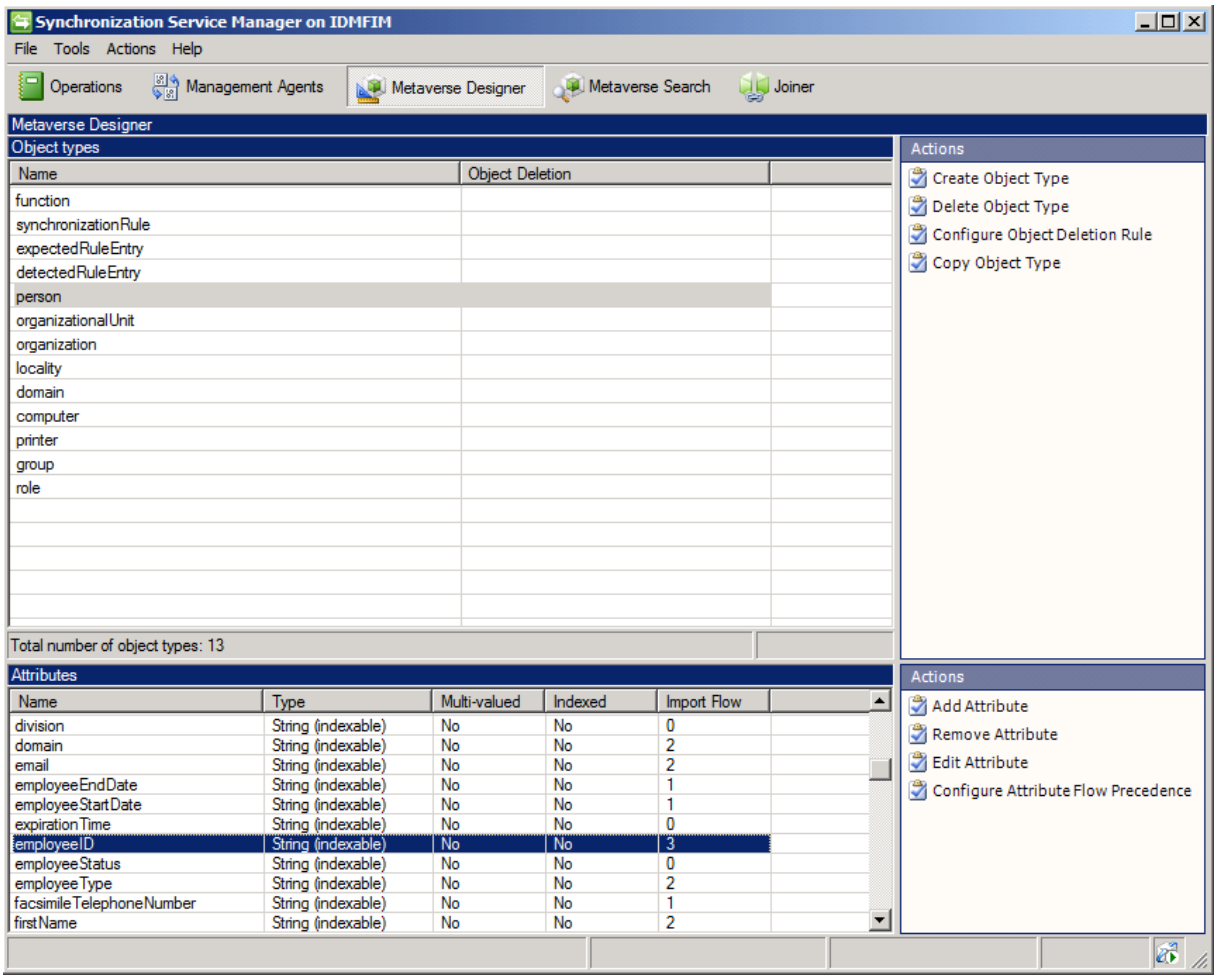
3.3.5.1 Priprava entitetnega modela

Po namestitvi FIM Synchronization Servicea ta že vsebuje nekaj vnaprej definiranih entitet z atributi. Za potrebe v obravnavanem podjetju so vnaprej definirane entitete ustrezne za uporabo, dodani so bili le manjkajoči atributi. Tabela 3-3 prikazuje preslikavo entitet obstoječih sistemov v entitete FIM Synchronization Service. Glede na to, da se nekatere obstoječe entitete preslikajo v isto entiteto FIM Synchronization Servicea, se te med seboj razlikujejo z vrednostjo atributa (npr. atribut *employeeType* določa ali gre za zaposlenega ali zunanjega sodelavca). Vendar v samem metaimeniku ni potrebe po razlikovanju, saj ciljni sistem (AD) ne rabi nobenega razlikovanja. Pri zaposlenih in zunanjih sodelavcih gre v obeh primerih na nivoju AD-ja le za uporabniški račun z določenimi dostopi, brez kakršnega koli razlikovanja. Vsebinska razdelitev je torej le na nivoju IdM sistema.

Entitete obstoječih sistemov	**Entiteta FIM Synchronization Service**
Zaposleni	person
Zunanji sodelavec	person
Organizacijska enota	organization
Varnostna skupina	group
Distribucijska skupina	group

Tabela 3-3: Preslikava entitet med obstoječimi sistemi in FIM Synchronization Serviceom

Slika 3-12 prikazuje urejevalnik entitet v FIM Synchronization Service, imenovan *Metaverse Designer*. Preko tega uporabniškega vmesnika se ureja entitete in njihove atribute. Poleg vrste podatka, ki ga hrani atribut, je potrebno določiti še prioritetni vrstni red atributa. Ta določa kateri upravljavski agent ima prednost pri zapisu vrednosti atributa v primeru, ko želi več upravljavskih agentov pisati v isti atribut. Tukaj morajo imeti prioriteto vedno tisti upravljavski agenti, ki berejo podatke iz avtoritativnih sistemov.



Slika 3-12: Urejevalnik entitet Metaverse designer

3.3.5.2 Inicialni uvoz

Pripravi entitetnega modela sledi inicialni uvoz podatkov. Z inicialnim uvozom poskrbimo, da iz vseh povezanih sistemov dobimo vse potrebne podatke, in da te podatke uvozimo v IdM sistem. S tem FIM Service pridobi podatke, ki jih potrebuje za upravljanje z identitetami in njihovimi dostopi.

Pri inicialnem uvozu, za razliko od redne sinhronizacije, so vsi povezani sistemi avtoritativni vir podatkov, IdM sistem pa je ciljni sistem. Pri prehodu iz inicialnega uvoza k rednim sinhronizacijam se običajno upravljavskih agentov za sisteme, ki ostanejo avtoritativni, ne spreminja, za sisteme, ki postanejo ciljni sistem, pa se naredi nove upravljavske agente.

Tabela 3-4 prikazuje entitete, ki se bodo sinhronizirale med sistemi, njihove atribute in avtoritativni vir atributov za vsako entiteto.

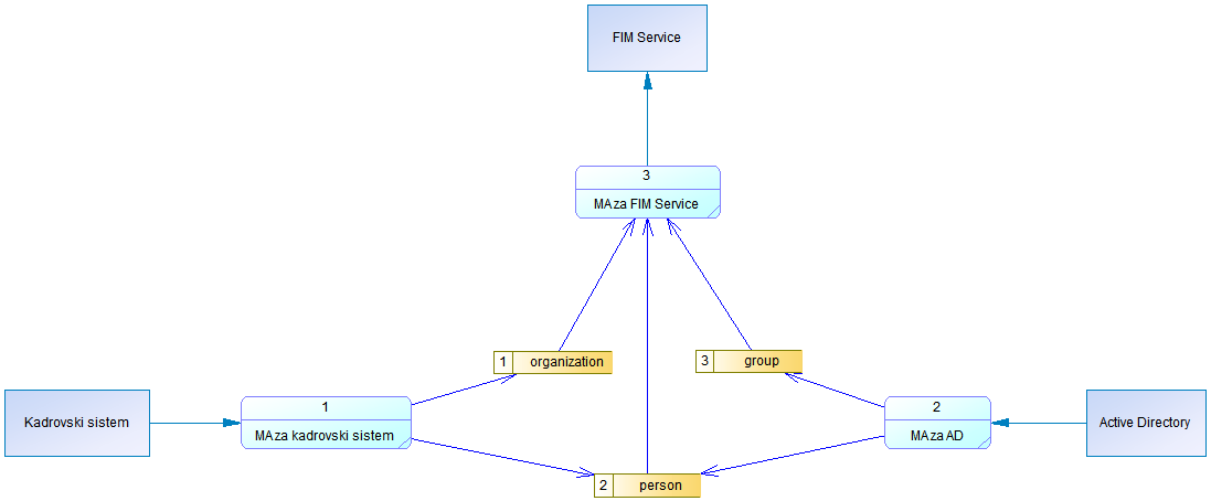
Entiteta	Atribut	Avtoritativni vir
Zaposleni	Kadrovska številka	Kadrovski sistem
	Ime	Kadrovski sistem
	Priimek	Kadrovski sistem
	Zaposlen od	Kadrovski sistem
	Zaposlen do	Kadrovski sistem
	Status	Kadrovski sistem
	Organizacijska enota	Kadrovski sistem
	Delovno mesto	Kadrovski sistem
	Vodja	Kadrovski sistem
	Telefonska številka	Kadrovski sistem
	Mobilna številka	Kadrovski sistem
	Faks številka	Kadrovski sistem
	Naslov	Kadrovski sistem
	Poštna številka	Kadrovski sistem
	Pošta	Kadrovski sistem
	Naselje	Kadrovski sistem
	Država	Kadrovski sistem
	Uporabniško ime	AD
	E-poštni naslov	AD
Zunanji sodelavec	Ime	AD
	Priimek	AD
	Podjetje	AD
	Telefonska številka	AD
	Vodja	AD
	Velja od	/
	Velja do	/
	Uporabniško ime	AD
	E-poštni naslov	AD
Organizacijska enota	Šifra	Kadrovski sistem
	Naziv	Kadrovski sistem
	Opis	Kadrovski sistem
Varnostna skupina	Naziv	AD
	Opis	AD
	E-poštni naslov	AD
	Lastnik skupine	AD
	Člani skupine	AD
Distribucijska skupina	Naziv	AD
	Opis	AD
	E-poštni naslov	AD
	Lastnik skupine	AD
	Člani skupine	AD

Tabela 3-4: Prikaz avtoritativnega vira entitet in njihovih atributov pri inicialnem uvozu

Kot lahko razberemo iz tabele je za organizacijske enote v celoti avtoritativni vir kadrovski sistem, za varnostne in distribucijske skupine pa v celoti AD. Prav tako je za zunanje

sodelavce avtoritativni vir AD, le da tam ni podatka o veljavnosti uporabniških računov. Ta dva podatka bodo vodje po inicialnem uvozu ročno napolnili preko FIM Portala. Entiteto zaposlenih pa sestavljajo atributi iz dveh sistemov. Vse atribute, razen uporabniškega imena in e-poštnega naslova, pridobimo iz kadrovskega sistema, izjemi pa dobimo iz AD-ja. Podatki zaposlenih se združijo na podlagi opredeljenega enoličnega identifikatorja, ki je v našem primeru kadrovska številka zaposlenega.

Tok podatkov entitet v metaimeniku oz. FIM Synchronization Serviceu je prikazan v spodnji sliki (Slika 3-13). V sliki je uporabljena kratica MA, ki pomeni upravljavski agent.



Slika 3-13: Tok podatkov v FIM Synchronization Service pri inicialnem uvozu

Pred samim uvozom podatkov pa je potrebno preko FIM Portala onemogočiti vse procese, ki se zaženejo ob dodajanju nove identitete ali grupe, saj pri inicialnem uvozu ne gre za nove identitete in grupe, temveč le za vzpostavitev obstoječega stanja.

Podrobnejšo implementacijo upravljavskih agentov za sinhronizacijo atributov entitet med posameznimi sistemi opisujejo naslednja poglavja.

3.3.5.2.1 Povezava AD-ja in metaimenika

Pri inicialnem uvozu podatkov iz AD-ja potrebujemo svojega upravljavskega agenta, ki bo prenesel vse uporabnike in skupine iz AD-ja v metaimenik. Za izvedbo te akcije sem naredil upravljavskega agenta *AD MA Initial Load*.

Pri konfiguraciji upravljavskega agenta je potrebno določiti:

- AD strežnik na katerega se agent poveže,
- objekte AD-ja in njihove atribute, ki jih bomo uvažali ter

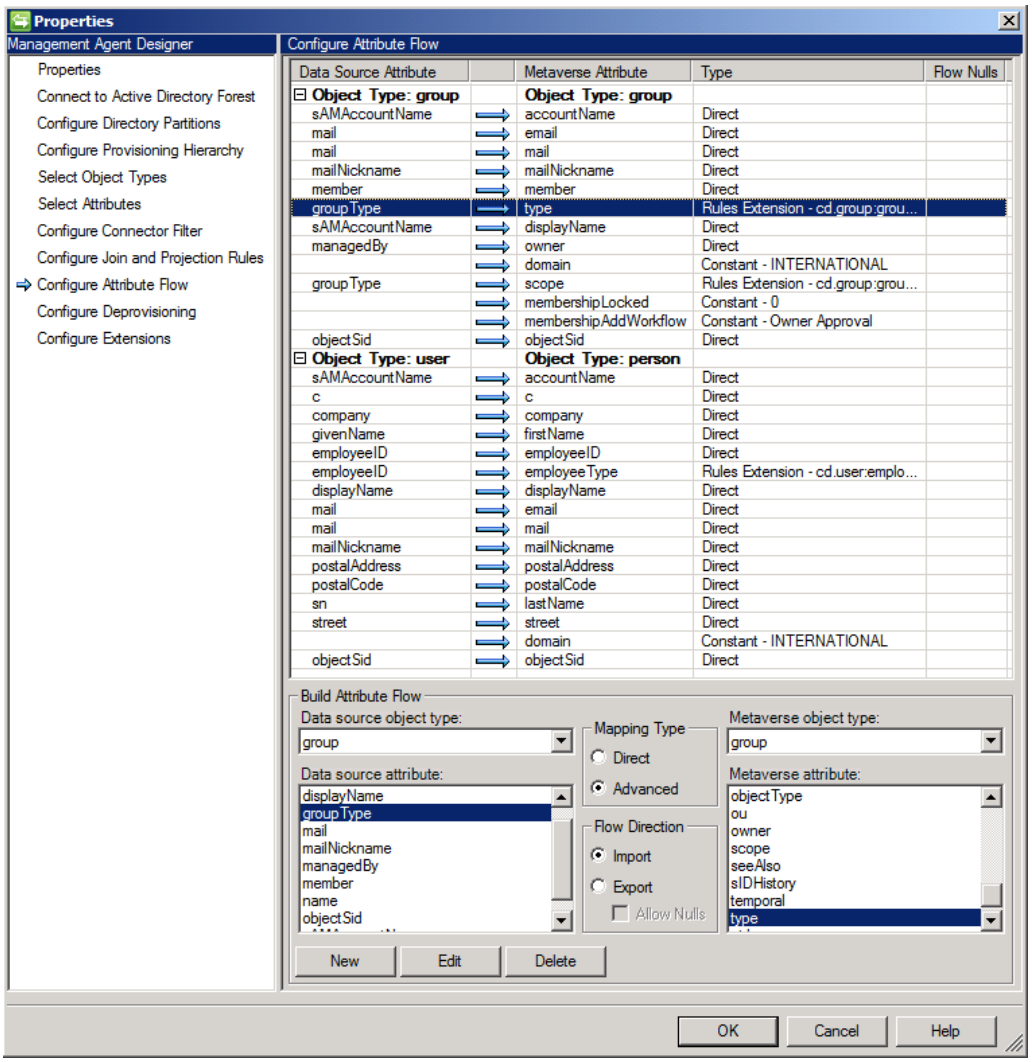
- preslikavo atributov objektov v atribute entitet metaimenika.

Slika 3-14 prikazuje konfiguracijo preslikav objektov v entitete metaimenika. Kot je razvidno je v preslikavi definiranih več atributov kot je določenih za preslikavo iz AD-ja, ki jih prikazuje Tabela 3-4. Te atribute potrebujeta ali metaimenik ali FIM Service, uporabniki pa jih ne vidijo, niti z njimi ne upravljajo. Ker pa s tem uvozom uvozimo tudi podatke zaposlenih (ime, priimek itd.), je ravno pri teh atributih potrebno biti pazljiv na prioritetni vrstni red v metaimeniku, saj se morajo za zaposlene vrednosti teh atributov prepisati iz kadrovskega sistema, ki je avtoritativni vir teh podatkov.

Poleg neposrednih preslikav (angl. *Direct*) in konstantnih vrednosti (angl. *Constant*), upravljavski agent vsebuje tudi razširjena pravila (angl. *Rules Extension*). Ta pravila so definirana v programski kodi, upravljavski agent pa jih potrebuje za:

- Razlikovanje varnostnih in distribucijskih skupin ter njihovega dosega: AD loči varnostne in distribucijske skupine ter njihov doseg v enem atributu (*groupType*), medtem ko FIM Service za to uporablja dva atributa (*type* in *scope*). Ker se take preslikave ne da izvesti neposredno, je potrebno definirati dve razširjeni pravili – za vsak atribut v FIM Service posebej.
- Razlikovanje med zaposlenimi in zunanjimi sodelavci: ker AD ne loči med zaposlenimi in zunanjimi sodelavci, je potrebno napisati pravilo, kjer se na podlagi kadrovske številke ugotovi, ali gre za zaposlenega ali zunanjega sodelavca – vsi, ki nimajo kadrovske številke, so zunanji sodelavci.

Vsa tri razširjena pravila prikazuje Priloga D.



Slika 3-14: Preslikave atributov upravljavskega agenta *AD MA Initial Load*

3.3.5.2.2 Povezava Exchangea 2010 in metaimenika

Neposredna povezava med Exchangeom 2010 in metaimenikom ni potrebna, saj podatke o poštnem naslovu dobimo iz AD-ja. E-poštni naslov je shranjen v atributu *mail* uporabniškega računa v AD in smo ga že definirali v upravljavskem agentu *AD MA Initial Load*.

3.3.5.2.3 Povezava kadrovskega sistema in metaimenika

Ker je kadrovski sistem vedno avtoritativni vir podatkov o zaposlenih in organizacijskih enotah, bomo isti upravljavski agent samo enkrat konfigurirali ter uporabljali za inicialni uvoz in redno sinhronizacijo.

Pri povezovanju FIM Synchronization Service in relacijske podatkovne zbirke imamo na voljo dva načina:

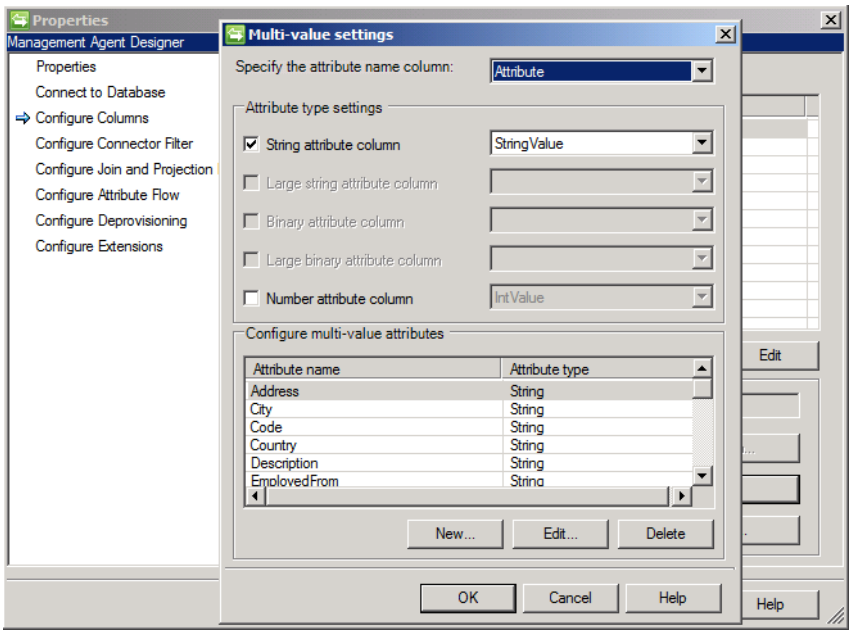
- za vsako tabelo ali pogled relacijske podatkovne zbirke se naredi posamezen upravljavski agent, saj lahko FIM Synchronization Service upravlja samo z eno tabelo oz. pogledom relacijske podatkovne zbirke, ali
- naredimo večvrednostno tabelo ali pogled. FIM Synchronization Service ponuja možnost, da lahko z enim upravljavskim agentom pridobimo podatke tudi iz več tabel oz. pogledov. To lahko naredimo tako, da imamo v eni tabeli oz. pogledu seznam entitet in njihovih enoličnih ključev, v drugi pa vrednosti atributov entitet vsebovanih v prvi tabeli oz. pogledu.

Razlika med načinoma je ta, da je pri prvi možnosti potrebno za vsako entiteto narediti svoj upravljavski agent, vendar se za to uporabi obstoječo tabelo ali pogled na relacijski bazi. Posledično je glede na drugi način sama implementacija enostavnejša in tudi samo delovanje upravljavskega agenta je hitrejše. Drugi način pa se običajno uporablja le v primerih, ko med samimi entitetami potrebujemo neposredno povezavo oz. referenco, saj se reference znotraj entitete ali med več entitetami uporablja samo znotraj enega upravljavskega agenta.

Pri povezovanju kadrovskega sistema sem izbral drugo možnost, kjer sem nad obstoječimi tabelami naredil dva pogleda:

- MV_Objects: vsebuje seznam entitet. Skripto za izdelavo pogleda prikazuje Priloga B.
- MV_ObjectValues: vsebuje vrednosti atributov entitet. Skripto za izdelavo prikazuje Priloga C.

V FIM Synchronization Service sem naredil upravljavskega agenta *SQL HR Read*, pri katerem je potrebno določiti le povezavo do strežnika podatkovne zbirke in ustrezno nastaviti atribute večvrednostnega pogleda, tako kot prikazuje Slika 3-15.

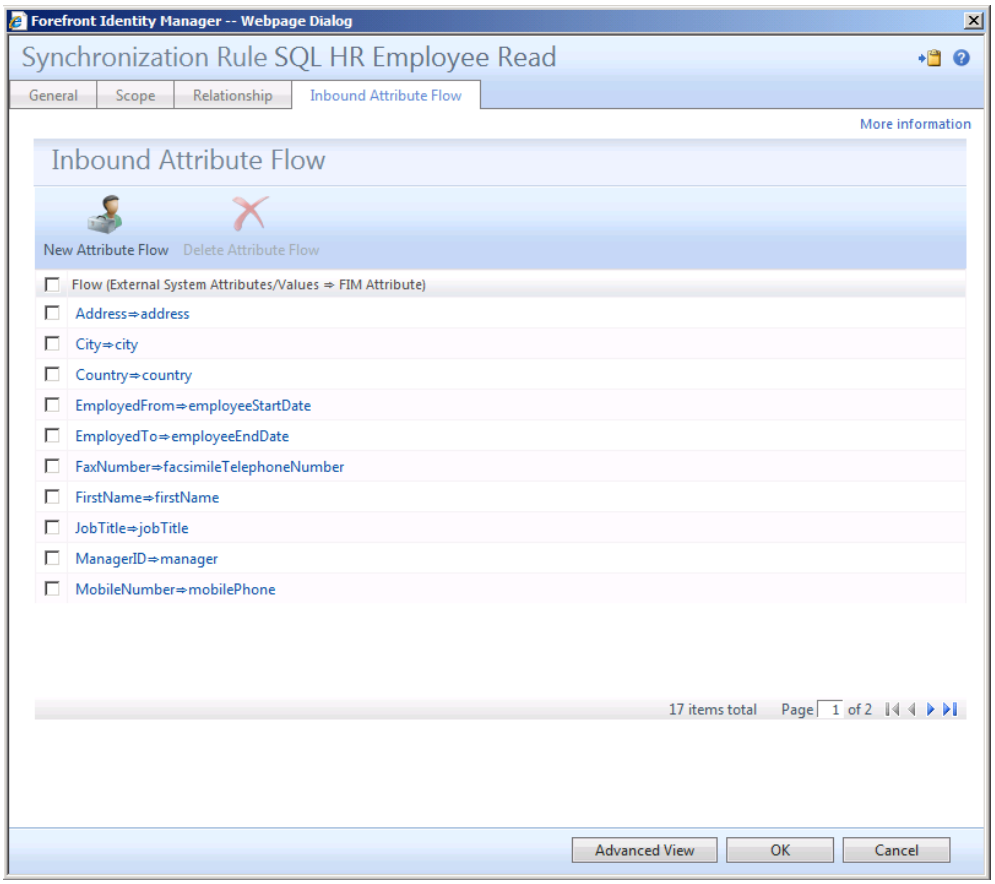


Slika 3-15: Nastavitve atributov večvrednostnega pogleda pri povezovanju s kadrovskim sistemom

Sama preslikava atributov entitet iz podatkovne zbirke v metaimenik pa se implementira v FIM Portal, z izdelavo sinhronizacijskega pravila (angl. *Synchronization Rule*). Tukaj pa za vsako entiteto potrebujemo svoje sinhronizacijsko pravilo, zato sem za potrebe sinhronizacije podatkov iz podatkovne zbirke pripravil dve sinhronizacijski pravili:

- SQL HR Org. Unit Read: definira preslikavo atributov organizacijskih enot pri uvozu v metaimenik.
- SQL HR Employee Read: definira preslikavo atributov zaposlenih pri uvozu v metaimenik.

Primer preslikave atributov sinhronizacijskega pravila za uvoz zaposlenih v metaimenik prikazuje Slika 3-16.

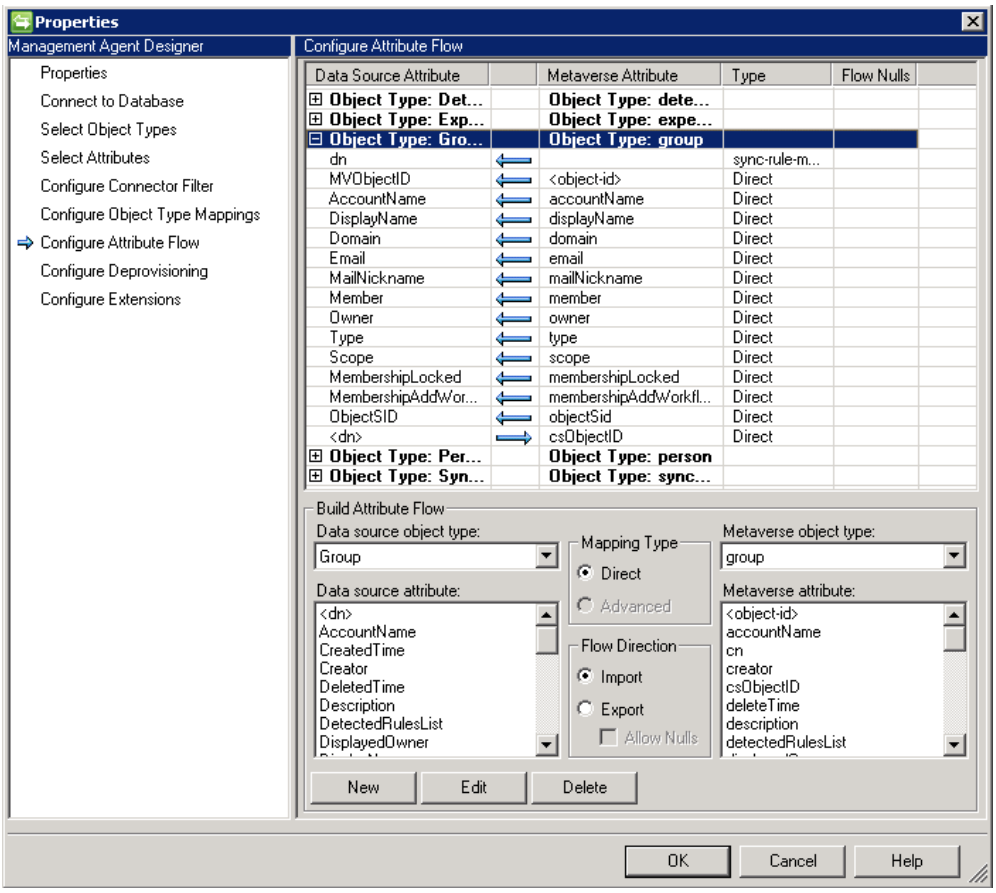


Slika 3-16: Preslikava atributov zaposlenih pri uvozu v metaimenik iz kadrovskega sistema

Pri uporabi sinhronizacijskih pravil je najprej potrebno preko upravljavskega agenta za FIM ta sinhronizacijska pravila prenesti v metaimenik, da jih lahko FIM Synchronization Service prične uporabljati.

3.3.5.2.4 Povezava metaimenika in FIM Servicea

Za namen inicialnega uvoza se za vzpostavitev povezave med metaimenikom in FIM Service naredi upravljavski agent imenovan *FIM MA Initial Load*. Agent se bo uporabil samo za inicialni uvoz, kjer se bo v FIM Service uvozilo podatke o zaposlenih iz kadrovskega sistema in njihovih uporabniških računih iz AD-ja, poleg tega pa se bo uvozilo še vse organizacijske enote ter varnostne in distribucijske skupine. Sama preslikava atributov temelji na podatkovnem toku atributov entitet, ki ga prikazuje Tabela 3-4. Primer preslikave atributov identitet iz metaimenika v FIM Service prikazuje Slika 3-17.



Slika 3-17: Preslikava atributov identitet med FIM Service in metaimenikom pri inicialnem uvozu

3.3.5.2.5 Izvedba inicialnega uvoza

Po ustrezni konfiguraciji upravljavskih agentov je nadaljnji korak izvedba inicialnega uvoza. Naslednje točke oz. koraki prikazujejo vrstni red akcij na posameznih upravljavskih agentih (posamezen korak je zapisan v obliki *ime upravljavskega agenta – izvedena akcija*):

1. **FIM MA – Full Import**: korak prenese vsa sinhronizacijska pravila v povezovalni prostor upravljavskega agenta.
2. **FIM MA – Full Synchronization**: korak prenese vsa sinhronizacijska pravila v metaimenik.
3. **AD MA Initial Load – Full Import**: prenese vse podatke iz AD-ja v povezovalni prostor upravljavskega agenta.
4. **AD MA Initial Load – Full Synchronization**: prenese vse podatke iz povezovalnega prostora upravljavskega agenta v metaimenik.
5. **HR SQL Read – Full Import**: prenese vse podatke iz kadrovskega SQL strežnika v povezovalni prostor upravljavskega agenta.

6. **HR SQL Read – Full Synchronization**: korak prenese in združi vse podatke iz povezovalnega prostora upravljavskega agenta v metaimenik.
7. **FIM MA – Export**: korak izvozi vse združene podatke iz metaimenika (podatke iz SQL strežnika in AD-ja) v FIM Service.

Pri sami izvedbi korakov inicialnega uvoza moramo preveriti rezultate 6. koraka preden nadaljujemo z naslednjim oz. zadnjim korakom. Ti rezultati nam namreč pokažejo kateri zaposleni so se v metaimenik na novo dodali in kateri zaposleni so se povezali z obstoječimi zapisi v metaimeniku (zapisi, ki smo jih pridobili iz AD-ja). Napaka nastane, če je bil uporabnik v metaimenik na novo dodan, čeprav že ima uporabniški račun. To pomeni, da se zapis ni povezal z obstoječim zapisom v metaimeniku. Najpogostejši razlog za to napako je ta, da nekateri uporabniki niso imeli vnesene kadrovske številke v AD-ju oz. enoličnega atributa, na podlagi katerega se povezujejo zapisi v metaimeniku. Do napak lahko pride tudi v primeru napačno vnesene kadrovske številke, vendar ne gre za napako v metaimeniku, ker je to vsebinska napaka, ki jo je na nivoju metaimenika težko odkriti. Take težave se običajno rešuje v IdM sistemu.

Po izvedbi vseh akcij je potrebno preveriti, ali so nastale kakšne napake oz. konflikti pri sinhronizaciji. Pred zaključkom celotne faze inicialnega uvoza je potrebno vse napake ali konflikte rešiti in še enkrat izpeljati celoten postopek sinhronizacije, ki se mora uspešno izvesti, brez kakršne koli spremembe podatkov.

3.3.5.3 Konfiguracija redne sinhronizacije

Pri prehodu iz inicialnega uvoza k redni sinhronizaciji se nekateri sistemi spremenijo iz avtoritativnih v ciljne. Kadrovski sistem ostane avtoritativni vir zaposlenih in organizacijskih enot, FIM Service postane avtoritativni vir za zunanje sodelavce ter varnostne in distribucijske skupine, AD pa se v celoti spremeni v ciljni sistem. Edina izjema je e-poštni naslov, katerega Exchange 2010 zapiše v atribut uporabniškega računa in ga nato iz AD-ja preberemo v metaimenik.

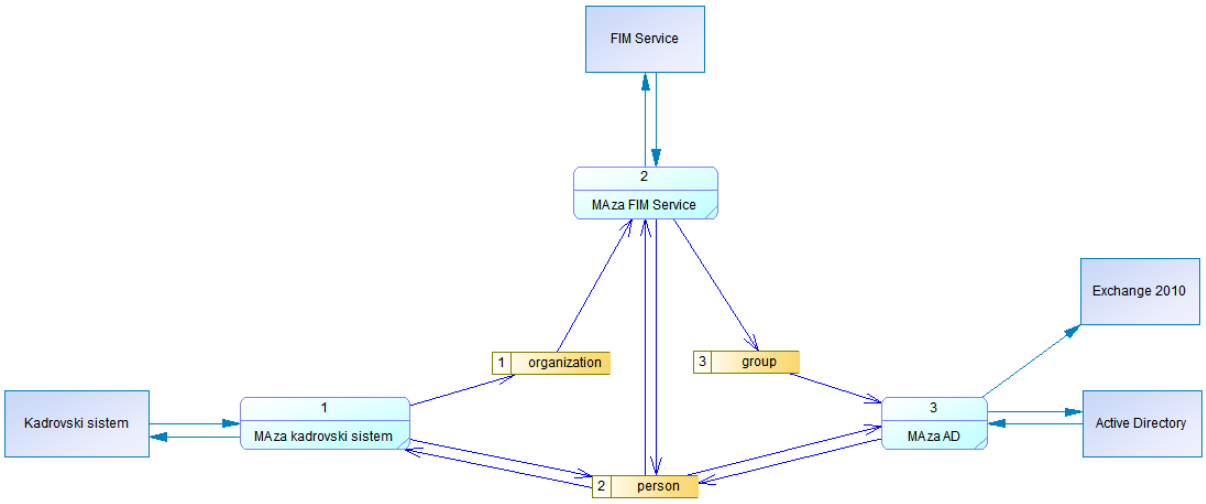
Tabela 3-5 prikazuje entitete in avtoritativne vire njenih atributov pri redni sinhronizaciji.

Entiteta	Atributi	Avtoritativni vir
Zaposleni	Kadrovska številka	Kadrovski sistem
	Ime	Kadrovski sistem
	Priimek	Kadrovski sistem
	Zaposlen od	Kadrovski sistem
	Zaposlen do	Kadrovski sistem
	Status	Kadrovski sistem
	Organizacijska enota	Kadrovski sistem
	Delovno mesto	Kadrovski sistem
	Vodja	Kadrovski sistem
	Telefonska številka	Kadrovski sistem
	Mobilna številka	Kadrovski sistem
	Faks številka	Kadrovski sistem
	Naslov	Kadrovski sistem
	Poštna številka	Kadrovski sistem
	Pošta	Kadrovski sistem
	Naselje	Kadrovski sistem
	Država	Kadrovski sistem
	Uporabniško ime	FIM
	E-poštni naslov	Exchange 2010
Zunanji sodelavec	Ime	FIM
	Priimek	FIM
	Podjetje	FIM
	Telefonska številka	FIM
	Vodja	FIM
	Velja od	FIM
	Velja do	FIM
	Uporabniško ime	FIM
	E-poštni naslov	FIM
Organizacijska enota	Šifra	Kadrovski sistem
	Naziv	Kadrovski sistem
	Opis	Kadrovski sistem
Varnostna skupina	Naziv	FIM
	Opis	FIM
	E-poštni naslov	FIM
	Lastnik skupine	FIM
	Člani skupine	FIM
Distribucijska skupina	Naziv	FIM
	Opis	FIM
	E-poštni naslov	FIM
	Lastnik skupine	FIM
	Člani skupine	FIM

Tabela 3-5: Prikaz avtoritativnega vira entitet in njihovih atributov pri redni sinhronizaciji

Tok podatkov entitet v FIM Synchronization Service pri redni sinhronizaciji prikazuje Slika 3-18. Kot je razvidno se tok upravljavskega agenta za FIM Service spremeni v dvosmernega,

upravljavski agent za AD pa zapisuje tako v AD kot v Exchange 2010. Zapis e-poštnega naslova iz AD-ja v kadrovski sistem je nakazan s tokom podatkov iz AD-ja v entiteto *person* in nato v kadrovski sistem. V sliki je uporabljena kratica MA, ki pomeni upravljavski agent.

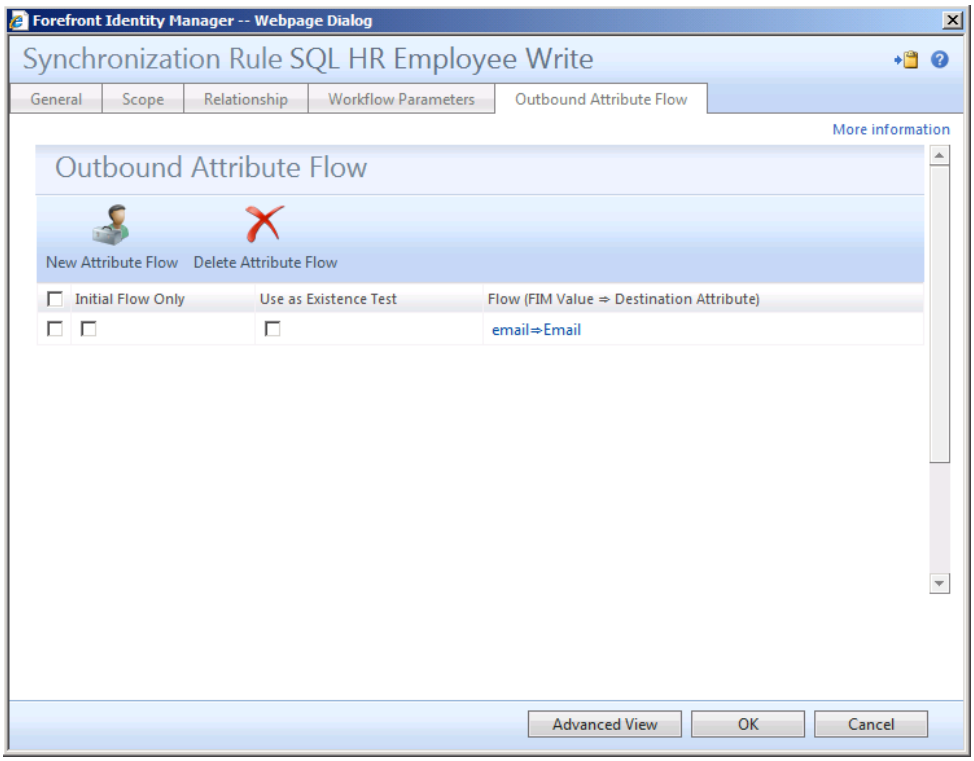


Slika 3-18: Tok podatkov v FIM Synchronization Service pri redni sinhronizaciji

Konfiguracije posameznih upravljavskih agentov so podrobneje opisane v sledečih podpoglavjih.

3.3.5.3.1 Povezava kadrovskega sistema in metaimenika

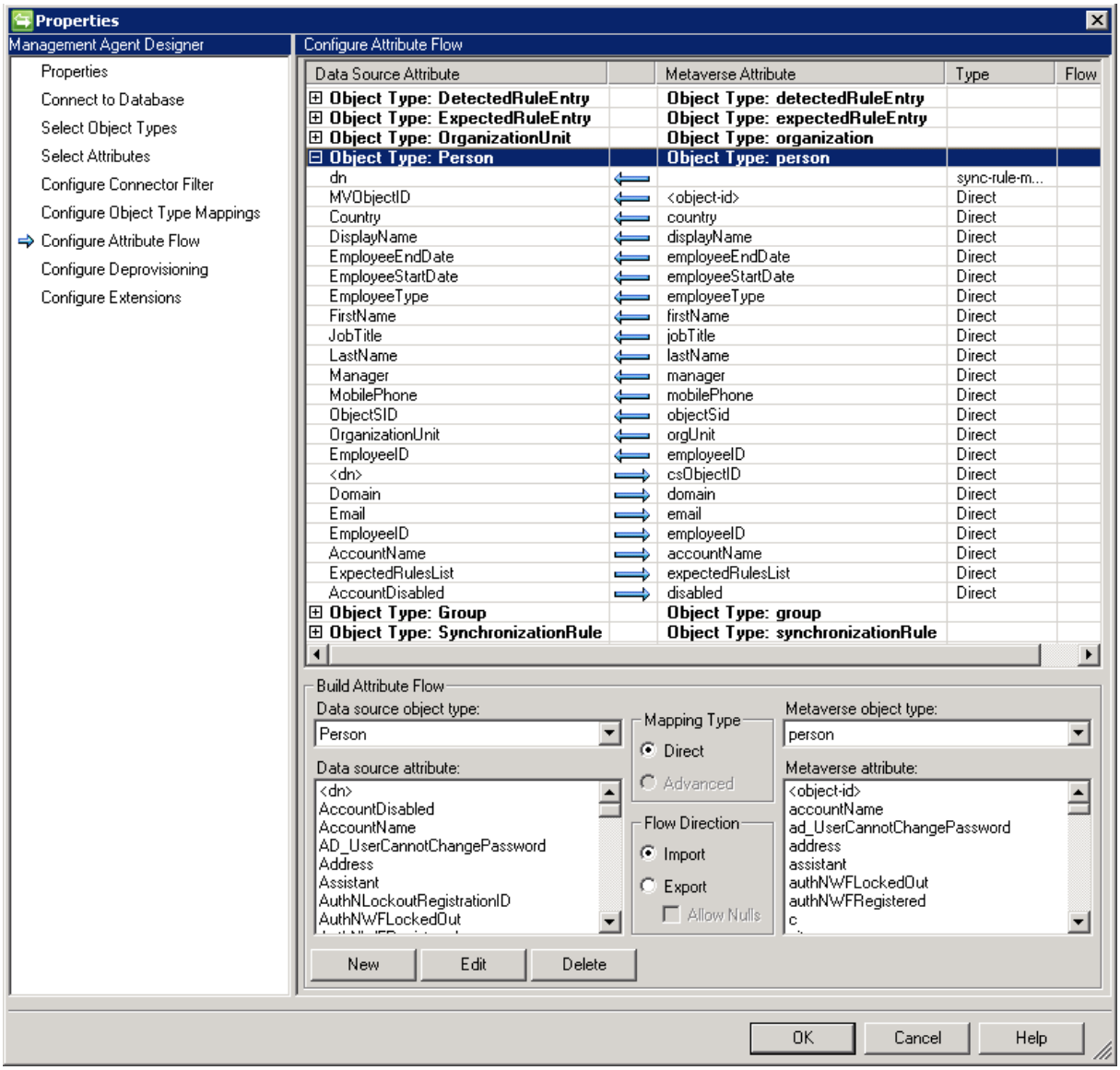
Pri uvozu podatkov o zaposlenih in organizacijskih enotah iz kadrovskega sistema v metaimenik ostaja postopek enak kot pri inicialnem uvozu. Celoten postopek je opisan v poglavju 3.3.5.2.3. Ker pa redna sinhronizacija vključuje tudi zapisovanje e-poštnega naslova v kadrovski sistem, je potrebno narediti nov upravljavski agent za zapisovanje, imenovan *SQL HR Write*. Podobno kot pri branju podatkov iz kadrovskega sistema, je tudi tukaj potrebno opredeliti povezavo do podatkovnega strežnika in določiti tabelo v katero se bodo podatki zapisovali. Sama preslikava atributov entitet pa se definira v sinhronizacijskem pravilu na FIM Portalu. Preslikavo atributov zaposlenih pri izvozu podatkov v kadrovski sistem prikazuje Slika 3-19.



Slika 3-19: Preslikava atributov zaposlenih z upravljavskim agentom *SQL HR Write*

3.3.5.3.2 Povezava FIM Servicea in metaimenika

Za namen povezovanja FIM Servicea in metaimenika se naredi upravljavski agent imenovan *FIM MA*. Pri konfiguraciji upravljavskega agenta je potrebno opredeliti povezavo do FIM Servicea, izbrati objekte in atribute, ki se sinhronizirajo, ter določiti preslikavo atributov med FIM Serviceom in metaimenikom. Preslikava upošteva vire in smeri prenosa podatkov atributov na podlagi podatkovnega toka, ki ga opredeljuje Tabela 3-5. Primer preslikave podatkov identitet med FIM Serviceom in metaimenikom pa prikazuje Slika 3-20. Sinhronizacijska pravila ali razširljiva programska koda za sinhronizacijo niso potrebna.



Slika 3-20: Preslikava atributov entitet med FIM Serviceom in metaimenikom

3.3.5.3.3 Povezava metaimenika in AD-ja

Povezavo metaimenika in AD-ja je potrebno vzpostaviti z novim upravljavskim agentom, saj se vloga AD-ja spremeni iz avtoritativnega v ciljni sistem. Novi upravljavski agent bo v celoti prevzel vse naloge pri povezovanju z AD-jem, zato se lahko stari upravljavski agent, s katerim se je izvedel inicialni uvoz iz AD-ja, odstrani iz FIM Synchronization Service.

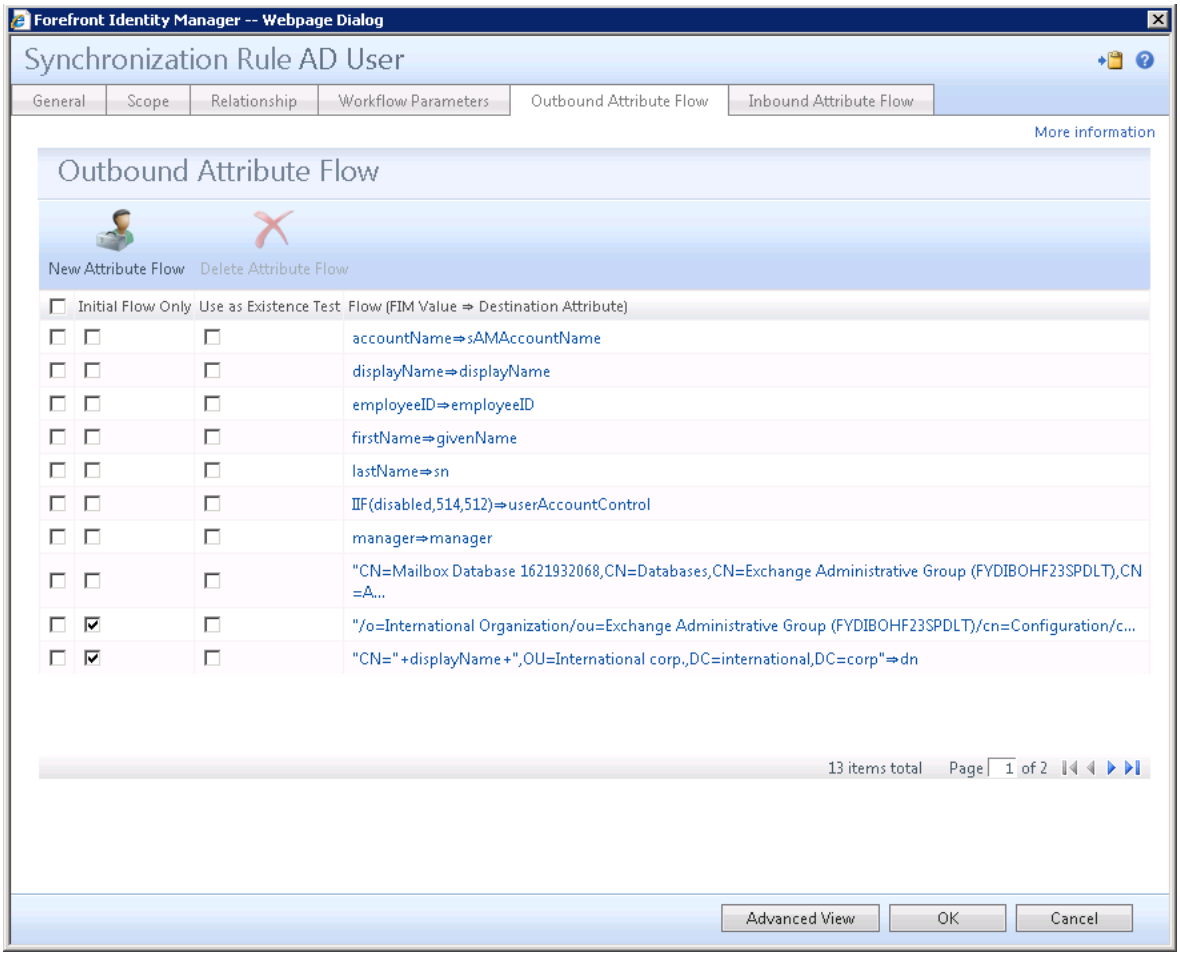
Pri vzpostavljanju povezave med metaimenikom in AD-jem je potrebno v prvem koraku narediti nov upravljavski agent, imenovan *AD MA*. V agentu je potrebno nastaviti le povezavo do AD-ja ter izbrati objekte in njihove atribute, ki se jih bo uporabljalo pri prenosu podatkov iz enega sistema v drug. Atribut preko katerega se povezujejo uporabniški računi in identitete je v tem upravljavskem agentu uporabniško ime in ne kadrovska številka kot pri inicialnem

uvozu. Razlog za to je, da se vse uporabniške račune povezuje na enak način, samo razlikovanje pa je uveljavljeno v FIM Service.

Preslikavo atributov entitet med metaimenikom in AD-jem pa se nastavi s sinhronizacijskimi pravili preko FIM Portala. Za ustrezno preslikavo potrebujemo dve sinhronizacijski pravili:

- *AD User*: pravilo za preslikavo uporabniških računov,
- *AD Group*: pravilo za preslikavo varnostnih in distribucijskih skupin.

Pri izvozu v AD se prenaša podatke uporabnika, kontrolne podatke, s katerimi se upravlja z računom, in podatke potrebne za izdelavo poštnih predalov. Vir teh podatkov sta kadrovski sistem in FIM. Primer preslikave podatkov uporabniških računov sinhronizacijskega pravila *AD User* prikazuje Slika 3-21. Iz AD-ja pa uvažamo le e-poštni naslov.



Slika 3-21: Preslikava izvoznih atributov identitet sinhronizacijskega pravila za AD

3.3.5.3.4 Povezava metaimenika in sistema Microsoft Exchange 2010

Povezovanje metaimenika in sistema Microsoft Exchange 2010 poteka posredno preko sinhronizacijskega pravila za izvoz v AD in upravljavskega agenta *AD MA*. Za avtomatsko izdelavo poštnih predalov v Microsoft Exchange 2010 je potrebno:

1. V upravljavskem agentu *AD MA* med dodatnimi nastavitvami izbrati izdajo Microsoft Exchange 2010 in vnesti spletni naslov RPS storitve.
2. Za preslikavo atributov je potrebno v AD-ju nastaviti naslednje atribute: *mailNickname, homeMDB, msExchHomeServerName* in *msExchRBACPolicyLink.*

Z nastavitvijo upravljavskega agenta in izvozom v zgoraj navedene atribute, se ob izdelavi novega uporabniškega računa izdela tudi poštni predal z nazivom, ki je podan v atributu *mailNickname.* Če je ta atribut prazen, se poštni predal ne bo ustvaril. Microsoft Exchange 2010 ob izdelavi novega poštnega predala samodejno zapiše e-poštni naslov v atribut *mail* uporabniškega računa uporabnika.

3.3.5.4 Izvedba redne sinhronizacije

Redna sinhronizacija se bo med sistemi izvajala v dveh fazah:

- urna sinhronizacija sprememb med FIM Service in AD-jem,
- dnevna sinhronizacija vseh podatkov med vsemi sistemi.

Obe zgoraj navedeni fazi pa se bosta izvajali avtomatsko in sicer s skripto, ki jo bo sprožal sistemski časovnik.

3.3.5.4.1 Urna sinhronizacija sprememb med FIM Service in AD-jem

Sinhronizacija bo izvajala t.i. »delta« sinhronizacije med sistemoma FIM Service in AD-jem. Delta uvozi in sinhronizacije uvažajo in sinhronizirajo samo podatke, ki so se spremenili od zadnjega uspešnega uvoza oz. sinhronizacije upravljavskega agenta ter so lahko s tem nekajkrat hitrejše od celotne sinhronizacije (običajno se posamezna delta sinhronizacija izvede znotraj parih minut). Sistemski časovnik bo nastavljen tako, da se bo od ponedeljka do petka vsako polno uro med 6:00 in 22:00 izvedla sinhronizacija med FIM Service in AD-jem v naslednjem zaporedju (posamezen korak je zapisan v obliki *ime upravljavskega agenta – izvedena akcija*):

1. **FIM MA – Delta Import**: prenese vse spremembe od zadnjega delta uvoza v povezovalni prostor upravljavskega agenta.

2. **FIM MA – Delta Synchronization**: v metaimenik združi vse spremembe od zadnje delta sinhronizacije in spremembe označi za izvoz v ciljne sisteme.
3. **AD MA – Export**: izvozi vse spremembe v AD.
4. **AD MA – Delta Import**: prenese vse spremembe od zadnjega delta uvoza v povezovalni prostor upravljavskega agenta.
5. **AD MA – Delta Syncrhonization**: v metaimenik združi vse spremembe od zadnje delta sinhronizacije in spremembe označi za izvoz v ciljne sisteme.
6. **FIM MA – Export**: izvozi vse spremembe v FIM Service.

3.3.5.4.2 Dnevna sinhronizacija vseh podatkov med vsemi sistemi

Dnevna sinhronizacija je namenjena uvozu novozaposlenih v FIM Service in celotni sinhronizaciji vseh sprememb med vsemi sistemi. Ker pa je celotna sinhronizacija med vsemi sistemi lahko dolgotrajna, se bo ta izvajala enkrat dnevno ob drugi uri zjutraj v naslednjem zaporedju (posamezen korak je zapisan v obliki *ime upravljavskega agenta – izvedena akcija*):

1. **SQL HR Read – Full Import**: prenese vse spremembe v kadrovskem sistemu v povezovalni prostor upravljavskega agenta.
2. **SQL HR Read – Full Synchronization**: v metaimenik združi vse spremembe in spremembe označi za izvoz v ciljne sisteme.
3. **FIM MA – Export**: izvozi vse spremembe v FIM Service.
4. **FIM MA – Import**: uvozi vse spremembe iz FIM Servicea v povezovalni prostor upravljavskega agenta.
5. **FIM MA – Full Synchronization**: v metaimenik združi vse spremembe in spremembe označi za izvoz v ciljne sisteme.
6. **AD MA – Export**: izvozi vse spremembe v AD.
7. **AD MA – Import**: uvozi vse spremembe iz AD-ja v povezovalni prostor upravljavskega agenta.
8. **AD MA – Full Synchronization**: v metaimenik združi vse spremembe in spremembe označi za izvoz v ciljne sisteme.
9. **FIM MA – Export**: izvozi vse spremembe v FIM Service.
10. **SQL HR Write – Export**: izvozi vse spremembe v kadrovski sistem.

4 Sklep

Dejstvo je, da je celotno področje upravljanja z identitetami zelo široko in kompleksno. Področja upravljanja z identitetami ne moremo obravnavati kot zaključeno celoto, saj se nanaša in je celo odvisno od drugih varnostnih področij. Izmed mnogih IdM sistemov, ki jih danes ponuja tržišče, sem za praktičen prikaz uvedbe izbral FIM predvsem zaradi lastnih izkušenj z njim in njegovim predhodnikom ILM. Po drugi strani pa ga je tudi dokaj preprosto namestiti in z njim je možno ustrezno podpreti zadane poslovne procese, saj je celoten sistem razširljiv.

Ključni rezultat diplomske naloge je prikaz na kakšen način lahko v podjetje vpeljemo IdM sistem, ki vsebuje osnovne procese za podporo zaposlovanja in upravljanja dostopa. Rezultat prikazane uvedbe sistema v podjetje je avtomatiziran proces izdelave uporabniškega računa in poštnega predala ob zaposlitvi ter odstranitev uporabniškega računa in ukinitev njegovih pravic ob prekinitvi zaposlitve. Poleg izdelave in ukinitve uporabniškega računa pa se za vsakega zaposlenega in zunanjega sodelavca vodi evidenca in nadzor nad vsemi dostopi, ki jih pridobi na podlagi članstva v AD skupini. Ti dostopi se dodeljujejo na podlagi zahtevka, katerega morajo potrditi tako vodje identitete kot tudi lastniki AD skupin. S tem lastniki skupin ter vodje zaposlenih in zunanjih sodelavcev sprejmejo vlogo skrbnikov nadzora, s čemer se tudi porazdeli odgovornost iz IT oddelka na dejanske vodje, ki vsebinsko poznajo ustreznost potreb zaposlenih po dostopih in pravicah. Poleg uvedenih procesov zaposlovanja pa FIM postane tudi enotni vir za upravljanje z zunanjimi sodelavci, katero je bilo do sedaj dokaj nenadzorovano vodeno s strani administratorjev AD-ja.

Z uvedbo sistema FIM je v obravnavanem podjetju omogočeno, da ima oseba ob zaposlitvi lahko že v nekaj urah uporabniški račun in vse potrebne dostope. Dostop je potrebno še vedno ročno zahtevati in potrjevati, vendar pa bi se z uvedbo RBAC v naslednjih fazah te osnovne dostope na podlagi vloge v podjetju lahko tudi avtomatiziralo. Prav tako je pomembno, da zaposleni ob odhodu iz podjetja nima več dostopa in da se zunanjim sodelavcem ustrezno časovno omejuje dostop do sistemov, za kar pa skrbijo njihovi vodje. Z uvedbo teh osnovnih procesov se v podjetju obvladuje dostope do vseh sistemov, ki uporabljajo avtentifikacijo preko AD-ja, preko enotnega spletnega uporabniškega vmesnika. Poleg uvedbe RBAC, naslednje faze predvidevajo tudi implementacijo poročil skladnosti in revizijo dostopov ter povezovanje z dodatnimi sistemi, ki ne izvajajo avtentifikacije preko AD-ja.

Pri iskanju glavnega razloga za uvedbo IdM sistema v organizacijo se na prvem mestu vsekakor pojavi zagotavljanje višjega nivoja varnosti in nadzora nad dostopi. Drugi razlogi za uvedbo sistema so tudi avtomatizacija obstoječih procesov upravljanja varnosti, potrebe po skladnosti z mednarodnimi standardi in predpisi, zmanjševanje stroškov in še nekaj bi jih lahko našteli. Če IdM sistemi nudijo toliko prednosti, zakaj torej niso prisotni v večini današnjih organizacij? V večini velikih, še posebej mednarodnih podjetjih, so IdM sistemi že prisotni, saj so nujni za obvladovanje nadzora dostopa, vendar pa marsikatero veliko ali srednje veliko podjetje tega sistema še nima, čeprav ga potrebuje. Eden izmed glavnih razlogov je finančna in časovna zahtevnost uvedbe ter tudi sama zahtevnost področja, v katerega se uvaja IdM sistem. Iz vidika finančne zahtevnosti so IdM sistemi dragi predvsem zato, ker se običajno poleg nekega fiksnega stroška za nakup sistema, ki ni majhen, pojavljajo tudi stroški nakupa ustreznih licenc za vsako identiteto. Poleg tega pa je potrebno celoten sistem prilagoditi zahtevah organizacije (npr. poslovni procesi, ustrezni entitetni modeli, kompleksnejša pravila upravljanja dostopov itd.). Tudi z vsebinskega in organizacijskega vidika je uvedba sistema zelo zahtevna, saj morajo v organizaciji prilagoditi ali celo vzpostaviti celotna standardizirana varnostna pravila, katera se bo vpeljalo v sistem. Tukaj pa se dostikrat pojavi veliko težav, saj mnoge organizacije nimajo zastavljenih ustreznih varnostnih pravil ali pa jih ne izvajajo. Mnoge organizacije, ki imajo ta pravila, pa vsebujejo veliko izjem, pri čemer morajo biti vse izjeme obravnavane. In ker se velikokrat sočasno z uvedbo IdM sistema opredeljuje in dopolnjuje ta varnostna pravila v organizaciji, je lahko uvedba tudi dolgotrajna.

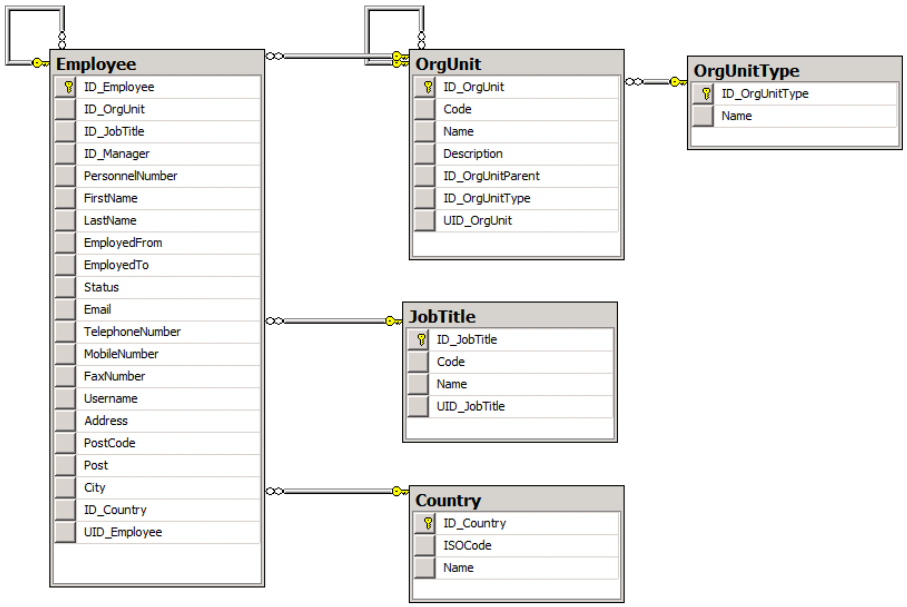
Tako lahko sklepamo, da je uvedba IdM sistema za organizacije na krajši rok lahko zelo velik strošek, za katerega niti ni nujno, da se v celoti povrne, vendar pa se na daljši rok določeni stroški znižujejo. Tukaj gre to predvsem na račun razbremenitve službe za pomoč uporabnikom ter skrajšanja odzivnih rokov glede dostopov in pravic na podlagi avtomatiziranih postopkov. Kljub velikim stroškom pa se mnoga podjetja vseeno odločijo za uvedbo sistema, saj ima v današnjih časih varnost izjemen pomen tako na družbenem kot poslovnem področju, zato stroški, ki nastajajo na podlagi zagotavljanja varnosti, niso nekaj nepotrebnega, temveč nekaj obveznega. Seveda vsako podjetje stremi k temu, da so stroški čim manjši in razen velikega začetnega vložka, se z uvedbo IdM sistema stroški namenjeni zagotavljanju varnosti, kljub rasti podjetja, ne zvišujejo, ampak se lahko celo znižujejo.

5 Slike

6 Tabele

7 Priloge

Priloga A: Osnovni relacijski model podatkovne zbirke HRM sistema



Priloga B: Skripta za izdelavo pogleda za pridobivanje entitet iz HRM sistema

```
CREATE VIEW [dbo].[MV_Objects]
AS
SELECT CAST(UID_Employee AS NVARCHAR(200)) as ObjectID, 'Employee' AS ObjectType
FROM Employee
UNION ALL
SELECT CAST(UID_OrgUnit AS NVARCHAR(200)) as ObjectID, 'Org. Unit' AS ObjectType
FROM OrgUnit

GO
```

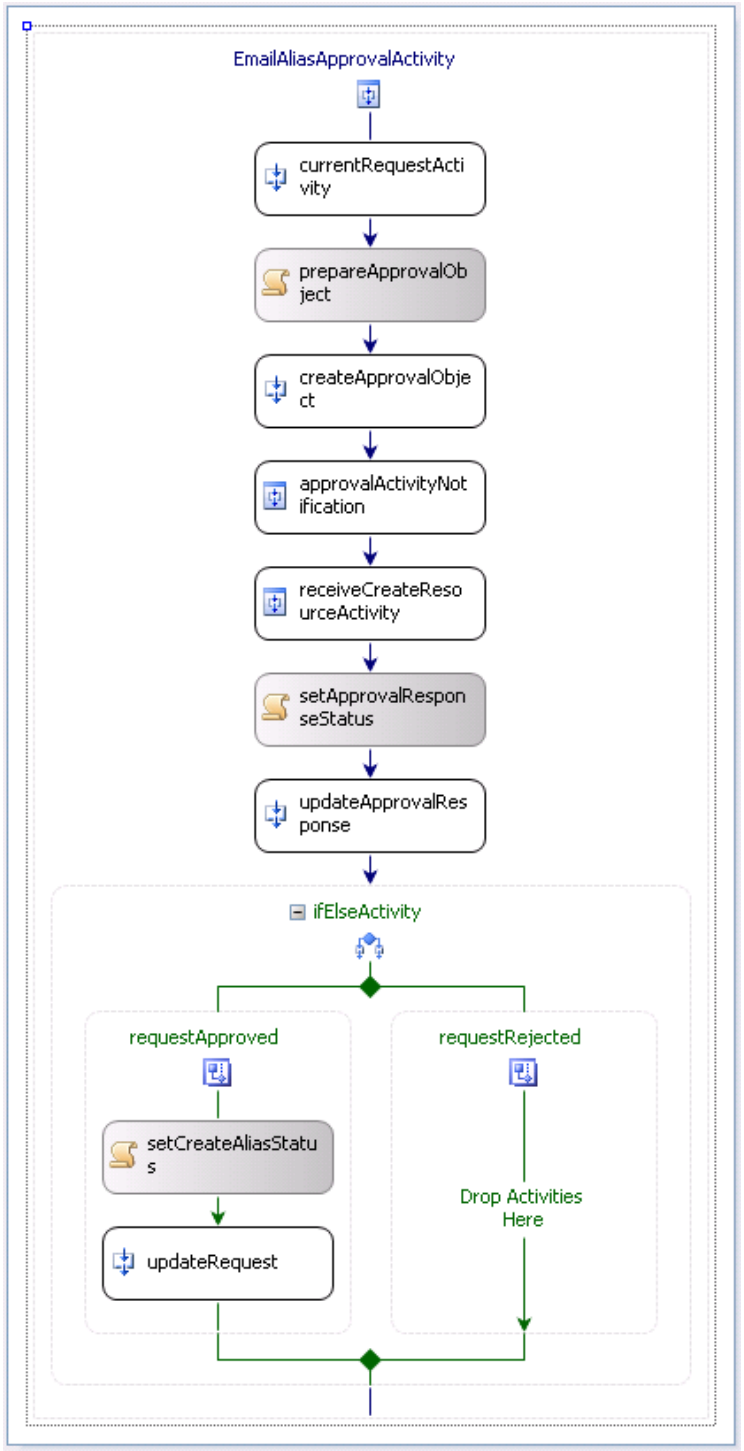
Priloga C: Skripta za izdelavo pogleda za pridobivanje podatkov identitet v HRM sistemu

```
CREATE VIEW [dbo].[MV_ObjectValues]
AS
SELECT CAST (UID_OrgUnit AS NVARCHAR(200)) AS ObjectID, Attribute, StringValue, NULL IntValue
FROM
  (SELECT  UID_OrgUnit,   CAST (ou.Code AS NVARCHAR(200)) Code,
    CAST (ou.Name AS NVARCHAR(200)) Name,
    ou.Description Description,
    CAST ((SELECT UID_OrgUnit FROM OrgUnit WHERE ID_OrgUnit = ou.ID_OrgUnitParent) AS NVARCHAR(200)) OrgUnitParentID
  FROM         dbo.OrgUnit ou) p
UNPIVOT
  (StringValue FOR Attribute IN
    (Code, Name, Description, OrgUnitParentID)) AS unpvt
UNION
SELECT CAST (UID_Employee AS NVARCHAR(200)) AS ObjectID, Attribute, StringValue, NULL
FROM
  (SELECT  employee.UID_Employee,
     CAST (employee.PersonnelNumber AS NVARCHAR(200)) PersonnelNumber,
     CAST (employee.FirstName AS NVARCHAR(200)) FirstName,
     CAST (employee.LastName AS NVARCHAR(200)) LastName,
     CAST (employee.FirstName + ' ' + Employee.LastName AS NVARCHAR(200)) DisplayName,
     CAST (employee.TelephoneNumber AS NVARCHAR(200)) TelephoneNumber,
     CAST (employee.MobileNumber AS NVARCHAR(200)) MobileNumber,
     CAST (employee.FaxNumber AS NVARCHAR(200)) FaxNumber,
     CAST (employee.[Address] AS NVARCHAR(200)) [Address],
     CAST (employee.PostCode AS NVARCHAR(200)) PostCode,
     CAST (employee.City AS NVARCHAR(200)) City,
     CAST (dbo.Country.Name AS NVARCHAR(200)) Country,
     CONVERT(nvarchar(196), employee.EmployedFrom, 126) + '.000' AS EmployedFrom,
     CONVERT(nvarchar(196), employee.EmployedTo, 126) + '.000' AS EmployedTo,
     CAST (dbo.OrgUnit.UID_OrgUnit AS NVARCHAR(200)) OrgUnitID,
     CAST (dbo.JobTitle.Name AS NVARCHAR(200)) JobTitle,
     CAST (dbo.JobTitle.UID_JobTitle AS NVARCHAR(200)) JobTitleID,
     CAST( managers.UID_Employee AS NVARCHAR(200)) ManagerID
  FROM         dbo.Employee employee INNER JOIN
    dbo.Country ON employee.ID_Country = dbo.Country.ID_Country INNER JOIN
    dbo.OrgUnit ON employee.ID_OrgUnit = dbo.OrgUnit.ID_OrgUnit INNER JOIN
    dbo.JobTitle ON employee.ID_JobTitle = dbo.JobTitle.ID_JobTitle LEFT OUTER JOIN
    dbo.Employee managers ON employee.ID_Manager = managers.ID_Employee) p
UNPIVOT
  (StringValue FOR Attribute IN
    (PersonnelNumber, FirstName, LastName, DisplayName, TelephoneNumber, MobileNumber, FaxNumber,
    [Address], PostCode, City, Country, EmployedFrom, EmployedTo, OrgUnitID,
    JobTitle, JobTitleID, ManagerID)) AS unpvt

GO
```

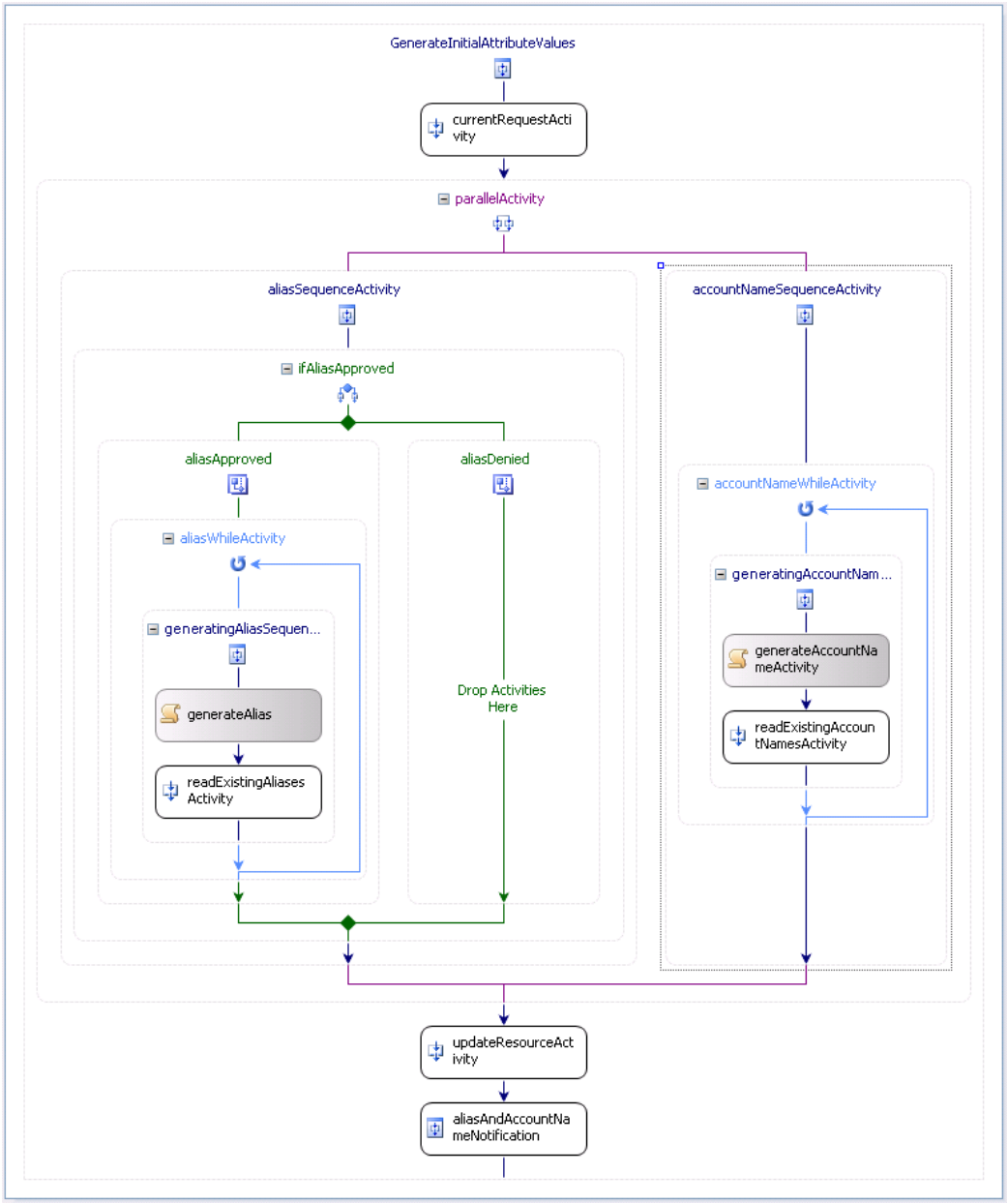
Priloga D: Razširjena pravila upravljavskega agenta *AD MA Initial Load*

```
void IMASynchronization.MapAttributesForImport( string FlowRuleName, CSEntry csentry, MVEntry mventry)
{
    long groupType;

    switch (FlowRuleName)
    {
        case "cd.user:employeeID->mv.person:employeeType":
            string personnelNumber = csentry["employeeID"].StringValue;

            if (string.IsNullOrEmpty(personnelNumber))
                mventry["employeeType"].StringValue = "Contractor";
            else
                mventry["employeeType"].StringValue = "Full Time Employee";
            break;
        case "cd.group:groupType->mv.group:type":
            groupType = csentry["groupType"].IntegerValue;

            if ((groupType & 0x80000000) == 0x80000000)
                mventry["type"].StringValue = "Security";
            else
                mventry["type"].StringValue = "Distribution";
            break;
        case "cd.group:groupType->mv.group:scope":
            groupType = csentry["groupType"].IntegerValue;

            if ((groupType & 0x2) == 0x2)
                mventry["scope"].StringValue = "Global";
            else if ((groupType & 0x4) == 0x4)
                mventry["scope"].StringValue = "DomainLocal";
            else if ((groupType & 0x8) == 0x8)
                mventry["scope"].StringValue = "Universal";
            else
                throw new IndexOutOfRangeException("Unknown group scope");

            break;
        default:
            throw new EntryPointNotImplementedException();
    }
}
```

Priloga E: Delovni tok potrjevanja poštnega predala



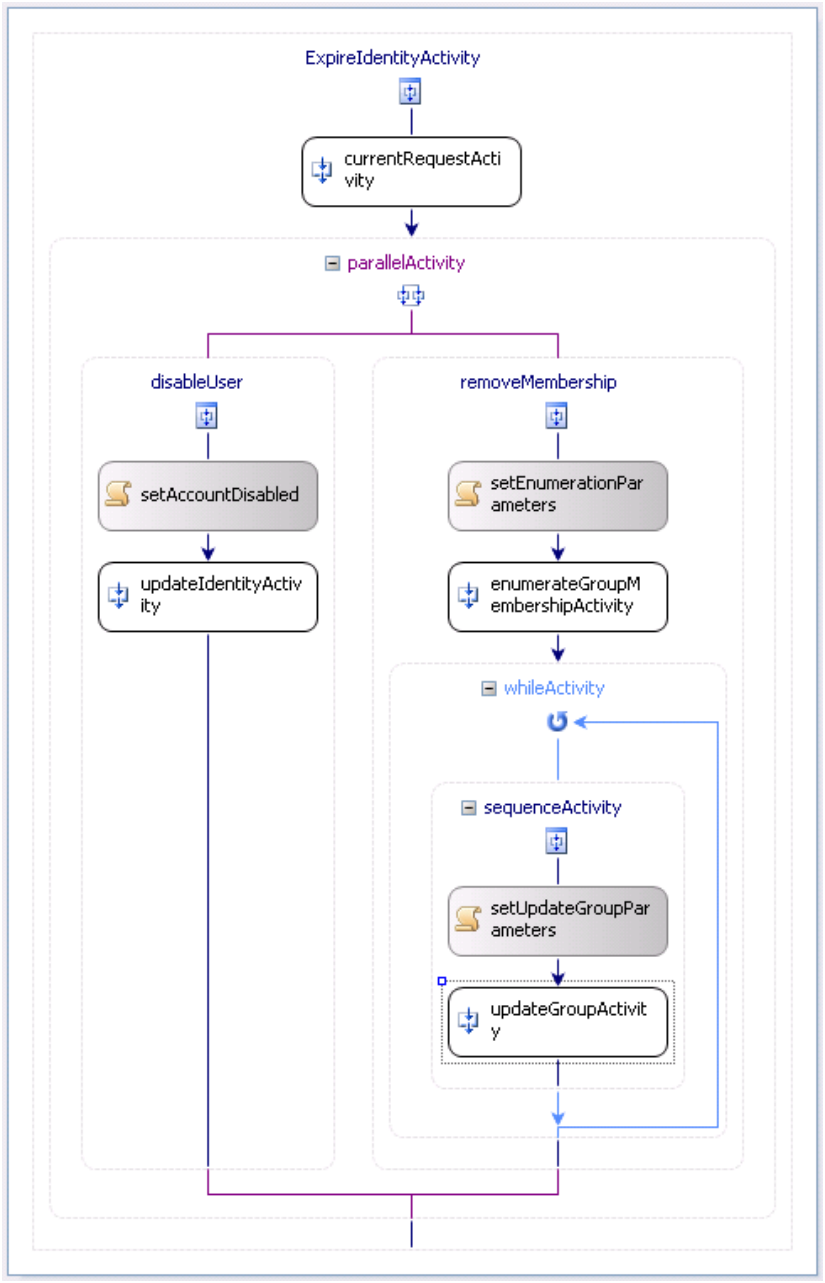
Priloga F: Delovni tok za izdelavo poštnega vzdevka in uporabniškega imena



Priloga G: Programska koda aktivnosti *aliasWhileActivity* in *generateAlias*, ki vsebujeta logiko za izdelavo poštnega vzdevka

```
private void IsEmailAliasUnique(object sender, ConditionalEventArgs e)
{
    e.Result = true;

    //preveri ali je bil alias ze generiran
    if (!string.IsNullOrEmpty(alias))
    {
        //preveri rezultate poizvedbe obstojecih aliasov
        bool result = readExistingAliasesActivity_TotalResultsCount == 0;

        //ce alias ne obstaja, ga dodaj identiteti
        if (result)
        {
            updateResourceActivity.ResourceId = currentRequestActivity.CurrentRequest.Target.GetGuid();
            updateResourceActivity.UpdateParameters = new UpdateRequestParameter[]
            {
                new UpdateRequestParameter("MailNickname", UpdateMode.Modify, alias)
            };
        }

        e.Result = !result;
    }
}

private void GenerateMailAlias(object sender, EventArgs e)
{
    //ce alias se ne obstaja, ga generiraj s sestavljanjem imena in priimka
    if (string.IsNullOrEmpty(alias))
    {
        string firstName = string.Empty;
        string lastName = string.Empty;

        ReadOnlyCollection<CreateRequestParameter> requestParameters =
            currentRequestActivity.CurrentRequest.ParseParameters<CreateRequestParameter>();

        //najdi ime in priimek med parametri
        foreach (CreateRequestParameter requestParameter in requestParameters)
        {
            if (requestParameter.PropertyName == "FirstName")
                firstName = requestParameter.Value.ToString();
            else if (requestParameter.PropertyName == "LastName")
                lastName = requestParameter.Value.ToString();
        }

        //nastavi osnovni alias
        baseAlias = string.Format("{0}.{1}", firstName, lastName);

        //odstrani vse sumnike
        baseAlias = baseAlias.ToLower().Replace('č', 'c').Replace('š', 's').Replace('ž', 'z');
    }

    //ce alias ze obstaja, mu dodaj naslednjo zaporedno stevilko
    if (index == 0)
        alias = baseAlias;
    else
        alias = string.Format("{0}{1}", baseAlias, index);

    index++;

    //varnostno preverjanje pred neskoncno zanko
    if (index > 10)
        throw new IndexOutOfRangeException("Too many possible aliases.");

    //nastavi filter za poizvedbo iskanja obstojecih identitet s tem aliasom
    readExistingAliasesActivity_XPathFilter = string.Format("/Person[MailNickname = '{0}']", alias);
}
```

Priloga H: Delovni tok onemogočanje uporabniškega računa in odstranjevanje dostopa



8 Viri

[1] American National Standard for Information Technology, *Role Based Access Control*, 2004

[2] M. Bergman, J. Cooper, »Identity Management for the Rest of Us: How to Grow a New Infrastructure«, predstavljeno na konferenci *Educause Mid-Atlantic Regional Conference*, Baltimore, ZDA, 2006

[3] J.L. Camp, »Digital Identity«, *IEEE Technology and Society Magazine*, št. 3, zv. 23, 2004

[4] D. Ferraiolo, R. Kuhn, R. Sandu, *The NIST Model for Role-Based Access Control: Towards A Unified Standar*, 2000

[5] IT Governance Institute, *CobiT 4.1*, 2008

[6] META Group, *The Value of Identity Management: How securing identity management provides value to the enterprise*, 2002

[7] FIM 2010 Technical Overview. Dostopno na: http://technet.microsoft.com/en-us/library/ff621362(WS.10).aspx

[8] Office of Government Commerce, *ITIL Service Operation*, The Stationery Office, 2007

[9] K. Tracy, »Identity Management Systems«, *IEEE Potentials*, št. 6, zv. 27, str. 34-37, 2008

[10] P. Windley, *Digital Identity*, O'Reilly Media, 2005

[11] COBIT. Dostopno na: http://en.wikipedia.org/wiki/COBIT#COBIT_Version_4

[12] Digital Identity. Dostopno na: http://en.wikipedia.org/wiki/Digital_identity

[13] Directory Services. Dostopno na: http://en.wikipedia.org/wiki/Directory_services

[14] Identity Management. Dostopno na: http://en.wikipedia.org/wiki/Identity_management.

[15] Information Technology Infrastructure Library. Dostopno na: http://en.wikipedia.org/wiki/Information_Technology_Infrastructure_Library