IMiS®/ARChive Server
Priročnik

Verzija 9.8.1710

IMAGING SYSTEMS
Imaging Systems, informacijski sistemi, d.o.o.
Brnčičeva 41g
Ljubljana

KAZALO VSEBINE

KAZALO SLIK

V nadaljevanju je uporabniku na voljo seznam slik uporabljenih v tem priročniku.

KAZALO TABEL

V nadaljevanju je uporabniku na voljo seznam tabel uporabljenih v tem priročniku.

1 PREDGOVOR

Predgovor opisuje vsebino in obliko IMiS®/ARChive Server dokumentacije in nudi koristne nasvete iz tehničnega in vsebinskega področja uporabe produkta.

1.1 O dokumentaciji

Dokumentacija opisuje arhitekturo strežnika, posamezne gradnike objektov, mehanizme za zagotavljanje avtentičnosti in varnosti, ter postopke nameščanja, konfiguriranja in upravljanja arhivskega strežnika IMiS®/ARChive Server.

1.2 Ciljno občinstvo

Dokumentacija je namenjena izkušenim sistemskim administratorjem z dobrim poznavanjem različnih izvedb operacijskih sistemov Linux ter izkušnjami pri nameščanju, konfiguriranju in upravljanju informacijskih sistemov.
Postopke mora poznati vsak administrator, ki je določen za namestitev in vzdrževanje strežnika IMiS®/ARChive Server.

1.3 Konvencije

V dokumentaciji uporabljamo različne stile in načine zapisa pomembnih informacij, ki so povzete v spodnji tabeli:

Vrsta pisave	Namen uporabe
Navadno	Osnovno besedilo v dokumentaciji
Navadno krepko	Naslovi poglavij v dokumentaciji (nivoji 1-6)
Navadno podčrtano	Dodatne možnosti izbora podpoglavij znotraj posameznega nivoja
»Navadno«	Nazivi funkcij ali akcij v okviru možnosti izbora
Navadno ležeče	Prehodi na druga poglavja
Enakomerna širina znakov (Monospace)	Prikaz konzolnih ukazov, datotek, imenikov, ...
Enakomerna širina znakov (Monospace Bold)	Prikaz uporabniškega vnosa

Tabela 1: Uporaba različnih stilov v dokumentaciji

1.4 Kratice

Spodnja tabela opisuje kratice, uporabljene v tekstu in grafikah tega dokumenta:

Kratica	**Opis**
AES	Advanced Encryption Standard (napreden algoritem šifriranja)
AIP	Archival Information Package (»povzetek« vsebine in metapodatkov entitete združen v XML dokument)
ACL	Access Control List (lista dostopnih pravic)
CA	Certificate Authority (zaupanja vreden izdajatelj digitalnih potrdil)
CIFS	Common Internet File System (komunikacijski protokol za podatke, tiskanje in druge servisne storitve v omrežju)
CRL	Certificate Revocation List (seznam preklicanih digitalnih potrdil)
CRM	Customer Relationship Management (sistem za upravljanje odnosov s strankami)
DMS	Document Management System (sistem za upravljanje z dokumenti)
EML	E-Mail Message (format za shranjevanje elektronskih sporočil)
ERP	Enterprise Resource Planning (poslovno informacijski sistemi)
ERS	Evidence Record Syntax (oblika podatkov za zagotavljanje dolgoročne stabilnosti časovnih žigov)
FIPS	Federal Information Processing Standard (standard za procesiranje informacij)
HA	High Availability (značilnost sistema - visoka razpoložljivost)
HMAC	Hash-based Message Authentication Code (način avtentikacije, ki vključuje funkcijo prstnega odtisa, ki je namenjena preverjanju integritete podatkov in avtentikacije)
HSM	Hierarchical Storage Management (koncept shranjevanja objektov - hierarhično arhiviranje dokumentov)
ISUD	Informacijski sistem za upravljanje z dokumenti

Kratica	**Opis**
LTANS	Long-Term Archive and Notary Services (skupina tehničnih postopkov in funkcionalnosti za zagotavljanje vzdržnosti arhiviranja skozi daljše časovno obdobje; tudi skupina, ki te standarde določa)
MIME	Multipurpose Internet Mail Extensions
NAS	Network Attached Storage
NAT	Network Address Translation (prevedba omrežnih naslovov, postopek skrivanja privatnih naslovov)
OCSP	Online Certificate Status Protocol (spletni protokol za preverjanje statusa veljavnosti digitalnih potrdil)
PDF/A	Portable Document Format) (format za dolgoročno hrambo dokumentov - .pdf)
POSIX	Portable Operating System Interface (standardni vmesnik med aplikacijsko programsko opremo in operacijskim sistemom)
RAID 5	Redundant Array of Independent Disks (sistem za uporabo in organizacijo diskovnih pogonov)
RFC	Request for Comments (tehnični in organizacijski dokument, specifikacija, javni dokument, namenjen izmenjavi mnenj o opisani tematiki)
RHEL	Red Hat Enterprise Linux (Linux strežniška platforma podjetja RedHat)
RPM	RedHat Package Manager (aplikacija za upravljanje nameščene programske opreme na platformi Linux RedHat; tudi namestitveni paket programske opreme za isto imenovano aplikacijo za upravljanje)
SCSI/SAS	Serial Attached Small Computer System Interface (standardni vmesnik diskovnega pogona)
RSA	Ronald **R**ivest, Adi **S**hamir, Leonard **A**dleman (algoritem za šifriranje z javnim ključem)
SLES	SUSE Linux Enterprise Server (Linux strežniška platforma podjetja Novell)
SRP	Secure Remote Password (šifrirni protokol za varno avtentikacijo uporabnika)
SHA	Secure Hash Algorithm (algoritmi za izračun prstnega odtisa vsebine)
TCP/IP	Transmission Control Protocol / Internet Protocol (družina omrežnih protokolov)
TSA	Time Stamp Authority (agencija za izdajo kvalificiranih časovnih žigov)

Kratica	Opis
TIFF	Tagged Image File Format (format za dolgoročno hrambo dokumentov)
URI	Uniform Resource Identifier (enotni označevalnik vira, ki predpisuje algoritem za pretvorbo AIP v normalizirano obliko)
X.509	(ITU-T standard za uporabo infrastrukture javnih ključev)
XML	Extensible Markup Language (označevalni jezik za hierarhično strukturiranje podatkov v obliki tekstovne datoteke)
XMLDSIG	XML Signature (specifikacija, ki določa XML zapis za elektronske podpise)
XSD	XML Schema Definition (priporočilo W3C za opredelitev strukture XML dokumentov)
W3C	World Wide Web Consortium (organ za standardizacijo ustreznih spletnih tehnik)
WORM	Write Once Read Many (princip shranjevanja objektov v arhivskem strežniku)
Container Vsebnik	Generičen element za shranjevanje podatkov

Tabela 2: Uporaba kratic v dokumentaciji

Spodnja tabela opisuje pojme, uporabljene v tekstu in grafikah tega dokumenta:

Pojem	Opis
IMiS®/ARChive Server	IMiS®/ARChive Storage Server strežnik (arhivski strežnik za shranjevanje objektov)
IMiS®/Scan	IMiS®/Scan odjemalec (IMiS® odjemalec za skeniranje papirnih dokumentov)
IMiS®/Storage Connector	IMiS®/Storage Connector vmesnik (vmesnik za prenos arhiviranih objektov med aplikativnim in arhivskim strežnikom)
IMiS®/View	IMiS®/View odjemalec (IMiS® odjemalec za prikazovanje skeniranih dokumentov)

Tabela 3: Seznam uporabljenih pojmov v dokumentaciji

2 UVOD

2.1 Predstavitev

IMiS®/ARChive Server je programski modul za varno dolgoročno hrambo vseh vrst objektov elektronskega izvora ali digitaliziranih preko različnih postopkov digitalizacije (npr. skeniranja). Gre za celovito rešitev za zagotavljanje varne dolgoročne hrambe dokumentarnega gradiva in s tem: trajnosti, nespremenljivosti, urejenosti, dokazljivosti izvora gradiva in dostopnosti gradiva ves čas trajanja hrambe. Je skalabilen, saj lahko arhiviramo praktično neomejene količine binarnih objektov. Hitrost posredovanja in prikazovanja arhiviranih objektov je praktično neodvisna od velikosti arhiva.

S tehnološkimi prijemi, ki jih zagotavljamo, je onemogočeno brisanje ali spreminjanje arhiviranega gradiva. Z upravljanjem dostopnih pravic zagotavljamo varnost dostopa kadarkoli in od kjerkoli. Za zagotavljanje avtentičnosti arhiviranega gradiva uporabljamo ustrezna dokazila (prstni odtis, elektronski podpis z digitalnim potrdilom, časovni žig). Struktura, ustvarjanje in preverjanje teh dokazil je izvedeno po najsodobnejšem standardu ERS. V revizijski sledi so zabeleženi vsi dostopi, vse poizvedbe in vse spremembe na strežniku IMiS®/ARChive Server.

Elektronski arhiv ima v primerjavi s fizični arhivom številne prednosti. Obvladovanje arhiviranega gradiva v elektronski obliki je lažje in bistveno bolj učinkovito. Samo iskanje gradiva po elektronskem arhivu je neprimerno hitrejše.

Na osnovi preprostega iskalnega niza lahko v trenutku dostopamo do gradiva, ki bi ga sicer s težavo v fizičnem arhivu. K večji učinkovitosti bistveno prispeva uporaba naprednih tehnologij (OCR, FTS), saj omogoča iskanje po polnem besedilu dokumentov.

Dostop do arhiviranega gradiva je hkrati na voljo neomejenemu številu uporabnikov s pravicami iz različnih lokacij in aplikacij. Čas od pojavitve potrebe po nekem gradivu do dostopa do informacije je krajša kot pri fizičnem gradivu.

Stroški hrambe in upravljanja s fizičnim gradivom naraščajo s količino hranjenega gradiva in hitro presežejo stroške elektronskega arhiviranja.

Pogoste so tudi poškodbe fizičnega gradiva, ki lahko povzročijo nenadomestljivo izgubo ključnih informacij. Za razliko od arhiviranega gradiva se fizično gradivo lahko založi, izgubi, ali pa pride celo do namerne odtujitve.

Strežnik IMiS®/ARChive Server lahko deluje kot samostojen elektronski arhiv ali se preko programskega vmesnika povezuje z različnimi aplikacijskimi strežniki.
Zaradi visoko zmogljivih povezovalnih modulov lahko aplikacije tretjih ponudnikov izvajajo vse postopke arhiviranja, izvajajo postopke življenjskega cikla arhiviranih entitet, upravljajo z dostopnimi pravicami, izvajajo poizvedbe (iskanja) po metapodatkih in polnem besedilu, itd.
Arhitektura arhivskega sistema je fleksibilna in omogoča postavitev večjega števila arhivskih strežnikov in vzpostavitev različnih vlog v hierarhiji.

2.2 Verzioniranje in označevanje

Označevanje verzij programskega modula IMiS®/ARChive Server je osnovano na sekvenčni shemi s štirimi ločenimi numeričnimi identifikatorji (MAJOR, MINOR, RELEASE, BUILD) in končnim identifikatorjem tarčne arhitekture procesorja (ARCHITECTURE) (Linux standard). Takšna shema je v svetu tudi najširše sprejeta.
Imenu namestitvene RPM datoteke so dodani še produktu specifični dodatni atributi, ki vsebujejo unikaten identifikator arhiva (ARCHIVEID), identifikator vsebovane baze (DATABASE) in identifikator namestitvene platforme (PLATFORM). Ti se ne beležijo v RPM podatkovno bazo. Na voljo le zaradi lažjega ločevanja in umeščanja namestitvene datoteke:imisarc.MAJOR.MINOR.RELEASE-BUILD.ARCHIVEID.DATABASE.PLATFORM.
ARCHITECURE.rpm

Primer: imisarc.9.7.1610-600.0001.rdm.el4.i386.rpm

Shema je torej sestavljena iz imena IMiS® modula (»imisarc«) in naslednjih elementov:

- MAJOR: Identifikator na mestu MAJOR označuje glavno/veliko verzijo produkta, ki se spreminja skladno z vsakokratno arbitrarno odločitvijo glede na obseg izvedenih sprememb in funkcionalnosti. Na tem mestu se identifikator spreminja najredkeje in v primeru spremembe označuje veliko razliko v produktu glede na predhodno izdane verzije z manjšo MAJOR verzijo. Identifikator ima nabor vrednosti od 1-n, je zvezen in se izključno povečuje.
- MINOR: Identifikator na mestu MINOR označuje manjšo verzijo produkta, ki se spreminja skladno z vsakokratno arbitrarno odločitvijo glede na obseg izvedenih sprememb, funkcionalnosti in popravkov. Identifikator se spreminja pogosto in označuje manjše spremembe in popravke v okviru iste generacije produkta, ki jo označuje neka MAJOR verzija. Nabor vrednosti je od 1-n,
  ni zvezen in se z vsako spremembo MAJOR verzije postavi na izhodišče (1).

- RELEASE: Pri tem identifikatorju gre v nasprotju z arbitrarnim naborom vrednosti, ki velja po svetu, pri našem za specifiko, saj označuje časovno komponento izdaje produkta po shemi »LLMM«. MM označuje mesec izdaje (nabor 01-12), LL označuje zadnji dve številki leta; primer izdaja produkta »oktober 2017« je v RELEASE identifikatorju označena kot 1710.
- BUILD: Identifikator na tem mestu označuje zaporedno unikatno številko izgradnje produkta, ki se nikoli ne ponovi. V primeru minimalne spremembe produkta znotraj enega meseca lahko pride le do zamenjave tega identifikatorja medtem, ko vsi ostali ostanejo enaki. Nabor vrednosti je od 1-n, ni zvezen in se izključno povečuje.
- ARCHIVE ID: Vsaki instanci nameščenega produkta je dodeljen unikaten identifikator, ki se uporablja tudi v segmentu zaščite produkta in objektnih identifikatorjev. Strankam dostavljamo RPM datoteke, ki niso prenosljive in namenjene le konkretni instanci produkta. Nabor vrednosti je od 0001 – nnnn.
- DATABASE: Identifikator označuje s katero bazo je produkt distribuiran. Nabor vrednosti je v času izdaje te verzije BDB in RDM.
- PLATFORM: Identifikator označuje tip namestitvene platforme, kateri je namestitveni paket namenjen. Nabor vrednosti sles8, el3 in el4,
  označuje tarčno platformo pri čemer je za neko diskretno platformo potrebno poiskati ustrezen ekvivalent glede na okolje zahtevanih sistemskih knjižnic.
  Stranki pri tej izbiri pomagamo, lahko pa si pomaga tudi z uporabo kompatibilnih sistemskih knjižnic, ki omogočajo izvajanje platformsko starejšega produkta tudi na novejših platformah (npr. el3 namestitveni paket + kompatibilne knjižnice za rhel3 na rhel4 platformi).

Kriteriji za določanje arbitrarno določljivih identifikatorjev so opredeljeni v notranjih pravilih družbe in so predmet postopka upravljanja s spremembami.

2.3 Funkcionalnosti

- strukturirano arhiviranje vsebin elektronskega izvora ali digitaliziranih preko postopka skeniranja;
- hramba metapodatkov arhiviranih vsebin preko strukturiranih atributov, povezanih v metapodatkovne sheme, določene v okviru hierarhičnih entitetnih predlog;
- hramba sistemsko pomembnih atributov, ki vplivajo na življenjski cikel hranjenega gradiva;
- hramba metapodatkov in elektronske pošte;
- upravljanje z metapodatki hrambe fizičnega gradiva;
- hramba nespremenljivih metapodatkov prenesenega gradiva;
- iskanje po metapodatkih preko poizvedb z logičnimi in prednostnimi operatorji;
- iskanje hranjenih vsebin po elementih njihove vsebine (iskanje po polnem besedilu);
- priklic vsebin preko uporabe zunanjih in notranjih identifikatorjev ter z uporabo klasifikacijske oznake;
- upravljanje s hierarhičnim načrtom razvrščanja gradiva;
- razvrščanje arhiviranih vsebin v načrtu razvrščanja gradiva z možnostjo dodajanja, brisanja, spreminjanja in premeščanja entitet;
- upravljanje z stopnjami tajnosti hranjenega gradiva;
- upravljanje z hierarhičnimi pravicami dostopa do hranjenih vsebin vse do nivoja posamičnega metapodatka;
- zagotavljanje avtentičnosti hranjenega gradiva preko ustvarjanja in dolgoročnega vzdrževanja dokaznih elementov o nespremenjenosti hranjenega gradiva (LTANS);
- imeniške storitve za podporo prijavi uporabnika, upravljanje z dostopnimi pravicami do hranjenih vsebin in razširitev podatkov entitet imenika v primeru uporabe v metapodatkovnih atributih;
- beleženje in vzdrževanje revizijske sledi dejanj nad hranjenimi vsebinami ter zaščiten vpogled v sledi s strani za to pooblaščenih oseb;
- zajem in preverjanje avtentičnosti vsebin preko preverjanja veljavnosti vgrajenih elektronskih podpisov;
- zmožnost uporabe IPv4 in IPv6 komunikacijskih skladov;
- šifriranje povezav med storitvijo in njenimi odjemalci;
- visoko skalabilen arhivski sistem s praktično nezaznavnim zamikom obdelave transakcij v primeru velikih inventarjev arhiviranih vsebin (nad 10 milijoni entitet).

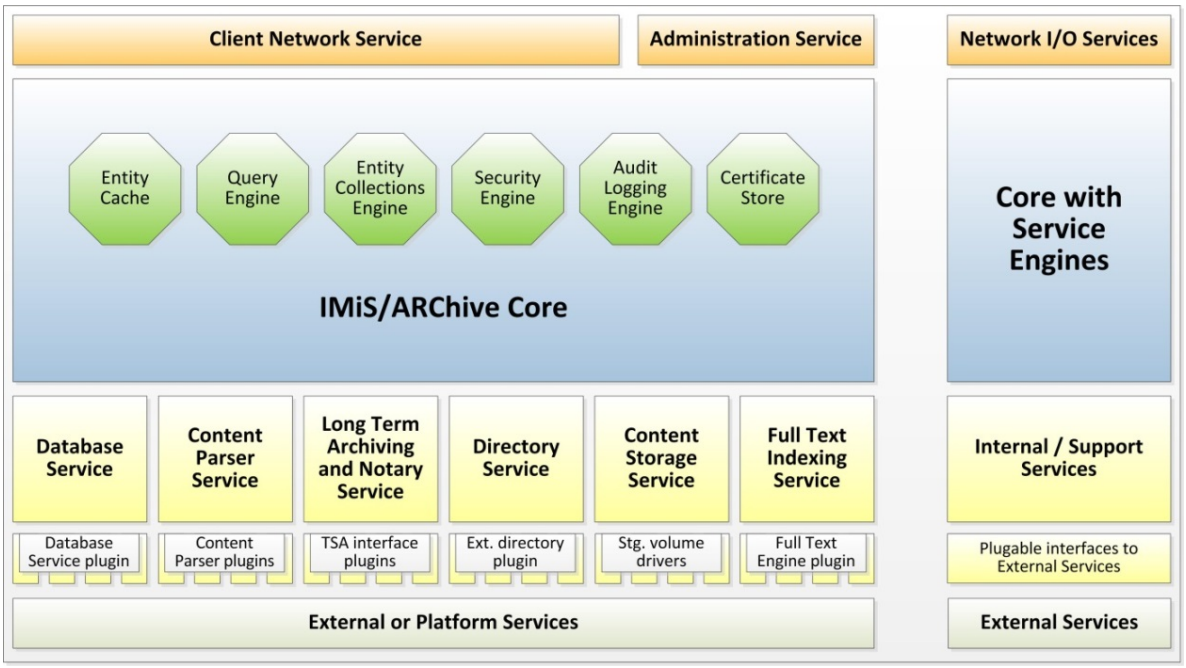
3 TEHNIČNA DOKUMENTACIJA

3.1 Arhitektura strežnika

IMiS®/ARChive Server predstavlja strežniško komponento sistema za upravljanje z gradivom v elektronski obliki IMiS®. Njegova modularna zasnova omogoča visok nivo prilagoditve na različne informacijske sisteme, katere danes večja podjetja uporabljajo za svoje poslovanje.

Jedro sistema predstavlja logično jedro, ki povezuje:

- Podporne servise, povezane z zunanjimi viri informacij ali samostojno delujoče v okviru instance strežnika.
- Notranje logične servise, ki jedru zagotavljajo podporo odločitvam glede sicer prilagodljive poslovne logike in prispevajo k visoki zmogljivosti in propustnosti sistema.
- Omrežne servise, ki na eni strani skrbijo za izmenjavo informacij z odjemalci strežnika in jedrom ter na drugi strani nudijo točko integracije z zalednimi aplikacijami.
  Slednje z jedrom izmenjujejo statistične informacije o delovanju in nastavitvene informacije.



Slika 1: Shematični prikaz arhitekture strežniške komponente IMiS®/ARChive Server

Fizično strežnik predstavljajo trije procesi:

- Glavni proces »iarcd«: povezovalni proces, ki upravlja s podpornimi procesi:
  - sprejema zahteve po novih povezavah s strani odjemalcev;
  - izbira logične procese za obdelavo odjemalskih povezav;
  - zbira statistične podatke o delovanju strežnika.
- Storage Volume Manager »iavol«: proces za upravljanje z volumni arhiviranih vsebin. Proces v svojih nitih skrbi za optimalno komunikacijo z viri stalnih diskovnih kapacitet glede na njihov tip in arhitekturo. Gre za podporni servis drugim procesom IMiS®/ARChive strežnika, ki njegove storitve potrebujejo za svoje operacije (npr. logični proces).
- Logični proces »iarcd«: proces s svojimi nitmi skrbi za operacije, ki jih odjemalci zahtevajo od arhiva. Je lastnik niti, ki vsaka zase izvaja nekatere podporne servise kot so:
  - indeksiranje vsebin po polnem besedilu;
  - vzdrževanje avtentičnosti arhiviranih vsebin;
  - opcijsko sinhronizacijo imenika z zunanjimi viri imeniških storitev, ...

Logični nivo strežnika sestavljajo samostojne ali med seboj odvisne storitve, ki jedru nudijo:

- podporo pri odločitvah glede načina obdelave dogodkov;
- medpomnjenje aktualnih objektov za katere je smiselno, da so naloženi v spominu;
- podporo odločanju glede dostopov;
- podporo vzdrževanju revizijske sledi;
- komunikacijski nivo storitve, ki izmenjuje informacije z odjemalci storitve, ...

Storitve so strukturirane na:

- Nivo vmesnikov do storitev fizičnega nivoja platforme ali zunanjih storitev, ki preko vtičnikov omogoča uporabo virov informacij, ki niso del sistema.
- Nivo podpornih storitev, ki jedru sistema nudijo nujne in ne-nujne storitve za njegovo delo. Omogočajo samostojno izvajanje določenih storitev, ki niso neposredno vezane na jedro. Na eni strani komunicirajo z jedrom, na drugi strani pa preko vtičnikov do virov platforme in zunanjih storitev.
- Jedro sistema z notranjimi storitvami in pogoni, ki jedru nudijo logične celote za funkcionalnosti, ki jih podpirajo. So neločljivi del jedra in z njimi razen v izjemnih primerih ni mogoče upravljati.
- Komunikacijski nivo, ki skrbi za izmenjavo podatkov med jedrom in odjemalci sistemi. Podpira različne omrežne protokole, šifriranje prometa in standarde kodiranja informacij.

Nadaljevanje poglavja obravnava posamične storitve v smislu njenega pomena za delovanje strežnika.

Database service [storitev podatkovne baze]

Pomen: Kritičen

Opis: Storitev sistemu nudi nivo podatkovne zbirke tabel in indeksov, potrebnih za delovanje. Storitev je zasnovana abstraktno in deluje preko vsakokratnega vtičnika podatkovne baze, specifičnega za tehnologijo in ponudnika podatkovne zbirke.

Trenutno podprti vtičniki podatkovnih zbirk so:

- Raima Embedded Database: vgrajena baza podatkov, do katere uporabniki s pravicami dostopajo lokalno preko knjižnic okolja za delo z bazo.

Sistem v podatkovno zbirko shranjuje večino informacij, povezanih s hranjenimi entitetami, razen hranjenih vsebin, ki so shranjene v okviru druge storitve in na drugih nosilcih.

Content Parser Service [storitev obdelave hranjenih vsebin]

Pomen: Obvezen

Opis: Sistem storitev se uporablja za obdelavo vsebin, ki jih odjemalci arhivirajo.

Zasnovan je na principu vtičnikov, ki zna vsak zase obdelati določeno vrsto vsebine, glede na njen »ContentType«(MIME tip).

Vsak vtičnik lahko storitvi nuditi naslednje funkcionalnosti:

- izvoz besedila vsebine;
- izvoz metapodatkov vsebine;
- izvoz vgrajenih elektronskih podpisov;
- izvoz digitalnih potrdil vgrajenih elektronskih podpisov;
- preverjanje veljavnosti vgrajenih elektronskih podpisov.

Sistem uporablja funkcionalnosti:

- obdelave vsebin za potrebe preverjanja veljavnosti vsebin;
- zajema vgrajenih elektronskih podpisov in z njimi povezanimi digitalnimi potrdili;
- podpore storitvi indeksiranja po polnem besedilu.

Trenutno storitev za vtičnike uporablja vgrajene tehnologije Apache projekta Tika in iText (obdelava »application/pdf« vsebin) ter lastno tehnologijo za obdelavo vsebin formata »image/tiff« in »text/xml«.

Long term Archiving and Notary Service - LTANS [sistem za zagotavljanje avtentičnosti hranjenih vsebin]

Pomen: Neobvezen

Opis: Sistem opcijsko zagotavlja izdelavo in dolgoročno vzdrževanje dokaznih elementov nespremenljivosti hranjenih vsebin. Sistem ni ključen za delovanje arhivskega sistema, v kolikor uporabnik tovrstnih storitev ne potrebuje.

Storitev zajame vse zaključene entitete in jih glede na nastavitvene parametre vključi v inventar entitet, katerim se zagotavljajo elementi avtentičnosti.

Hkrati samodejno vzdržuje že obstoječe elemente avtentičnosti in skupaj z novimi elementi gradi drevesa zgoščenih dokaznih elementov. Te ščiti s časovnimi žigi različnih ponudnikov zaupanja vrednih časovnih žigov. Pridobi jih preko vtičnikov, specifičnih za vsakokratnega ponudnika časovnega žiga. Dodatno zaščito pred zlomom veljavnosti časovnega žiga lahko zagotovimo z uporabo več vzporednih časovnih žigov različnih ponudnikov zaupanja vrednih časovnih žigov.

S tem sistem zagotovi veljavnost dokaznih elementov avtentičnosti tudi v redkih primerih preklica veljavnosti časovnih žigov pred njihovim iztekom veljavnosti.

Več informacij se nahaja v poglavju 3.6 Avtentičnost.

Directory Service [Imeniška storitev]

Pomen: Kritičen

Opis: Storitev zagotavlja podporo različnim storitvam imenika, katere uporabljajo:

- podsistem za dostopno logiko do hranjenih vsebin;
- podsistem za hrambo vrednosti metapodatkovnih atributov.

Storitev omogoča vzdrževanje atributov imeniških entitet (podatki o uporabniku, njegovih varnostnih komponent, ki se uporabljajo pri prijavi/avtentikaciji, podatki o skupinah, katerim uporabnik pripada, ...) in članstva entitet imenika (uporabniki, skupine) v različnih skupinah.

Opcijsko se preko vtičnikov lahko imeniška storitev poveže z zunanjimi viri imeniških storitev v smislu konsolidiranih in centralno orientiranih imeniških storitev, ki se distribuirano uporabljajo v okviru različnih sistemov.

Več informacij se nahaja v poglavju 3.7 Imeniške storitve.

Content Storage Service [storitev hrambe hranjenih vsebin]

Pomen: Obvezen

Opis: Sistem storitev uporablja za trajno hrambo digitalnih vsebin, ki jih odjemalci arhivirajo. Zasnovan je na principu vtičnikov, ki zna vsak zase optimalno upravljati z različnimi vrstami nosilcev digitalnih vsebin. Tako sistem loči med različnimi vrstami lokalnih nosilcev vsebin in različnimi vrstami oddaljenih nosilcev vsebin. Tehnologijam primerno, jedru zagotavlja enoten nivo hrambe digitalnih vsebin. Inventar arhiviranih vsebin je hranjen po principih hierarhične hrambe (http://en.wikipedia.org/wiki/Hierarchical_storage_management) na logičnem nivoju storitve za optimalno izrabo in nizke stroške arhiviranja. Brez pravilno nameščene storitve sistem sicer lahko deluje in hrani atribute entitet, digitalnih vsebin pa ne.

Full Text Indexing Service [storitev indeksiranja po polnem besedilu hranjenih vsebin]

Pomen: Neobvezen

Opis: Storitev sistemu in njegovim uporabnikom zagotavlja funkcionalnost iskanja po besedilu hranjenih vsebin. Zasnovana je na modelu vtičnikov, ki sistemu na abstraktnem nivoju nudijo poenoten dostop do storitev indeksiranja vsebin in kasnejšemu iskanju po indeksu besedil hranjenih vsebin. Trenutno sistem preko vtičnika uporablja vgrajeno tehnologijo Apache projekta Lucene. V kolikor uporabnik ne potrebuje funkcije iskanja po polnem besedilu arhiviranih vsebin, je možno storitev onemogočiti, saj ni ključna za delovanje sistema.

Entity Cache [medpomnilnik odprtih entitet]

Pomen: Kritičen

Opis: Gre za notranjo storitev s katero ni mogoče upravljati. Sistem uporablja visoko zmogljiv model medpomnjenja za vse entitete, ki so v neki časovni točki odprte s strani uporabnikov ali notranje za potrebe njihove obdelave. Sistem skladno s svojim modelom notranje odpre celotno vejo entitete do korenskega nivoja in instance entitet drži odprte skladno z obstoječimi sklici na njih. Ko se zapre zadnji sklic na entiteto, se ta iz medpomnilnika odstrani. Enako velja za vse njene starševske entitete do korena. Tako sistem zagotavlja visok nivo odzivnosti za vzporedna odpiranja entitet oziroma pridobivanja podatkov iz le-teh. Celoten model medpomnjenja informacij temelji na konceptu »lenega« nalaganja informacij, vse v smeri visokega nivoja odzivnosti sistema. Informacije se namreč nalagajo pri prvi dejanski potrebi po informaciji in ne na zalogo.

V primeru spreminjanja entitet so vrednosti atributov instance za vpogled in instance za spreminjanje v spominu naložene le enkrat in se podvojijo šele ob dejanski spremembi vrednosti atributa, vse zaradi minimalne porabe spomina za medpomnilnik.

Zaradi vseh uporabljenih konceptov algoritma medpomnjenja pripomorejo k sistemu,
ki nudi visoko zmogljiv, hiter in učinkovit sistem dostopa do hranjenih vsebin ob minimalni porabi delavnega spomina.

Query Engine [podsistem za iskanje po metapodatkih]
Pomen: Obvezen
Opis: Gre za notranjo storitev, preko katere uporabniki in sam sistem išče entitete glede na vrednosti njihovih indeksiranih atributov. Sistem nudi gramatičen pogon za spremembo opisne oblike poizvedbe z njegovo logično obliko, primerno za izvajanje podsistema za iskanje po vrednostih atributov. Gramatična pravila omogočajo poljubne poizvedbe z različnimi logičnimi in prednostnimi operatorji. Sistem omogoča tudi učinkovito iskanje po praznih vrednostih atributov. Deluje v povezavi s Entity Collections storitvijo.

Entity Collections Engine [podsistem za izgradnjo ad-hoc zbirk entitet]
Pomen: Kritičen
Opis: Storitev sistemu omogoča gradnjo zbirk entitet na podlagi različnih vrst poizvedb.
Po teh ad-hoc zbirkah je možno naključno poizvedovati in pridobivati informacije o zbranih entitetah. Ta podsistem pogosto uporabljajo funkcije odjemalcev za hierarhični vpogled v drevo arhiviranih entitet (vse pod-entitete neke starševske entitete, ipd.) in podsistem za iskanje po metapodatkih, ki zgradi ad-hoc zbirko entitet na podlagi specifične poizvedbe.
Ker so lahko zbirke entitet obsežne in ker jih je lahko vzporedno odprtih več,
je ključno dobro upravljanje sistema s spominskimi viri platforme, ki so mu na voljo v smeri minimalnega vpliva na obseg porabljenega delovnega spomina.

Security Engine [podsistem za izračun dostopnih pravic do arhiviranih vsebin]
Pomen: Kritičen
Opis: Vsakokratno odpiranje entitete v sistemu povzroči ustvaritev instance entitete,
če ta ni že prisotna v medpomnilniku. Takrat se v instanco naložijo tudi shranjena pravila dostopa, ki hierarhično omogočajo visoko zmogljiv sistem določanja dostopnih pravic po modelu liste dostopnih pravic (t.i. Access Control List - ACL).
Pravice se dedujejo od nadrejenih entitet, lahko pa se tudi nadomestijo z eksplicitnimi pravicami.
Pravila za izračun so ločena od jedra zato govorimo o podsistemu.
Na pravila ni mogoče vplivati in z njimi upravljati, omogočen je le omejen nabor nastavitev, ki bistveno ne vplivajo na zaščito informacij.

Audit Logging Engine [podsistem za beleženje in vzdrževanje revizijske sledi dogodkov na arhiviranih vsebinah in njen vpogled]

Pomen: Neobvezen

Opis: Podsistem jedru sistema omogoča natančno beleženje dogodkov nad arhiviranimi vsebinami v realnem času, ki so del transakcij samih operacij.

Operacije so v sistemu z razliko od beleženja revizijskih sledi podatkovnih strežnikov logično zaokrožene, celovite, konsolidirane in za to poklicanemu osebju jasne.

Z nivojem beleženja revizijske sledi in obsegom podatkov, ki se zajemajo, je možno upravljati vendar so morebitne spremembe nastavitev zabeležene v revizijski sledi.

V kolikor uporabnik ne potrebuje funkcije beleženja revizijske sledi, jo je mogoče onemogočiti.

Certificate Store [podsistem za upravljanje z digitalnimi potrdili]

Pomen: Obvezen

Opis: Podsistem jedru sistema omogoča storitve povezane z digitalnimi potrdili, ki jih uporablja jedro in/ali z njim povezane storitve. Hrani digitalna potrdila zaupanja vrednih izdajateljev digitalnih potrdil in nudi logično podporo pri preverjanju veljavnosti različnih digitalnih potrdil, ki jih sistem pri svojem delu srečuje.

Podsistem nudi funkcionalnosti pridobivanja podatkov o preklicih digitalnih potrdil s tehnologijami CRL in OSCP. V kolikor je skladišče digitalnih potrdil zaupanja vrednih izdajateljev digitalnih potrdil prazno, operacije v smislu preverjanja veljavnosti digitalnih potrdil ne delujejo.

Client Network Service [podsistem za izmenjavo informacij med jedrom in odjemalci sistema]

Pomen: Kritičen

Opis: Podsistem jedru sistema omogoča varno izmenjavo informacij z njegovimi odjemalci. Uporabljen asinhron omrežni model omogoča visoko propustno in učinkovito delovanje storitve. Šifriranje prometa med storitvijo in odjemalci ščiti prenesene informacije pred nepooblaščenim dostopom. Storitev omogoča povezovanje preko več nastavljivih TCP/IP vrat in naslovov obeh IPv4 in IPv6 skladov.

Content Conversion Service [podsistem za samodejno pretvorbo vsebin]

Pomen: Neobvezen

Opis: Podsistem omogoča samodejno pretvorbo vsebin entitet. Zasnovan je na principu vtičnikov, katere lahko povežemo v verige. Na podlagi teh se samodejno pretvarjajo vsebine.

Za več informacij glej poglavje 3.3.10.1 Samodejno pretvarjanje vsebin.

3.2 Atribut

V svetu velikih količin posamičnih, med seboj povezanih ali nepovezanih informacij,
jih je za uspešno hrambo in učinkovit dostop nujno logično opisati in jih povezati.
Za to uporabljamo t.i. metapodatke, ki abstraktno predstavljajo »informacije o informaciji,«
oziroma »podatke o podatku«, ki je predmet hrambe.
Da bi vzpostavili in dolgoročno ohranili strukturo in normalizacijo posameznih metapodatkov,
bjih je potrebno uokviriti in jim predpisati pravila.
To v strežniku IMiS®/ARChive Server dosežemo z uporabo koncepta Atribut,
katerih skupine pripišemo entiteti. Za več informacij glej poglavje 3.3 Entiteta.

Atribut je osnovna celica ali vsebnik (angl. Container) metapodatka.
Ta predpisuje pravila in okvirje za vnos, vzdrževanje in hrambo vrednosti metapodatkov,
ki pripadajo entiteti. Lahko ga označimo tudi za model metapodatka, kateremu se morajo
vrednosti prilagoditi, če jih želimo v atribut shraniti.
Poznamo več tipov atributov, ki primarno določajo tip vrednosti, katere je možno shraniti
v atribut. Atributi vsebujejo tudi kontrolne parametre, ki določajo razpone vrednosti, njihovo
formo, zmožnost iskanja po vrednostih, ...
Atribute po določitvi povežemo v t.i. metapodatkovne ali atributne sheme,
ki jih združujemo v predlogah entitet. Za več informacij glej poglavje 3.3.4 Predloge.

Povezave med atributi in predlogami določajo kontrolne parametre, specifične za povezavo med atributom in entiteto, kot so:

- obveznost
- zmožnost vsebovanja več vrednosti
- nespremenljivost
- forma in razpon vrednosti, ki je specifična za atribut v okviru nekega tipa entitete
- drugi parametri.

Atribut je ključen gradnik in steber učinkovitega elektronskega arhiva, saj hranjeno informacijo (vsebino) normalizirano opisuje in uporabnikom omogoča dostop do vsebine, ko telo vsebine ne omogoča neposrednega dostopa. Razlog za to je lahko sama narava vsebine (zvočni ali slikovni zapis) ali njen obseg.

3.2.1 Vrste

Identifikator: 1

Naziv: DirectoryEntity

Opis: Predstavlja identifikator entitete iz imenika. Podatek lahko smiselno uporabimo za pridobitev podatkov entitete iz imenika.

Razpon vrednosti: Registrirana entiteta imenika

Reference: /

Identifikator: 2

Naziv: Bool

Opis: Logična oziroma bitna vrednost, uporabna v logični in Boolean algebri.
Označuje resnično (pozitivno) in neresnično (negativno) stanje.

Razpon vrednosti: True, False ali 1, 0

Reference: http://en.wikipedia.org/wiki/Boolean_data_type

Identifikator: 3

Naziv: Int8

Opis: Predznačeno 8 bitno celo število.

Razpon vrednosti: Min: -128, Max: 127

Reference: http://en.wikipedia.org/wiki/Character_%28computing%29

Identifikator: 4

Naziv: UInt8

Opis: Ne-predznačeno 8 bitno celo število.

Razpon vrednosti: Min: 0, Max: 255

Reference: http://en.wikipedia.org/wiki/Byte

Identifikator: 5

Naziv: Int16

Opis: Predznačeno 16 bitno celo število.

Razpon vrednosti: Min: -32768, Max: 32767

Reference: http://en.wikipedia.org/wiki/Integer_%28computing%29#Short_integer

Identifikator: 6

Naziv: UInt16

Opis: Ne-predznačeno 16 bitno celo število.

Razpon vrednosti: Min: 0, Max: 65535

Reference: http://en.wikipedia.org/wiki/Integer_%28computing%29#Short_integer

Identifikator: 7

Naziv: Int32

Opis: Predznačeno 32 bitno celo število.

Razpon vrednosti: Min: -2147483648, Max: 2147483647

Reference: http://en.wikipedia.org/wiki/Integer_%28computing%29#Long_integer

Identifikator: 8

Naziv: UInt32

Opis: Ne-predznačeno 32 bitno celo število.

Razpon vrednosti: Min: 0, Max: 4294967295

Reference: http://en.wikipedia.org/wiki/Integer_%28computing%29#Long_integer

Identifikator: 9

Naziv: Int64

Opis: Predznačeno 64 bitno celo število.

Razpon vrednosti: Min: -9223372036854775808, Max: 9223372036854775807

Reference: http://en.wikipedia.org/wiki/Integer_%28computing%29#Long_integer

Identifikator: 10

Naziv: UInt64

Opis: Nepredzančeno 64 bitno celo število.

Razpon vrednosti: Min: 0, Max: 18446744073709551615

Reference: http://en.wikipedia.org/wiki/Integer_%28computing%29#Long_integer

Identifikator: 11

Naziv: Int128

Opis: Predznačeno 128 bitno celo število.

Razpon vrednosti: Min: -170141183460469231731687303715884105728,

Max: 170141183460469231731687303715884105727

Reference: http://en.wikipedia.org/wiki/Integer_%28computing%29#Long_integer

Identifikator: 12

Naziv: UInt128

Opis: Nepredzančeno 128 bitno celo število.

Razpon vrednosti: Min: 0, Max: 340282366920938463463374607431768211455

Reference: http://en.wikipedia.org/wiki/Integer_%28computing%29#Long_integer

Identifikator: 13

Naziv: Double

Opis: Realno (racionalno) število v plavajoči vejici z dvojno natančnostjo.

Ni primeren za vrednosti kjer je nujna absolutna natančnost (npr. bančne transakcije).

V takšnih primerih je primernejši tip Decimal.

Razpon vrednosti: / (dinamični razpon)

Reference: http://en.wikipedia.org/wiki/Double-precision_floating-point_format

Identifikator: 14

Naziv: Date

Opis: Datumska vrednost brez časovne komponente.

Razpon vrednosti: Min: 1.1.8192 pr. n. š., Max: 31.12.8191

Reference: http://en.wikipedia.org/wiki/ISO_8601

Identifikator: 15

Naziv: Time

Opis: Časovna vrednost brez datumske komponente.

Razpon vrednosti: Vseh 24 ur natančno na 1 milisekundo.

Reference: http://en.wikipedia.org/wiki/ISO_8601

Identifikator: 16

Naziv: DateTime

Opis: Datumska vrednost s časovno komponento.

Razpon vrednosti: Min: 1.1.8192 pr. n. š. s časovno komponento (glej Date + Time),

Max: 31.12.8191 s časovno komponento (glej Date + Time)

Reference: http://en.wikipedia.org/wiki/ISO_8601

Identifikator: 17

Naziv: StringMax

Opis: Neomejen niz UTF-8 znakov.

Razpon vrednosti: / (omejeno z zmožnostjo platforme)

Reference: http://en.wikipedia.org/wiki/String_%28computer_science%29

http://en.wikipedia.org/wiki/UTF-8

Identifikator: 18

Naziv: String10

Opis: Niz UTF-8 znakov, dolgih 10 bajtov.

Razpon vrednosti: 10 bajtov UTF-8 znakov

Reference: http://en.wikipedia.org/wiki/String_%28computer_science%29

http://en.wikipedia.org/wiki/UTF-8

Identifikator: 19

Naziv: String20

Opis: Niz UTF-8 znakov, dolgih 20 bajtov.

Razpon vrednosti: 20 bajtov UTF-8 znakov

Reference: http://en.wikipedia.org/wiki/String_%28computer_science%29

http://en.wikipedia.org/wiki/UTF-8

Identifikator: 20

Naziv: String30

Opis: Niz UTF-8 znakov, dolgih 30 bajtov.

Razpon vrednosti: 30 bajtov UTF-8 znakov

Reference: http://en.wikipedia.org/wiki/String_%28computer_science%29

http://en.wikipedia.org/wiki/UTF-8

Identifikator: 21

Naziv: String40

Opis: Niz UTF-8 znakov, dolgih 40 bajtov.

Razpon vrednosti: 40 bajtov UTF-8 znakov

Reference: http://en.wikipedia.org/wiki/String_%28computer_science%29

http://en.wikipedia.org/wiki/UTF-8

Identifikator: 22

Naziv: String50

Opis: Niz UTF-8 znakov, dolgih 50 bajtov.

Razpon vrednosti: 50 bajtov UTF-8 znakov

Reference: http://en.wikipedia.org/wiki/String_%28computer_science%29

http://en.wikipedia.org/wiki/UTF-8

Identifikator: 23

Naziv: String100

Opis: Niz UTF-8 znakov, dolgih 100 bajtov.

Razpon vrednosti: 100 bajtov UTF-8 znakov

Reference: http://en.wikipedia.org/wiki/String_%28computer_science%29

http://en.wikipedia.org/wiki/UTF-8

Identifikator: 24

Naziv: String200

Opis: Niz UTF-8 znakov, dolgih 200 bajtov.

Razpon vrednosti: 200 bajtov UTF-8 znakov

Reference: http://en.wikipedia.org/wiki/String_%28computer_science%29

http://en.wikipedia.org/wiki/UTF-8

Identifikator: 31

Naziv: Decimal1

Opis: Predznačeno realno (racionalno) število, natančno na 1 decimalno mesto.

Razpon vrednosti: Min: -92233720368547758078, Max: 92233720368547758087

Reference: http://en.wikipedia.org/wiki/Decimal

Identifikator: 32

Naziv: Decimal2

Opis: Predznačeno realno (racionalno) število, natančno na 2 decimalni mesti.

Razpon vrednosti: Min: -922337203685477580.78, Max: 922337203685477580.87

Reference: http://en.wikipedia.org/wiki/Decimal

Identifikator: 33

Naziv: Decimal3

Opis: Predznačeno realno (racionalno) število, natančno na 3 decimalna mesta.

Razpon vrednosti: Min: -92233720368547758.078, Max: 92233720368547758.087

Reference: http://en.wikipedia.org/wiki/Decimal

Identifikator: 34

Naziv: Decimal4

Opis: Predznačeno realno (racionalno) število, natančno na 4 decimalna mesta.

Razpon vrednosti: Min: -9223372036854775.8078, Max: 9223372036854775.8087

Reference: http://en.wikipedia.org/wiki/Decimal

Identifikator: 35

Naziv: Decimal5

Opis: Predznačeno realno (racionalno) število, natančno na 5 decimalnih mest.

Razpon vrednosti: Min: -922337203685477.58078, Max: 922337203685477.58087

Reference: http://en.wikipedia.org/wiki/Decimal

Identifikator: 36

Naziv: Decimal6

Opis: Predznačeno realno (racionalno) število, natančno na 6 decimalnih mest.

Razpon vrednosti: Min: -92233720368547.758078, Max: 92233720368547.758087

Reference: http://en.wikipedia.org/wiki/Decimal

Identifikator: 37

Naziv: Decimal7

Opis: Predznačeno realno (racionalno) število, natančno na 7 decimalnih mest.

Razpon vrednosti: Min: -9223372036854.7758078, Max: 9223372036854.7758087

Reference: http://en.wikipedia.org/wiki/Decimal

Identifikator: 38

Naziv: Decimal8

Opis: Predznačeno realno (racionalno) število, natančno na 8 decimalnih mest.

Razpon vrednosti: Min: -922337203685.47758078, Max: 922337203685.47758087

Reference: http://en.wikipedia.org/wiki/Decimal

Identifikator: 39

Naziv: Decimal9

Opis: Predznačeno realno (racionalno) število, natančno na 9 decimalnih mest.

Razpon vrednosti: Min: -92233720368.547758078, Max: 92233720368.547758087

Reference: http://en.wikipedia.org/wiki/Decimal

Identifikator: 40

Naziv: Decimal10

Opis: Predznačeno realno (racionalno) število, natančno na 10 decimalnih mest.

Razpon vrednosti: Min: -9223372036.8547758078, Max: 9223372036.8547758087

Reference: http://en.wikipedia.org/wiki/Decimal

Identifikator: 41

Naziv: Binary

Opis: Neomejen niz zlogov s podatkom o tipu vsebine (MIME). Zmožen je posredovanja velikosti hranjenega niza.

Razpon vrednosti: / (omejeno z zmožnostjo platforme)

Reference: http://en.wikipedia.org/wiki/Bitstring

http://en.wikipedia.org/wiki/Internet_media_type

Identifikator: 42

Naziv: File

Opis: Atribut je zmožen hrambe datotek s: podatkom o tipu vsebine (MIME),

opisom, časom nastanka, zadnje spremembe in zadnjega dostopa.

Zmožen je tudi posredovanja velikosti hranjenega niza.

Razpon vrednosti: Vsebina z velikostjo do 92233720368547758O7 bajtov,

opis z velikostjo do 4096 bajtov UTF-8 znakov.

Reference: http://en.wikipedia.org/wiki/Internet_media_type

3.2.2 Parametri atributov

Atribut poleg njegovega tipa opisujejo še naslednji podatki:

- »Ime« (angl. Name): naziv, obvezen unikaten niz UTF-8 znakov, dolg do 256 bajtov UTF-8 znakov.
- »Opis« (angl. Description): opis atributa, opcijski niz UTF-8 znakov, dolg do 512 bajtov UTF-8 znakov.
- »Preverjanje ustrezne vrednosti« (angl. Validation expression): niz UTF-8 znakov, ki predstavljajo regularni izraz (angl. Regular expression) s katerim se nove ali spremenjene vrednosti atributa preverjajo.
  Več o sintaksi in pravilih: http://en.wikipedia.org/wiki/Regular_expression.
- »Parametri« (angl. Parameters): dodatni parametri atributa, ki poleg njegovega podatkovnega tipa dodatno uokvirjajo pripisane vrednosti: »Searchable«, »Unique« in »PickList«.

»Searchable«

Vrednosti atributa postanejo del indeksa, po katerem je možno s funkcijami iskanja iskati entitete, katerim vrednosti so pripisane. Nasprotno pa, po vrednostih atributov, ki niso označeni kot »Searchable« ni mogoče iskati s funkcijami iskanja.
Na podlagi tega dejstva se je zato potrebno ob nastavitvi strežnika IMiS®/ARChive Server in atributov odločati, ali atributu ta parameter pripisati.
Ko atribut dobi prvo vrednost, tega parametra ni mogoče spreminjati.
Ta parameter lahko bistveno vpliva na delovanje in zmožnosti (angl. Performance) strežnika. Podatki se namreč shranjujejo v iskalna drevesa, ki lahko postanejo velika in počasna.
Na drugi strani se vrednosti »Non-searchable« atributov le shranijo, zato njihova količina bistveno ne vpliva na delovanje strežnika.

»Unique«

Vsaka vrednost atributa je unikatna skozi celoten arhiv. Ta parameter vključimo, ko želimo zagotoviti, da ne pride do vnosa vrednosti atributa, ki ga že določa druga entiteta. Kršitev tega pravila onemogoča shranitev entitete. S tem parametrom je možno upravljati tudi, ko so atributu že pripisane vrednosti. Možno ga je le izključiti. Dokler atribut nima pripisanih vrednosti je s parametrom možno prosto upravljati.

»PickList«

Vrednosti atributa so vnaprej določene zato ročni vnos izven seznama dovoljenih vrednosti ni možen. Seznam dovoljenih vrednosti atributa določimo ob nastavitvi strežnika IMiS®/ARChive Server.

Ko so seznamu enkrat pripisane vrednosti, jih lahko le dodajamo. Dokler atribut nima vrednosti lahko vrednosti tudi odvzemamo. Dodatno lahko vsaki vrednosti določimo:

- Atribute:

  - »alias«: sinonim, uporabniku prijazen naziv vrednosti, ki ga odjemalci lahko uporabijo za izbire vrednosti.

  - »alias.xx_YY«: preveden sinonim, uporabniku prijazen naziv vrednosti, preveden v področje, ki ga identificira identifikator xx_YY. Odjemalci lahko uporabijo to vrednost za izbiro vrednosti, ko odjemalec deluje v področnih nastavitvah xx_YY.
  Področni identifikatorji xx_YY so strukturirani po ISO standardu 639 (http://en.wikipedia.org/wiki/ISO_639).

  - »default«: privzeta vrednost atributa v primeru novih entitet. To lastnost atributa lahko odjemalci uporabijo v uporabniških vmesnikih za prikaz privzete vrednosti atributa.
- Pogoje: Navedemo pogoje, pod katerimi je neka vrednost dovoljena. Uporabimo lahko sintakso parov pogojev ali samo vrednost, če želimo, da se nanaša na atribut, kateremu so pripisani pogoji.

Primera:

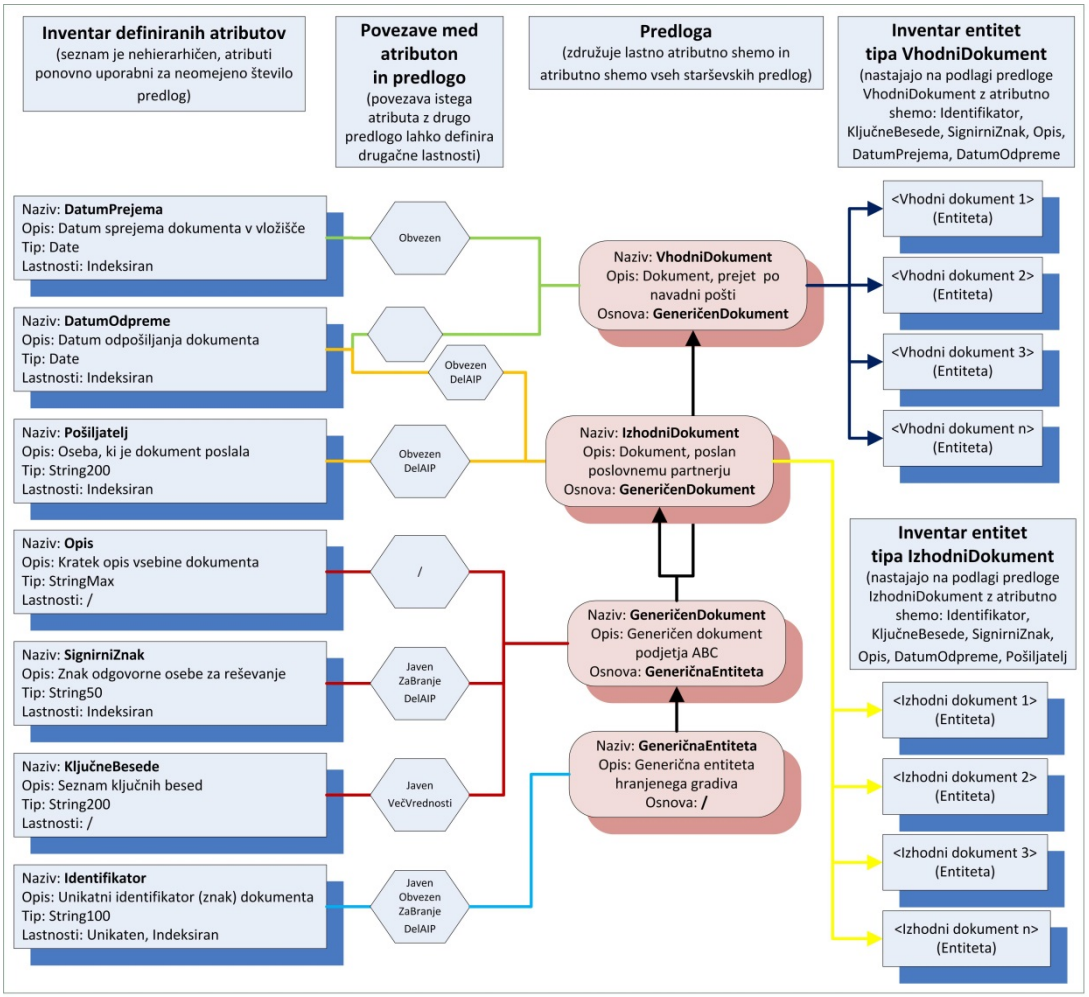
- *"1:{default,alias="Opened",alias.sl_SI="Odprto"} – Vrednost atributa »1« identificirata njegova sinonima "Opened" in preveden v slovenščino "Odprto" in je privzet za nove entitete.*
  Pogojev, kdaj je na voljo, nima določenih.
- *"2:{alias="Closed",alias.sl_SI="Zaprto"}:{"":"1"] – Vrednost atributa »2« identificirata njegova sinonima "Closed" in preveden v slovenščino "Zaprto".*
  Na voljo je pod pogojem, da atribut nima vrednosti (nova entiteta) ali ko je njegova vrednost 1.

3.2.3 Povezava s predlogami

Atribut samostojno ni sposoben hrambe metapodatkov, je le osnova, oziroma izvor kontejnerja (angl. Container) podatkov. Kontejner vrednosti metapodatka dokončno določa vzpostavljena povezava med atributom in predlogo. Poenostavljeno gledano so predloge nastavitve, na podlagi katerih nastajajo v arhivu entitete. Za več informacij glej poglavje 3.3.4 Predloge.

V povezavi z atributom se je potrebno zavedati tudi koncepta dedovanja predlog.

Dedovanje bistveno vpliva na razumevanje metapodatkovnih shem izvedenih predlog. Osnovni element metapodatkovne sheme entitete je povezava med predlogo in atributom. Ta določa t.i. »pogodbo« oziroma vse parametre in omejitve, ki se jih morajo vnesene metapodatkovne vrednosti za nek atribut držati, če se jih želi vnesti v atribut, ki jih vsebuje.



Slika 2: Logične povezave med atributi, predlogami in entitetami

Model omogoča ponovno uporabo istega atributa v različnih predlogah, pri čemer je možno nekatere parametre atributa določati na nivoju povezave.

Ti parametri so:

- »Validation expression«: Niz UTF-8 znakov, ki predstavljajo regularni izraz (angl. Regular expression) s katerim se nove ali spremenjene vrednosti atributa preverjajo. Več o sintaksi in pravilih na naslovu: http://en.wikipedia.org/wiki/Regular_expression. Ta izraz nadomesti izraz, nastavljen na nivoju atributa. Tako se preverjanje glede na izraz atributa nadomesti z izrazom na nivoju povezave med atributom in predlogo.

- »Sequence«: Sekvenca atributa v metapodatkovni shemi predloge.
  Gre za vrstni red atributa v shemi pri čemer je potrebno upoštevati naslednja pravila:
  1. shema vsake starševske predloge ima svoj sekvenčni red, ki ni odvisen od starševske predloge in/ali morebitnih predlog, ki iz nje nastanejo;
  2. atributi sheme korenske starševske predloge so na začetku konsolidirane atributne sheme;
  3. atributi sheme izvedene predloge, iz katere nastajajo entitete, so na koncu konsolidirane atributne sheme te predloge.
- »PickList values«: V kolikor je to potrebno, je možno nadomestiti nabor vnaprej določenih vrednosti atributa, ki so mu določene globalno. Seznam je shranjen v kontekstu povezave med atributom in predlogo. Ročni vnos vrednosti izven seznama dovoljenih vrednosti ni dovoljen. Seznam dovoljenih vrednosti atributa določamo ob nastavitvi produkta pri čemer lahko seznamu vrednosti le dodajamo, ko so mu enkrat pripisane vrednosti. Dokler vrednosti nima, je možno tudi odvzemanje vrednosti.
  Za več informacij glej poglavje 3.2.2 Parametri atributov, odstavek »PickList«.
- »Parameters«: dodatni parametri atributa, ki poleg njegovega podatkovnega tipa dodatno uokvirjajo pripisane vrednosti opisne v nadaljevanju:

Public

Javni atribut. Vrednosti atributa v okviru neke entitete so javnega značaja in dostopne tudi entitetam imenika (uporabniki, skupine), ki sicer nimajo pravic dostopa do entitete.
S tem parametrom je možno upravljati v celotnem življenjskem ciklu atributa in/ali predloge, s katero je atribut povezan. Upošteva se pri prvem odpiranju entitete s strani uporabnika po spremembi nastavitve.

Multivalue

Atribut, ki lahko vsebuje več vrednosti. Atributu je v okviru ene entitete možno pripisati več vrednosti. V kolikor ta parameter ni določen, je možno atributu v okviru ene entitete pripisati natančno eno ali nobene vrednosti. S tem parametrom je možno upravljati v celotnem življenjskem ciklu atributa in/ali predloge s katero je atribut povezan, dokler ni ustvarjena prva entiteta na podlagi predloge.
Po tem je možno parameter kadarkoli vključiti in nikoli izključiti. Upošteva se pri prvem odpiranju entitete s strani uporabnika po spremembi nastavitve.

Required

Vrednost atributa mora biti ob shranjevanju entitete nujno prisotna. Shranjevanje entitete bo torej zavrnjeno, če shranjevalec entitete v okviru atributa ne določi vsaj ene veljavne vrednosti.
S tem parametrom je možno upravljati v celotnem življenjskem ciklu atributa in/ali predloge, s katero je atribut povezan dokler ni ustvarjena prva entiteta na podlagi predloge.
Po tem je možno parameter kadarkoli izključiti in nikoli vključiti. Upošteva se pri prvem odpiranju entitete s strani uporabnika po spremembi nastavitve.

ReadOnly

Vrednosti atributa se po prvi shranitvi ne smejo spreminjati.
Vrednosti atributa je možno določiti ob prvi shranitvi entitete. Vsi poizkusi kasnejših sprememb atributa bodo zavrnjeni. S tem parametrom je možno upravljati v celotnem življenjskem ciklu atributa in/ali predloge, s katero je atribut povezan.
Upošteva se pri prvem odpiranju entitete s strani uporabnika po spremembi nastavitve.

Inherited

Vrednosti atributa dedujejo vrednosti iz nadrejene hierarhije. V kolikor v entiteti niso določene eksplicitne vrednosti, se te dedujejo iz prve nadrejene entitete, ki ima določene eksplicitne vrednosti. Eksplicitne vrednosti v celoti nadredijo podedovane vrednosti in učinkujejo za vse podrejene entitete.
Dedovanje vrednosti deluje po principu referenc. Če se vrednost podedovanega atributa spremeni v nadrejeni hierarhiji neke entitete, začne nemudoma učinkovati za vse spremenjeni entiteti podrejene entitete, ki nimajo pripisanih eksplicitnih vrednosti.
Vrednosti atributa se dedujejo le v primeru, ko ima entiteta in njej neposredno nadrejena entiteta atribut v svoji shemi atributov.

S tem parametrom je možno upravljati v celotnem življenjskem ciklu atributa in/ali predloge, s katero je atribut povezan. Upošteva se pri prvem odpiranju entitete s strani uporabnika po spremembi nastavitve.

AppendOnly

Vrednosti atributa je možno k obstoječim le dodajati. Odjemalec s strani strežnika dobi vse vrednosti, pri zapisu pa sprejema le nove vrednosti, ki jih doda obstoječim.
S tem parametrom je možno upravljati v celotnem življenjskem ciklu atributa in/ali predloge, s katero je atribut povezan. Upošteva se pri prvem odpiranju entitete s strani uporabnika po spremembi nastavitve.

IncludeInAIP

Vrednosti atributa so del arhivskega informacijskega paketa. V postopku zagotavljanja avtentičnosti hranjenega gradiva, postanejo vrednosti atributov, katerim je ta parameter vključen, del arhivskega informacijskega paketa. Za več informacij glej poglavje 3.6.5 AIP).
S tem parametrom je možno upravljati v celotnem življenjskem ciklu atributa in/ali predloge, s katero je atribut povezan. Nastavitev strežnik upošteva, ko začne postopek zagotavljanja avtentičnosti in nespremenjenosti gradiva, v času njegove hrambe.

IsNonEmpty

Vrednost atributa ne sme biti prazna. Definicija prazne vrednosti je odvisna od vrste atributa.Tako je prazna vrednost za vrsto »String10« prazen niz, prazna vrednost za vrsto »File« pa predstavlja prazno datoteko (datoteko velikosti 0 bajtov). S tem parametrom je možno upravljati v celotnem življenjskem ciklu atributa in/ali predloge. To velja za atribut, ki je povezan dokler ni ustvarjena prva entiteta na podlagi predloge. Po tem je možno parameter kadarkoli izključiti in nikoli vključiti. Upošteva se pri prvem odpiranju entitete s strani uporabnika po spremembi nastavitve.

Omejitve:

Pri konfiguraciji atributov na predlogah za ketere že obstajajo entitete, veljajo naslednje omejitve:

- *Pri dodajanju novega atributa, lastnost »Required« ne sme biti nastavljena na »True«.*
  V kolikor je nastavljena na »False«, to povzroči nekonsistenco z vsemi obstoječimi entitetami, saj le-te ne smejo imeti vrednosti za nov atribut.

- *Pri spreminjanju obstoječega atributa na predlogi veljajo naslednje omejitve:*
  - *sprememba lastnosti atributa »Multivalue« iz »True« v »False« ni dovoljena zaradi nekonsistence z vsemi obstoječimi entitetami, ki imajo nastavljenih več vrednosti za ta atribut;*
  - *sprememba lastnosti atributa »Required« iz »False« v »True« ni dovoljena zaradi nekonsistence z vsemi obstoječimi entitetami, ki nimajo vrednosti za ta atribut;*
  - *sprememba lastnosti atributa »Inherited« iz »True« v »False« ni dovoljena zaradi nekonsistence z vsemi obstoječimi entitetami, ki nimajo eksplicitne vrednosti za ta atribut.*
- *Brisanje atributa iz predloge, za katerega obstajajo entitete ni dovoljeno.*

Vse ostale kombinacije so dovoljene.

3.2.4 Poimenovanje

Vsak atribut je potrebno poimenovati s poljubnim unikatnim nizom UTF-8 znakov, dolgim do 256 bajtov UTF-8 znakov. Dovoljeni so vsi znaki, upoštevati pa je potrebno naslednje omejitve:

- Naziv atributa ne sme biti prazen.
- Predpone (imenski prostori)»int:«, »sys:«, »imp:«, »trf:« so rezervirane in uporaba ni dovoljena. Poizkus ustvarjanja tako imenovanega atributa bo zavrnjen.
- Imena atributov naj bodo smiselna in primerno kratka.
- Dovoljene so predpone oziroma imenski prostori, ki jih od imena delimo z znakom »:«. Dovoljene so gnezdene predpone (npr.»global:accounting:InvoiceNo«).
- Sistem ne predpisuje pravil poimenovanja atributov vendar je smiselno, da ga določite na nivoju internega pravilnika upravljanja s strežnikom IMiS®/ARChive Server.

Dodatno je smiselno atributu določiti opis, opcijski niz UTF-8 znakov, dolg do 512 bajtov UTF-8 znakov, ki naj bo logičen opis, kaj atribut predstavlja in kakšne vrednosti so dovoljene, če obstajajo omejitve. Podatek se prenaša na odjemalce, ki ga lahko prikazujejo v uporabniškem vmesniku. Lokalizacija na nivoju strežnika ni podprta.
V ta namen lahko uporabimo nize znakov kot konstante, ki jih glede na regionalne nastavitve na odjemalcih primerno prevajate.

3.2.5 Sistemski atributi

Na delovanje arhivskega sistema močno vplivajo tudi nekateri atributi in njihove vrednosti.
Te atribute imenujemo »Sistemski atributi«, ki so v nasprotju z nastavljivimi atributi v sistemu vnaprej določeni, njihove lastnosti pa nespremenljive.

Razlog za to je, da njihova prisotnost/odsotnost, vrednosti in lastnosti vplivajo na delovanje strežnika, življenjski cikel entitet in postopke hrambe gradiva.
Sistemski atributi so vgrajeni v strežnik IMiS®/ARChive Server in z njimi ni potrebno upravljati. Povezani so na sistemske predloge, iz katerih izvedemo predloge,
s katerimi ustvarjamo entitete. Ustvarjene entitete jih zato implicitno pridobijo v atributne sheme. Tako lahko vplivajo na njihov življenjski cikel in prisotnost sistemskih metapodatkov. Vrednosti nekaterih sistemskih atributov določa strežnik v postopkih ustvarjanja, spreminjanja in shranjevanja entitet. Nekaterim pa je uporabnik dolžan vnesti vrednost(i), saj le tako lahko uspešno upravlja z življenjskim ciklom entitete.

V nadaljevanju so podrobneje opisani sistemski atributi:

Atribut »sys:ExternalIds«

Tip: String100

Lastnosti: Unique, Searchable, Public, MultiValue, IsNonEmpty

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti. Ena entiteta lahko vsebuje do 65536 zunanjih identifikatorjev.

Uporaba: Razred, Zadeva, Dokument, Dokument v postopku odbiranja in izločanja

Opis: Unikatni zunanji identifikatorji entitete. Uporabni kot podatki za dostop do entitete (odpiranje). Klicatelj je odgovoren za unikatnost identifikatorja, shranitev ni dovoljena, v kolikor ima druga entiteta že pripisan enak zunanji identifikator.

Atribut »sys:Title«

Tip: String200

Lastnosti: Searchable, Public, Required, IsNonEmpty, [ReadOnly pri kopijah politik hrambe in izbrisanih entitetah]

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Razred, Zadeva, Dokument, Dokument v postopku odbiranja in izločanja, Politike hrambe, Postopek odbiranja in izločanja, Kopija politike hrambe, Zadržanje postopka odbiranja in izločanja, Kontejner sistemskih entitet.

Opis: Obvezen naziv (naslov) entitete, ki jo opisuje. Spremenljiv v celotnem življenjskem ciklu entitete. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami,
služi le kot nosilec informacije.

Atribut »sys:Description«

Tip: String200

Lastnosti: Public, [ReadOnly pri izbrisanih entitetah in kopijah politik hrambe]

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Razred, Zadeva, Dokument, Dokument v postopku odbiranja in izločanja, Politike hrambe, Postopek odbiranja in izločanja, Kopija politike hrambe, Zadržanje postopka odbiranja in izločanja, Kontejner sistemskih entitet.

Opis: Neobvezen kratek opis entitete. Spremenljiv v celotnem življenjskem ciklu entitete.
Če pride do izbrisa entitete, postane obvezen. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:Content«

Tip: File

Lastnosti: Searchable, IncludeInAIP, MultiValue, [Public v Pregledu postopkov odbiranja in izločanja, Public in ReadOnly pri Kopijah politike hrambe].

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti. Ena entiteta lahko vsebuje do 65536 vrednosti.

Uporaba: Dokument, Dokument v postopku odbiranja in izločanja, Politike hrambe, Postopek odbiranja in izločanja, Kopija politike hrambe.

Opis: Atribut predstavlja kontejner digitalnih vsebin, zmožen hranjenja opisovalnih struktur vsebin. Atribut je vključen v skupino atributov, ki se indeksirajo. Specifično za File atribut je, da to predstavlja indeks po polnem besedilu (Full Text Index).
V kolikor je vključen sistem za zagotavljanje avtentičnosti in nespremenljivosti v času hrambe gradiva, postanejo prstni odtisi vrednosti iz tega kontejnerja del arhivskega informacijskega paketa (AIP). Spremenljiv je v celotnem življenjskem ciklu entitete. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:Keywords«

Tip: String30

Lastnosti: Searchable, Public, MultiValue

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti. Ena entiteta lahko vsebuje do 65536 vrednosti.

Uporaba: Razred, Zadeva, Dokument, Dokument v postopku odbiranja in izločanja, Postopek odbiranja in izločanja.

Opis: Neobvezne ključne besede, ki določajo entiteto. Spremenljiv je v celotnem življenjskem ciklu entitete. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:SecurityClass«

Tip: UInt32

Lastnosti: Searchable, Public, ReadOnly, Inherited, PickList, IsNonEmpty

Omejitve: Veljavne vrednosti (prednastavljene):

- 1, alias »Unclassified«: Entiteta nima posebej določene stopnje tajnosti.
- 2, alias »Restricted«: Entiteta je interne narave. Do nje lahko dostopajo le uporabniki s stopnjo tajnosti »Restricted« ali višjo.
- 3, alias »Confidential«: Entiteta je zaupne narave. Do nje lahko dostopajo le uporabniki s stopnjo tajnosti »Confidential« ali višjo.
- 4, alias »Secret«: Entiteta je tajne narave. Do nje lahko dostopajo le uporabniki s stopnjo tajnosti »Secret« ali višjo.
- 5, alias »Top Secret«: Entiteta je strogo tajne narave. Do nje lahko dostopajo le uporabniki s stopnjo tajnosti »Top Secret« ali višjo.

Stopnje tajnosti so sicer nastavljive glede na pravilnike arhiviranja uporabnika arhiva.
Neveljavna vrednost je 0, ki je rezervirana in interne narave, zato se med veljavne vrednosti ne sme vnašati.

Uporaba: Razred, Zadeva, Dokument

Opis: Sistemski atribut omejuje dostop do entitete, ki ga določa. Entitete, do katere uporabnik z svojo stopnjo tajnosti nima dostopa, se uporabniku v celoti skrije. Uporabnik ne more potrditi njenega obstoja ali izvajati katerekoli aktivnosti, ki bi njen obstoj potrdila. To poleg branja njenih atributov, odpiranja, brisanja, premikanja ipd., velja tudi za ustvarjanje entitet pod njo.
Vrednost stopnje tajnosti se deduje po podrejenih entitetah, če jih same ne določajo.
Podrejenim entitetam je možno določiti lastno stopnjo tajnosti, vendar le do nivoja nadrejene, torej lahko je enaka ali nižja od stopnje tajnosti nadrejene entitete.

Atribut »sys:Status«

Tip: UInt32

Lastnosti: Searchable, Public, Required, PickList, Inherited, ReadOnly, IsNonEmpty, [Inherited lastnost v primeru Postopkov odbiranja in izločanja ne velja].

Omejitve: Veljavne vrednosti:

- 1, alias »Opened«: entiteto je dovoljeno spreminjati in pod njo ustvariti podrejene entitete.
- 2, alias »Closed«: entiteta ne dovoljuje spreminjana in ustvarjanja podrejenih entitet.

Prisotnost atributa je obvezna na razredu, dokumentu pod razredom ter zadevah.
Uporaba: Razred, Zadeva, Dokument, Postopek odbiranja in izločanja

Opis: Atribut predstavlja stanje entitete,oziroma njen status. Ta se odraža v postopkih, ki so dovoljeni ali prepovedani na entiteti. V odprtem stanju je entiteto možno spreminjati in z njo prosto upravljati. Nivo dostopa določajo dostopne pravice.
Ko je statusno entiteta enkrat zaprta, so ne glede na dostopne pravice spreminjanja onemogočene. V kolikor je omogočen servis za zagotavljanje avtentičnosti in nespremenjenosti v času hrambe, je zapiranje entitete signal za začetek postopka zagotavljanja avtentičnosti za konkretno entiteto in morebitne podrejene entitete (glej poglavje 3.3.7 Življenjski cikel).

Atribut »sys:Creator«

Tip: DirectoryEntity
Lastnosti: Searchable, Public, Required, ReadOnly, IsNonEmpty, [Inherited v primeru Kontejnerja sistemskih entitet].
Omejitve: Vrednost določa strežnik in ni spremenljiva. Vrednost vedno predstavlja registrirano entiteto iz imenika.
Uporaba: Razred, Zadeva, Dokument, Dokument v postopku odbiranja in izločanja, Politike hrambe, Postopek odbiranja in izločanja, Kopija politike hrambe, Zadržanje postopka odbiranja in izločanja, Kontejner sistemskih entitet.
Opis: Vrednost predstavlja entiteto imenika - uporabnika, ki je entiteto ustvaril. Podatek je nespremenljiv, določen pri ustvaritvi entitete s strani strežnika. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:Owner«

Tip: DirectoryEntity
Lastnosti: Searchable, Public, IsNonEmpty
Omejitve: Dovoljene vrednosti so identifikatorji registriranih entitet iz imenika.
Uporaba: Razred, Zadeva, Dokument, Dokument v postopku odbiranja in izločanja, Postopek odbiranja in izločanja.

Opis: Vrednost predstavlja entiteto imenika – uporabnika ali skupino, ki je odgovorna za entiteto (lastnik). Podatek je spremenljiv v celotnem življenjskem ciklu entitete.
Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami,
služi le kot nosilec informacije.

Atribut »sys:Significance«

Tip: UInt32
Lastnosti: Searchable, Public, Inherited, PickList, IsNonEmpty
Omejitve: Veljavne vrednosti:

- 1, alias »Vital«: Entiteta, vitalnega pomena za lastnika arhiva. Entiteto je prepovedano izbrisati preko administratorskega zahtevka ali v postopku odbiranja in izločanja. Entiteta je opcijsko pod posebnim režimom varnostnega arhiviranja.
- 2, alias »Permanent«: Entitete ni dovoljeno izbrisati ali preko administratorskega zahtevka ali v postopku odbiranja in izločanja. Gre le za opozorilo, ki ga administrator lahko upošteva ali se odloči drugače.
- 3, alias »Retain«: Predstavlja opozorilo odgovorni osebi, ki izvaja odbiranje in izločanje. V postopku izločanja, da naj postopek izločitve na dotični entiteti zadrži.
- 4, alias »Delete«: Priporočilo administratorju, naj entiteto izbriše mimo postopka odbiranja in izločanja. Priporočilo lahko izda vsakokratni urejevalec entitete. Navadno se na tak način izbrišejo entitete, ki so bile napačno vnesene ali pa gre za zahtevan izbris na zahtevo lastnika dokumenta (v primeru osebnih podatkov, ...).

Uporaba: Razred, Zadeva, Dokument
Opis: Atribut predstavlja pomembnost entitete v kontekstu lastnika arhiva.

Atribut »sys:Opened«

Tip: DateTime
Lastnosti: Searchable, Public, ReadOnly, IsNonEmpty
Omejitve: Vrednost določa strežnik in ni spremenljiva. Vedno je prisoten ob sistemskem atributu »sys:Status«.
Uporaba: Razred, Zadeva, Dokument, Postopek odbiranja in izločanja.
Opis: Vrednost predstavlja datum in čas ko je atribut »sys:Status« dotične entitete dobil vrednost »Opened« (glej poglavje 3.2.5 Sistemski atribut »sys:Status«).
Podatek je nespremenljiv, določen s strani strežnika. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:Closed«

Tip: DateTime

Lastnosti: Searchable, Public, ReadOnly, Inherited, IsNonEmpty, [Inherited lastnost v primeru Postopkov odbiranja in izločanja ne velja].

Omejitve: Vrednost določa strežnik in ni spremenljiva. Vedno je prisoten ob sistemskem atributu »sys:Status«.

Uporaba: Razred, Zadeva, Dokument, Postopek odbiranja in izločanja.

Opis: Vrednost predstavlja datum in čas ko je atribut »sys:Status« dotične entitete dobi vrednost »Closed« (glej poglavje 3.2.5 Sistemski atribut »sys:Status«).

Podatek je nespremenljiv, določen s strani strežnika. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:CommitLog«

Tip: StringMax

Lastnosti: ReadOnly, MultiValue, AppendOnly, IncludeInAIP, IsNonEmpty

Omejitve: Vrednost določa strežnik in ni spremenljiva.

Uporaba: Razred, Zadeva, Dokument, Dokument v postopku odbiranja in izločanja, Politike hrambe, Postopek odbiranja in izločanja, Kopija politike hrambe.

Opis: Vrednost predstavlja vsakokratni dnevnik dogodkov samodejnih dejanj preverjanja strežnika ob shranitvi entitete. Poleg samega tekstovnega dnevnika, vsebuje še zajete elektronske podpise, vsebovane v arhiviranih vsebinah, elektronska potrdila celotne verige potrdil, ki so izdale potrdilo, s katerim je bil elektronski podpis ustvarjen ter trenutne sezname preklicanih potrdil vseh izdajateljev omenjenih potrdil. Podatki so nespremenljivi, določeni s strani strežnika. Podatki tega atributa se vključijo v arhivski informacijski paket in so predmet dolgoročne hrambe gradiva v postopku zagotavljanja avtentičnosti. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:move:Reason«

Tip: String200

Lastnosti: Searchable, MultiValue, ReadOnly, AppendOnly, IsNonEmpty

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Razred, Zadeva, Dokument

Opis: Vrednosti predstavljajo razloge za vsakokratni premik (re-klasifikacijo) entitete.
Gre za vrednost, ki je posredovana s strani uporabnika ob premiku entitete v hierarhiji razvrščanja gradiva. Istoležne vrednosti lahko povezujemo z atributoma »sys:move:Agent«, »sys:move:DateTime« in »sys:move:ClassificationCode« za popolnejšo informacijo o premiku entitete v hierarhiji razvrščanja gradiva.
Vrednost se vnese v metapodatke entitete, ki se premika, ne pa tudi v vse njej podrejene entitete. Za več informacij glej poglavje 3.3.7 Življenjski cikel.
Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami,
služi le kot nosilec informacije.

Atribut »sys:move:Agent«

Tip: DirectoryEntity
Lastnosti: Searchable, MultiValue, ReadOnly, AppendOnly, IsNonEmpty
Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.
Vrednosti vedno predstavljajo identifikatorje registriranih entitet iz imenika.
Uporaba: Razred, Zadeva, Dokument
Opis: Vrednosti predstavljajo vsakokratne entitete imenika (uporabnike),
ki so izvajali premike (reklasifikacijo) entitete. Gre za vrednost, ki jo strežnik samodejno določi iz uporabniške seje, oziroma iz prijavnih podatkov uporabnika,
ki je vsakokratni premik izvedel. Isto ležne vrednosti lahko uporabnik povezuje z atributoma »sys:move:Reason«, »sys:move:DateTime«in »sys:move:ClassificationCode« za popolnejšo informacijo o premiku entitete v hierarhiji razvrščanja gradiva.
Vrednost se vnese v metapodatke entitete, ki se premika, ne pa tudi v vse njej podrejene entitete. Za več informacij glej poglavje 3.3.7 Življenjski cikel. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:move:DateTime«

Tip: DateTime
Lastnosti: Searchable, MultiValue, ReadOnly, AppendOnly, IsNonEmpty
Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.
Uporaba: Razred, Zadeva, Dokument

Opis: Vrednosti predstavljajo vsakokratni datum in čas, ko se je izvedel premik (reklasifikacija) entitete. Gre za vrednost, ki jo strežnik samodejno določi.
Isto ležne vrednosti lahko uporabnik povezuje z atributoma »sys:move:Reason«, »sys:move:Agent« in »sys:move:ClassificationCode« za popolnejšo informacijo o premiku entitete v hierarhiji razvrščanja gradiva. Vrednost se vnese v metapodatke entitete, ki se premika, ne pa tudi v vse njej podrejene entitete.
Za več informacij glej poglavje 3.3.7 Življenjski cikel. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:move:ClassificationCode«

Tip: String200
Lastnosti: Searchable, MultiValue, ReadOnly, AppendOnly, IsNonEmpty
Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.
Uporaba: Razred, Zadeva, Dokument
Opis: Vrednosti predstavljajo vsakokratno klasifikacijsko oznako entitete ob premiku (reklasifikaciji). Gre za vrednost, ki jo strežnik samodejno določi.
Isto ležne vrednosti lahko uporabnik povezuje z atributoma »sys:move:Reason«, »sys:move:Agent« in »sys:move:DateTime« za popolnejšo informacijo o premiku entitete v hierarhiji razvrščanja gradiva. Vrednost se vnese v metapodatke entitete, ki se premika, ne pa tudi v vse njej podrejene entitete.
Za več informacij glej poglavje 3.3.7 Življenjski cikel. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:del:Reason«

Tip: String200
Lastnosti: Searchable, Public, Required, ReadOnly, IsNonEmpty
Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.
Uporaba: Razred, Zadeva, Dokument
Opis: Vrednost predstavlja razlog za izločanje ali izbris entitete. Gre za vrednost, ki je posredovana s strani uporabnika ob izbrisu entitete, v primeru izločanje pa se prenese iz postopka odbiranja in izločanja. Za več informacij glej poglavje 3.3.7 Življenjski cikel.
Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:del:Agent«

Tip: DirectoryEntity

Lastnosti: Searchable, Public, Required, ReadOnly, IsNonEmpty

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Vrednosti vedno predstavljajo identifikatorje registriranih entitet iz imenika.

Uporaba: Razred, Zadeva, Dokument

Opis: Vrednost predstavlja entiteto imenika (uporabnike), ki je izvedel izločanje ali izbris entitete. Gre za vrednost, ki jo strežnik samodejno določi iz uporabniške seje, oziroma prijavnih podatkov uporabnika, ki je akcijo izvedel. Za več informacij glej poglavje 3.3.7 Življenjski cikel. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:del:DateTime«

Tip: DateTime

Lastnosti: Searchable, Public, Required, ReadOnly, IsNonEmpty

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Razred, Zadeva, Dokument

Opis: Vrednost predstavlja datum in čas, ko je izvedeno izločanje ali izbris entitete. Gre za vrednost, ki jo strežnik samodejno določi.

Za več informacij glej poglavje 3.3.7 Življenjski cikel. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:del:ClassificationCode«

Tip: String200

Lastnosti: Searchable, Public, Required, ReadOnly, IsNonEmpty

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Razred, Zadeva, Dokument

Opis: Vrednost predstavlja klasifikacijsko oznako entitete v času izbrisa ali izločanja.

Gre za vrednost, ki jo strežnik samodejno zajame.

Za več informacij glej poglavje 3.3.7 Življenjski cikel. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:del:Reference«

Tip: String200

Lastnosti: Searchable, Public, ReadOnly

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Razred, Zadeva, Dokument

Opis: Vrednost predstavlja referenco na prenešeno entiteto. Gre za vrednost, ki jo uporabnik vnese po uspešnem prenosu entitete v postopku odbiranja in izločanja. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije

Atribut »sys:scc: Reason«

Tip: String200

Lastnosti: Searchable, MultiValue, ReadOnly, AppendOnly, IsNonEmpty

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Razred, Zadeva, Dokument

Opis: Vrednosti predstavljajo razloge za vsakokratno spremembo stopnje tajnosti entitete. Gre za vrednost, ki je posredovana s strani uporabnika ob spremembi stopnje tajnosti. Istoležne vrednosti lahko povezujemo z atributoma »sys:scc:Agent«, »sys:sccDateTime«, »sys:scc:From« in »sys:scc:To« za popolnejšo informacijo o spremembi stopnje tajnosti entitete. Vrednost se vnese v metapodatke entitete, ki se ji spremeni stopnja tajnosti, ne pa tudi v vse njej podrejene entitete. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:scc:Agent«

Tip: DirectoryEntity

Lastnosti: Searchable, MultiValue, ReadOnly, AppendOnly, IsNonEmpty

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Vrednosti vedno predstavljajo identifikatorje registriranih entitet iz imenika.

Uporaba: Razred, Zadeva, Dokument

Opis: Vrednosti predstavljajo vsakokratne entitete imenika (uporabnike), ki so izvajali spremembe stopenj tajnosti entitete. Gre za vrednost, ki jo strežnik samodejno določi iz uporabniške seje, oziroma iz prijavnih podatkov uporabnika, ki je stopnjo tajnosti vsakokrat izvedel. Isto ležne vrednosti lahko uporabnik povezuje z atributoma »sys:scc:Reason«, »sys:scc:DateTime«, »sys:scc:From« in »sys:scc:To« za popolnejšo informacijo o spremembi stopnje tajnosti entitete. Vrednost se vnese v metapodatke entitete, ki se ji spremeni stopnja tajnosti, ne pa tudi v vse njej podrejene entitete. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:scc: DateTime«

Tip: DateTime

Lastnosti: Searchable, MultiValue, ReadOnly, AppendOnly, IsNonEmpty

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Razred, Zadeva, Dokument

Opis: Vrednosti predstavljajo vsakokratni datum in čas, ko se je izvedela sprememba stopnje tajnosti entitete. Gre za vrednost, ki jo strežnik samodejno določi. Isto ležne vrednosti lahko administrator povezuje z atributoma »sys:scc: Reason«, »sys:scc: Agent«, »sys:scc: From« in »sys:scc: To« za popolnejšo informacijo o spremembi stopnje tajnosti entitete. Vrednost se vnese v metapodatke entitete, ki se ji spremeni stopnja tajnosti, ne pa tudi v vse njej podrejene entitete. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:scc:From«

Tip: UInt32

Lastnosti: Searchable, MultiValue, ReadOnly, AppendOnly, PickList

Omejitve: Vrednosti morajo ustrezati vsem vrednostim iz atributa "sys:SecurityClass", ki so kadarkoli od namestitve produkta obstajale. Dodatno je potrebno določiti vrednost za "0", ki je vnaprej nastavljena na sinonim »Unspecified«, možno pa jo je tudi spremeniti.

Uporaba: Razred, Zadeva, Dokument

Opis: Vrednosti predstavljajo vsakokratno stopnjo tajnosti pred spremembo le-te.
Gre za vrednost, ki jo strežnik samodejno določi iz vrednosti atributa "sys:scc" pred njegovo spremembo. Isto ležne vrednosti lahko uporabnik povezuje z atributoma »sys:scc: Reason«, »sys:scc: Agent«, »sys:scc: DateTime« in »sys:scc: To« za popolnejšo informacijo o spremembi stopnje tajnosti entitete. Vrednost se vnese v metapodatke entitete, ki se ji spremeni stopnja tajnosti, ne pa tudi v vse njej podrejene entitete. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:scc:To«

Tip: UInt32
Lastnosti: Searchable, MultiValue, ReadOnly, AppendOnly, PickList
Omejitve: Vrednosti morajo ustrezati vsem vrednostim iz atributa »sys:SecurityClass«, ki so kadarkoli od namestitve produkta obstajale. Dodatno je potrebno določiti vrednost za »0«, ki je vnaprej nastavljena na sinonim »Unspecified«, možno pa jo je tudi spremeniti.
Uporaba: Razred, Zadeva, Dokument
Opis: Vrednosti predstavljajo vsakokratno stopnjo tajnosti po spremembi le-te.
Gre za vrednost, ki jo strežnik samodejno določi iz vrednosti atributa »sys:scc« po njegovi spremembi. Isto ležne vrednosti lahko administrator povezuje z atributoma »sys:scc: Reason«, »sys:scc: Agent«, »sys:scc: DateTime« in »sys:scc: From« za popolnejšo informacijo o spremembi stopnje tajnosti entitete. Vrednost se vnese v metapodatke entitete, ki se ji spremeni stopnja tajnosti, ne pa tudi v vse njej podrejene entitete. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

3.2.6 Atributi dokumentov elektronske pošte

Za arhiviranje elektronske pošte so v strežniku IMiS®/ARChive Server vnaprej določeni atributi in predloge, ki so namenjeni za hrambo metapodatkov o elektronskih sporočilih.
Ti atributi so na strežniku vnaprej določeni in z njihovimi lastnostmi ni možno upravljati.

Atribut »sys:eml:MessageId«

Tip: String100
Lastnosti: Searchable, ReadOnly
Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.
Uporaba: Dokument

Opis: Vrednost predstavlja unikatni identifikator sporočila, določenega s strani poštnega strežnika ob dostavi. Vrednost posreduje odjemalec, navadno ga izlušči iz samega sporočila elektronske pošte, čeprav je natančnost informacije odvisna od samega odjemalca.
Vrednost predstavlja vrednost iz atributa »message-id« sporočila po specifikaciji RFC 2822. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:eml:Date«

Tip: DateTime
Lastnosti: Searchable, Required, ReadOnly, IsNonEmpty
Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.
Uporaba: Dokument

Opis: Vrednost predstavlja datum in čas pošiljanja sporočila. Po specifikacijah je to trenutek, ko se pošiljatelj odloči, da je sporočilo primerno za pošiljanje in začne postopek pošiljanja sporočila. Vrednost posreduje odjemalec, navadno ga izlušči iz samega sporočila elektronske pošte, čeprav je natančnost informacije odvisna od samega odjemalca. Vrednost predstavlja vrednost iz atributa »orig-date« sporočila po specifikaciji RFC 2822. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:eml:From«

Tip: String200
Lastnosti: Searchable, Required, ReadOnly, IsNonEmpty
Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.
Uporaba: Dokument
Opis: Vrednost predstavlja veljaven elektronski naslov pošiljatelja elektronskega sporočila. Vrednost posreduje odjemalec, navadno ga izlušči iz samega sporočila elektronske pošte, čeprav je natančnost informacije odvisna od samega odjemalca. Vrednost predstavlja vrednost iz atributa »from« sporočila po specifikaciji RFC 2822. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami,
služi le kot nosilec informacije.

Atribut »sys:eml:To«

Tip: String200

Lastnosti: Searchable, MultiValue, ReadOnly, IsNonEmpty

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Dokument

Opis: Vrednosti predstavljajo veljavne elektronske naslove prejemnikov elektronskega sporočila. Vrednosti posreduje odjemalec, navadno jih izlušči iz samega sporočila elektronske pošte, čeprav je natančnost informacije odvisna od samega odjemalca.
Vrednosti predstavljajo vrednosti iz atributa »to« sporočila po specifikaciji RFC 2822.
Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:eml:ToCC«

Tip: String200

Lastnosti: Searchable, MultiValue, ReadOnly, IsNonEmpty

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Dokument

Opis: Vrednosti predstavljajo veljavne elektronske naslove prejemnikov kopije elektronskega sporočila. Vrednosti posreduje odjemalec, navadno jih izlušči iz samega sporočila elektronske pošte, čeprav je natančnost informacije odvisna od samega odjemalca. Vrednosti predstavljajo vrednosti iz atributa »cc« sporočila po specifikaciji RFC 2822.
Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:eml:ToBCC«

Tip: String200

Lastnosti: Searchable, MultiValue, ReadOnly, IsNonEmpty

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Dokument

Opis: Vrednosti predstavljajo veljavne elektronske naslove skritih prejemnikov kopije elektronskega sporočila. Vrednosti posreduje odjemalec, navadno jih izlušči iz samega sporočila elektronske pošte, čeprav je natančnost informacije odvisna od samega odjemalca.
Vrednosti predstavljajo vrednosti iz atributa »bcc« sporočila po specifikaciji RFC 2822.
Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:eml:Subject«

Tip: String200

Lastnosti: Searchable, ReadOnly

Omejitve: ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Dokument

Opis: Vrednost predstavlja naziv zadeve elektronskega sporočila. Vrednost posreduje odjemalec, navadno ga izlušči iz samega sporočila elektronske pošte, čeprav je natančnost informacije odvisna od samega odjemalca.
Vrednost predstavlja vrednost iz atributa »subject« sporočila po specifikaciji RFC 2822. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:eml:Priority«

Tip: String20

Lastnosti: Searchable, PickList, ReadOnly, IsNonEmpty

Omejitve: Veljavne vrednosti:

- Normal: običajna prioriteta dostave elektronskega sporočila (RFC 1327: »normal«);
- Low: nizka prioriteta dostave elektronskega sporočila (RFC 1327: »non-urgent«);
- High: visoka prioriteta dostave elektronskega sporočila (RFC 1327: »urgent«).

Uporaba: Dokument

Opis: Vrednost predstavlja prioriteto dostave in obdelave sporočila elektronske pošte. Vrednost posreduje odjemalec, navadno ga izlušči iz samega sporočila, čeprav je natančnost informacije odvisna od samega odjemalca. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:eml:Signed«

Tip: Bool

Lastnosti: Searchable, ReadOnly, IsNonEmpty

Omejitve: ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Dokument

Opis: Vrednost predstavlja podatek, ali je elektronsko sporočilo elektronsko podpisano. Vrednost posreduje odjemalec, navadno ga izlušči iz samega sporočila elektronske pošte, čeprav je natančnost informacije odvisna od samega odjemalca.
Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

3.2.7 Atributi upravljanja s fizičnim gradivom

V kolikor elektronski arhiv hrani digitalizirane vsebine/gradivo ali le njihove metapodatke, je nujno vzpostaviti učinkovit sistem povezave elektronskega dela gradiva s povezanim fizičnim gradivom. V ta namen sistem nudi t.i. atribute upravljanja s fizičnim gradivom (angl. Physical Records Management Attributes).

Vzdrževanje aktualnih vrednosti v teh atributih omogoča enostavno, natančno in sledljivo upravljanje s fizičnim gradivom. Za to vzdrževanje so odgovorni skrbniki fizičnega gradiva, ki morebitne izposoje ali prenose vestno beležijo v sistem.

Vsaka sprememba atributov fizičnega gradiva se vnese v revizijsko sled elektronske entitete, ki fizično gradivo opisuje. Zato je sledljivost v tem primeru zagotovljena.

Vsi atributi upravljanja s fizičnim gradivom so registrirani v naslovnem prostoru »prm«.

Teh atributov ni mogoče spreminjati ali jim dodeljevati dodatne lastnosti. Povezani so na sistemske predloge iz katerih izpeljemo predloge, zmožne arhiviranja zadev in/ali dokumentov. Razen vpliva na vnose v revizijsko sled, atributi upravljanja s fizičnim gradivom nimajo vpliva na poslovno logiko strežnika v smislu operacij nad entitetami. Služijo le kot nosilci informacij.

Atribut »sys:prm:Identifier«

Tip: String100

Lastnosti: Searchable, IsNonEmpty, Inherited

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Zadeva, Dokument

Opis: Vrednost predstavlja unikatni identifikator fizičnega gradiva.

Enolično označuje pripadajoče fizično gradivo. Vrednost je spremenljiva v celotnem življenjskem ciklu entitete. Določa jo skrbnik fizičnega gradiva, ki mora imeti možnost spreminjanja metapodatkov entitete. Podatek je neobvezen. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:prm:Description«

Tip: String200

Lastnosti: Searchable, Inherited

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Zadeva, Dokument

Opis: Vrednost predstavlja opis fizičnega gradiva. V podatek kar se da natančno zapišemo opis gradiva, njegov format, fizične nosilce, obseg, ipd. Vrednost je spremenljiva v celotnem življenjskem ciklu entitete. Določa jo skrbnik fizičnega gradiva, ki mora imeti možnost spreminjanja metapodatkov entitete. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami. Služi le kot nosilec informacije.

Atribut »sys:prm:Status«

Tip: String20

Lastnosti: Searchable, PickList, IsNonEmpty, Inherited

Omejitve: Veljavne vrednosti:

- CheckedIn: Fizično gradivo je v hrambi na domači lokaciji. Vrednost se določi, ko fizično gradivo hranimo na njegovi »domači lokaciji«, lokaciji trajne hrambe. Podatek spreminjamo v primeru izposoje ali posredovanja gradiva tretjim osebam.
- CheckedOut: Fizično gradivo je posredovano tretji osebi in NI v hrambi na domači lokaciji. Vrednost se določi, ko fizično gradivo posredujemo, oziroma posodimo tretji osebi in ni v hrambi na njegovi »domači lokaciji«, lokaciji trajne hrambe. Podatek spreminjamo v primeru izposoje ali posredovanja gradiva tretjim osebam. V tem primeru je smiselno osvežiti tudi metapodatek »sys:prm:CurrentLocation«, »sys:prm:Custodian« in »sys:prm:ReturnDue«.

Uporaba: Zadeva, Dokument

Opis: Vrednost predstavlja status fizičnega gradiva glede na njegovo trenutno lokacijo oziroma hrambo. Določa ali spreminja se v primeru izposoje, oziroma posredovanja fizičnega gradiva tretji osebi, ki ga hrani izven domače lokacije. Vrednost je spremenljiva v celotnem življenjskem ciklu entitete. Določa jo skrbnik fizičnega gradiva, ki mora imeti možnost spreminjanja metapodatkov entitete. Datum in čas zadnje spremembe se zabeleži v metapodatek »sys:prm:StatusChange«. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:prm:StatusChange«

Tip: DateTime

Lastnosti: Searchable, ReadOnly, IsNonEmpty, Inherited

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Zadeva, Dokument

Opis: Vrednost predstavlja datum in čas zadnje spremembe statusa fizičnega gradiva, ostale spremembe so vidne v revizijski sledi entitete. Vrednost se določi samodejno s strani strežnika ob spremembi vrednosti atributa »sys:prm:Status« in ni spremenljiva. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:prm:HomeLocation«

Tip: String100

Lastnosti: Searchable, IsNonEmpty, Inherited

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Zadeva, Dokument

Opis: Vrednost predstavlja opis domače lokacije fizičnega gradiva.

V podatek kar se da natančno zapišemo »domačo lokacijo« gradiva, kjer je gradivo v trajni hrambi (naslov, soba, omara, fascikel, ...). Vrednost je spremenljiva v celotnem življenjskem ciklu entitete. Določa jo skrbnik fizičnega gradiva, ki mora imeti možnost spreminjanja metapodatkov entitete. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:prm:CurrentLocation«

Tip: String100

Lastnosti: Searchable, IsNonEmpty, Inherited

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Zadeva, Dokument

Opis: Vrednost predstavlja opis trenutne lokacije fizičnega gradiva, če ta ni domača in če fizično gradivo izposojamo oziroma dajemo v hrambo tretji osebi.

V podatek kar se da natančno zapišemo zunanjo lokacijo gradiva, kjer je gradivo v trenutni hrambi (naslov, soba, omara, fascikel, ...).

V tem primeru smiselno spremenimo tudi vrednost atributa »sys:prm:Status« v »CheckedOut«.

Vrednost je spremenljiva v celotnem življenjskem ciklu entitete.

Določa jo skrbnik fizičnega gradiva, ki mora imeti možnost spreminjanja metapodatkov entitete.

Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:prm:Custodian«

Tip: String100

Lastnosti: Searchable, Inherited

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Zadeva, Dokument

Opis: Vrednost predstavlja identifikacijo trenutnega skrbnika fizičnega gradiva.
Če je ta domača (vrednost atributa »sys:prm:Status« je »CheckedIn«), je to navadno skrbnik fizičnega gradiva. Če je zunanja (vrednost atributa »sys:prm:Status« je »CheckedOut«) je to oseba, ki ji je bilo gradivo zaupano za omejen čas.
Vrednost je spremenljiva v celotnem življenjskem ciklu entitete.
Določa jo skrbnik fizičnega gradiva, ki mora imeti možnost spreminjanja metapodatkov entitete.
Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:prm:ReturnDue«

Tip: Date

Lastnosti: Searchable, Inherited

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Zadeva, Dokument

Opis: Vrednost predstavlja datum in čas do katerega je potrebno fizično gradivo vrniti v domačo hrambo. Za vračilo je odgovoren skrbnik gradiva, naveden v atributu »sys:prm:Custodian«.
Vrednost je spremenljiva v celotnem življenjskem ciklu entitete.
Določa jo skrbnik fizičnega gradiva, ki mora imeti možnost spreminjanja metapodatkov entitete.
Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami,
služi le kot nosilec informacije.

3.2.8 Atributi prenesenega gradiva

Pri prenosu hranjenega gradiva iz drugih elektronskih arhivov je potrebno nekatere atribute prenesenega gradiva hraniti v t.i. sistemskih atributih, da zagotavljamo kontinuiteto življenjskega cikla hranjenega gradiva. Nekateri sistemski atributi tretjih sistemov se vnašajo v sistemske atribute strežnika IMiS®/ARChive Server.
Za določene to ni možno,oziroma zaradi standardov ni dovoljeno.
V ta namen so v strežniku vnaprej določeni atributi, ki hranijo tovrstne metapodatkovne vrednosti.

Pri postopku »Uvoza« ali »Prenosa« gradiva v IMiS®/ARChive Server, strežnik samodejno dodeljuje vrednosti tem atributom na podlagi informacij, ki jih prejema s strani odjemalca, ki izvaja operacijo.
Pravico operaciji »Uvoza« določa prisotnost vloge »UvozIzvoz« (angl. ImportExport).
Pravico operaciji »Prenosa« določa prisotnost vloge »Pregledi« (angl. Reviews).
Atributi prenesenega gradiva nimajo vpliva na poslovno logiko strežnika v smislu operacij nad entitetami, služijo le kot nosilci informacij.

Atribut »sys:trf:AuditLog«

Tip: StringMax
Lastnosti: ReadOnly
Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.
Uporaba: Razred, Zadeva, Dokument
Opis: Vrednost predstavlja revizijsko sled entitete iz prejšnjega sistema.
Kljub temu, da je informacija ključna za kontinuiteto in revizijo življenjskega cikla entitete, podatek ni obvezen. Ni namreč nujno, da jo tretji sistem pri izvozu doda v metapodatkovno shemo izvožene entitete.
V primeru izvoza iz ISUD se podatek izvozi v metapodatkovno shemo in ga lahko tretji ISUD ponovno uvozi v atribute prenesenih entitet. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:trf:SystemId«

Tip: String100
Lastnosti: ReadOnly
Omejitve: ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.
Uporaba: Razred, Zadeva, Dokument
Opis: Vrednost predstavlja sistemski identifikator entitete iz prejšnjega sistema.
Kljub temu, da je informacija ključna za kontinuiteto in revizijo življenjskega cikla entitete, podatek ni obvezen. Ni namreč nujno, da jo tretji sistem pri izvozu doda v metapodatkovno shemo izvožene entitete. Podatek je možno uporabiti za sledljivost instance entitete iz prejšnjega ISUD z instanco entitete v tem ISUD.
V primeru izvoza iz ISUD se podatek izvozi v metapodatkovno shemo in ga lahko tretji ISUD ponovno uvozi v atribute prenesenih entitet. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:trf:ClassificationCode«

Tip: String100

Lastnosti: ReadOnly, IsNonEmpty

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Razred, Zadeva, Dokument

Opis: Vrednost predstavlja klasifikacijsko oznako entitete iz prejšnjega sistema.
Kljub temu, da je informacija ključna za kontinuiteto in revizijo življenjskega cikla entitete, podatek ni obvezen. Ni nujno namreč, da jo tretji sistem pri izvozu doda v metapodatkovno shemo izvožene entitete. Podatek je možno uporabiti za sledljivost instance entitete iz prejšnjega ISUD z instanco entitete v tem ISUD.
V primeru izvoza iz ISUD se podatek izvozi v metapodatkovno shemo in ga lahko tretji ISUD ponovno uvozi v atribute prenesenih entitet. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:trf:Imported«

Tip: DateTime

Lastnosti: ReadOnly, Searchable, IsNonEmpty

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Razred, Zadeva, Dokument

Opis: Vrednost predstavlja datum in čas uvoza v primeru, da gre za »Uvoz«.
Ker se v sistemskem atributu »sys:Created« zabeleži dejanski čas nastanka entitete (prenos iz prejšnjega ISUD) je nujno, da novi ISUD v primeru »Uvoza« ali »Prenosa« zabeleži tudi datum in čas ustvarjanja instance entitete v novem ISUD.
Vrednost določi ISUD samodejno in ni posredovana s strani odjemalca.
V primeru izvoza iz ISUD se podatek izvozi v metapodatkovno shemo in ga lahko tretji ISUD ponovno uvozi v atribute prenesenih entitet. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:trf:Evidence«

Tip: StringMax

Lastnosti: ReadOnly

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Razred, Zadeva, Dokument

Opis: Vrednost predstavlja dokazni zapis avtentičnosti entitete iz prejšnjega ISUD v primeru, da gre za »Uvoz«.
V primeru izvoza iz ISUD se podatek izvozi v metapodatkovno shemo in ga lahko tretji ISUD ponovno uvozi v atribute prenesenih entitet. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:trf:MoveReason«
Tip: String200
Lastnosti: Searchable, ReadOnly, Multivalue
Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.
Uporaba: Razred, Zadeva, Dokument
Opis: Vrednosti predstavljajo razloge za vsakokratni premik (re-klasifikacijo) entitete v prejšnjem ISUD (v primeru, da gre za »Uvoz«). Isto ležne vrednosti lahko uporabnik povezuje z atributoma »sys:trf:MoveAgent«, »sys:trf:MoveDateTime« in »sys:trf:MoveClassificationCode« za popolnejšo informacijo o premiku entitete v hierarhiji razvrščanja gradiva prejšnjega ISUD.
V primeru izvoza iz ISUD se podatek izvozi v metapodatkovno shemo in ga lahko tretji ISUD ponovno uvozi v atribute prenesenih entitet. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:trf:MoveAgent«
Tip: String200
Lastnosti: Searchable, ReadOnly, Multivalue
Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.
Uporaba: Razred, Zadeva, Dokument
Opis: Vrednosti predstavljajo vsakokratne entitete imenika (uporabnike),
ki so izvajali premike (reklasifikacijo) entitete v prejšnjem ISUD (v primeru, da gre za »Uvoz«). Isto ležne vrednosti lahko uporabnik povezuje z atributoma »sys:trf:MoveReason«, »sys:trf:MoveDateTime« in »sys:trf:MoveClassificationCode« za popolnejšo informacijo o premiku entitete v hierarhiji razvrščanja gradiva prejšnjega ISUD.
V primeru izvoza iz ISUD se podatek izvozi v metapodatkovno shemo in ga lahko tretji ISUD ponovno uvozi v atribute prenesenih entitet. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:trf:MoveDateTime«

Tip: DateTime

Lastnosti: Searchable, ReadOnly, Multivalue

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Razred, Zadeva, Dokument

Opis: Vrednosti predstavljajo vsakokratni datum in čas, ko se je izvedel premik (reklasifikacija) entitete v prejšnjem ISUD (v primeru, da gre za »Uvoz«). Isto ležne vrednosti lahko uporabnik povezuje z atributoma »sys:trf:MoveAgent«, »sys:trf:MoveReason« in »sys:trf:MoveClassificationCode« za popolnejšo informacijo o premiku entitete v hierarhiji razvrščanja gradiva prejšnjega ISUD. V primeru izvoza iz ISUD se podatek izvozi v metapodatkovno shemo in ga lahko tretji ISUD ponovno uvozi v atribute prenesenih entitet. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:trf:MoveClassificationCode«

Tip: String200

Lastnosti: Searchable, ReadOnly, Multivalue

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Razred, Zadeva, Dokument

Opis: Vrednosti predstavljajo vsakokratno klasifikacijsko oznako entitete ob premiku (reklasifikaciji) entitete v prejšnjem ISUD (v primeru, da gre za »Uvoz«). Isto ležne vrednosti lahko povezujemo z atributoma »sys:trf:MoveAgent«, »sys:trf:MoveReason« in »sys:trf:MoveDateTime« za popolnejšo informacijo o premiku entitete v hierarhiji razvrščanja gradiva prejšnjega ISUD. V primeru izvoza iz ISUD se podatek izvozi v metapodatkovno shemo in ga lahko tretji ISUD ponovno uvozi v atribute prenesenih entitet. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

3.2.9 Atributi politik hrambe in zadržanj

Za politike hrambe (angl. retention policies) so na strežniku IMiS®/ARChive Server vnaprej določeni atributi in predloge, ki so namenjeni shranjevanju metapodatkov politik hrambe. Atribute lahko razdelimo v naslednji skupini:

- atributi, ki vplivajo na delovanje politik hrambe
- atributi, ki ne vplivajo na delovanje in so samo nosilci informacije o politikah hrambe.

V nadaljevanju so podrobneje opisani posamezni atributi in njihov pomen:

Atribut »sys:ret:pol:Action«

Tip: Uint32

Lastnosti: Public, Required, Searchable, Picklist, IsNonEmpty, [ReadOnly v primeru Kopije politik hrambe]

Omejitve: Veljavne vrednosti:

- 1, alias »Dispose«: privzeta akcija politike hrambe je uničenje entitet;
- 2, alias »Permanent«: privzeta akcija politike hrambe je trajna hramba entitet;
- 3, alias »Transfer«: privzeta akcija politike hrambe je prenos entitet v tretji arhivski sistem, ter po potrditvi o uspešnem prenosu njihovo uničenje;
- 4, alias »Review«: privzeta akcija politike hrambe je, da se entiteto pusti za naslednji postopek odbiranja in izločanja.

Uporaba: Politike hrambe, Kopija politike hrambe

Opis: Obvezen atribut, ki predstavlja privzeto akcijo politike hrambe, ki se uporablja v postopku odbiranja in izločanja. Operacije, ki se za posamezno akcijo zgodijo so naslednje:

- Akcija »Dispose«: če ima uporabnik v imenu katerega se izvaja akcija pravico brisanja na entiteti, se entiteta nepreklicno izbriše, izbrišejo se vsi njeni metapodatki razen tistih, ki jih vsebuje izbrisana entiteta. Operacija je nepovratna kar pomeni, da se take entitete in njenih metapodatkov ne da več povrniti.
- Akcija »Permanent«: če ima uporabnik v imenu katerega se izvaja akcija pravico branja in pisanja na entiteti, se entiteta označi kot trajna. Take entitete ni mogoče več urejati, prav tako je onemogočeno ustvarjanje vsebovanih entitet.
- Akcija »Transfer«: če ima uporabnik, v imenu katerega se izvaja akcija pravico brisanja na entiteti in je v postopku odbiranja in izločanja potrdil uspešen prenos entitete, se entiteta nepovratno uniči. V atribut »sys:ret:pol:Reference« na izbrisani entiteti se zapiše referenca na preneseno entiteto, v kolikor je referenca vpisana v postopku odbiranja in izločanja.
- Akcija »Review«: entiteto se pusti za naslednji postopek odbiranja in izločanja.

Vse naštete akcije se zabeležijo v revizijsko sled entitete z razlogom, ki je naveden v postopku odbiranja in izločanja in komentarjem (v kolikor je prisoten).

Atribut »sys:ret:pol:DetailedDescription«

Tip: StringMax

Lastnosti: Public, [ReadOnly v primeru Kopije politik hrambe]

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Politike hrambe, Kopija politik hrambe

Opis: Neobvezen podroben opis rokov hrambe entitete, ki je spremenljiv v celotnem življenjskem ciklu entitete. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:ret:pol:Reason«

Tip: String200

Lastnosti: Public, Required, IsNonEmpty, [ReadOnly v primeru Kopije politik hrambe]

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Politike hrambe, Kopija politik hrambe

Opis: Obvezen atribut, njegova vrednost se uporabi kot privzet komentar za akcijo v postopku odbiranja in izločanja. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:ret:pol:Reference«

Tip: String200

Lastnosti: Public, ReadOnly, Searchable

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Kopija politike hrambe

Opis: Obvezen atribut, ki ga v postopku priprave odbiranja in izločanja določi strežnik in je nespremenljiv. Vsebuje referenco na politike hrambe, ki je vključena v pripravo odbiranja in izločanja. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:ret:pol:Trigger«

Tip: String200

Lastnosti: Public, Required, [ReadOnly v primeru Kopije politik hrambe]

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Politike hrambe, Kopija politik hrambe

Opis: Obvezen atribut, ki mora vsebovati veljaven pogoj za iskanje po metapodatkih (glej poglavje 3.5.2 Pravila iskalnega niza), v nasprotnem primeru se postopek priprave odbiranja in izločanja, ki vsebuje politiko hrambe z neveljavnim pogojem prekine z napako.

Atribut »sys:ret:hold:Reason«

Tip: String200

Lastnosti: Public, Required, ReadOnly

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Zadržanje postopkov odbiranja in izločanja

Opis: Obvezen atribut, ki vsebuje razlog, zakaj je prišlo do zadržanja postopkov odbiranja in izločanja. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

3.2.10 Atributi postopkov odbiranja in izločanja

Za politike hrambe (angl. retention policies) so na strežniku IMiS®/ARChive Server vnaprej določeni atributi in predloge, ki so namenjeni shranjevanju metapodatkov o samem postopku odbiranja in izločanja. Kot pri politikah hrambe in zadržanju odbiranja (glej poglavje 3.3.9 Roki hrambe) lahko atribute razdelimo na tiste, ki vplivajo na samo izvajanje postopkov odbiranja in izločanja in na tiste, ki so samo nosilci informacij.

V nadaljevanju so podrobneje opisani posamezni atributi in njihov pomen:

Atribut »sys:ret:rev:Action«

Tip: UInt32

Lastnosti: Searchable, Picklist, IsNonEmpty

Omejitve: Veljavne vrednosti:

- 1, alias »Reviewing«: vrednost predstavlja akcijo pregleda v postopku odbiranja in izločanja in na strežniku nima vpliva;
- 2, alias »Complete«: vrednost predstavlja akcijo zaključka v postopku odbiranja in izločanja na strežniku;
- 3, alias »Discard«: vrednost predstavlja akcijo preklica v postopku odbiranja in izločanja na strežniku.

Uporaba: Postopek odbiranja in izločanja

Opis: Neobvezen atribut, ki pa je ključen za končanje ali zavrženje postopka odbiranja in izločanja na strežniku. Uporaba atributa je naslednja:

- Ob ustvarjanju postopka odbiranja in izločanja je atribut nima vrednosti.
- Ob pregledu postopka odbiranja in izločanja se lahko atribut postavi na »Reviewing«, kar pa ni nujno. S tem je postopek odbiranja in izločanja označen, da je v postopku pregledovanja.
- Po končanem pregledu se vrednost postavi na »Complete« ali »Discard«.
  V vsakem primeru gre zahtevek v čakalno vrsto za izvršitev akcije.
  V primeru vrednosti »Complete« se bo postopek odbiranja in izločanja izvršil, v primeru vrednosti »Discard« pa se postopek zavrže. V obeh primerih se status odbiranja in izločanja postavi na »Closed«. Kakršno koli spreminjanje postopka odbiranja in izločanja je onemogočeno.

Atribut »sys:ret:rev:Comments«

Tip: StringMax

Lastnosti: /

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti.

Uporaba: Postopek odbiranja in izločanja

Opis: Neobvezen atribut, uporablja se ga za vpis različnih komentarjev, obrazložitev ter ostalih informacij, ki so kakor koli povezane s postopkom odbiranja in izločanja. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:ret:rev:Lists«

Tip: File

Lastnosti: Multivalue, ReadOnly, AppendOnly, IncludeInAIP, Searchable

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti. Ena entiteta lahko vsebuje do 65536 vrednosti.

Uporaba: Postopek odbiranja in izločanja

Opis: Atribut predstavlja kontejner digitalnih vsebin. V primeru postopka odbiranja in izločanja so to XML dokumenti, ki predstavljajo rezultat postopka priprave odbiranja in izločanja. XML dokumente pripravi in zapiše strežnik in so za uporabnika nespremenljivi.
Atribut je vključen v skupino atributov, ki se indeksirajo. Posebnost tipa atributa »File« je, da to predstavlja indeks po polnem besedilu (Full Text Index). V kolikor je vključen sistem za zagotavljanje avtentičnosti in nespremenljivosti v času hrambe gradiva, postanejo prstni odtisi vrednosti iz tega kontejnerja del arhivskega informacijskega paketa (AIP).

Atribut »sys:ret:rev:Members«

Tip: String200

Lastnosti: Multivalue, Required, Searchable, IsNonEmpty

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti

Uporaba: Postopek odbiranja in izločanja

Opis: Atribut je obvezen in predstavlja člane komisije, ki so prisotni v postopku odbiranja in izločanja. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:ret:rev:Message«

Tip: String200

Lastnosti: Public, ReadOnly

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti

Uporaba: Postopek odbiranja in izločanja

Opis: Atribut ni obvezen in je namenjen strežniku, da zapiše kratek opis napake v primeru, da je pri izdelavi ali izvajanju postopka odbiranja in izločanja prišlo do napake.
V primeru, da je izvajanje postopka odbiranja in izločanja uspešno, se to zabeleži v vrednost atributa. Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:ret:rev:Query«

Tip: String200

Lastnosti: ReadOnly, Searchable

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti

Uporaba: Postopek odbiranja in izločanja

Opis: Vrednost atributa ni obvezna, mora pa biti prisotna pri izdelavi postopka odbiranja in izločanja, če vrednost atributa »sys:ret:rev:Schedules« ni prisotna. Prav tako ni dovoljeno, da imata oba atributa »sys:ret:rev:Query« in »sys:ret:rev:Schedules« nastavljene vrednosti. Vrednost mora vsebovati veljaven pogoj za iskanje po metapodatkih (glej poglavje 3.5.2 Pravila iskalnega niza). V nasprotnem primeru se priprava postopka odbiranja in izločanja, ki vsebuje politike hrambe z neveljavnim pogojem prekine z napako.

Atribut »sys:ret:rev:Schedules«

Tip: String200

Lastnosti: Multivalue, ReadOnly, Searchable, IsNonEmpty

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti

Uporaba: Postopek odbiranja in izločanja

Opis: Vrednost atribut ni obvezna, mora pa biti prisotna pri izdelavi postopka odbiranja in izločanja, če vrednost atributa »sys:ret:rev:Query« ni prisotna. Prav tako ni dovoljeno, da imata oba atributa »sys:ret:rev:Query« in »sys:ret:rev:Schedules« nastavljene vrednosti. Vrednost mora vsebovati veljaven pogoj za iskanje po metapodatkih (glej poglavje 3.5.2 Pravila iskalnega niza). V nasprotnem primeru se postopek priprave odbiranja in izločanja, ki vsebuje politike hrambe z neveljavnim pogojem prekine z napako.

Atribut »sys:ret:rev:Scope«

Tip: String200

Lastnosti: ReadOnly, Searchable, IsNonEmpty

Omejitve: Ni omejitev razen tistih, ki izhajajo iz tipa in lastnosti

Uporaba: Postopek odbiranja in izločanja

Opis: Vrednost atributa ni obvezna, predstavlja pa klasifikacijsko oznako entitete, pod katero se bo izvedel postopek priprave odbiranja in izločanja. Če vrednost ni prisotna, potem se priprava izvede na celotnem arhivu.

Atribut »sys:ret:rev:State«

Tip: Uint32

Lastnosti: Public, Required, ReadOnly, Searchable, Picklist, IsNonEmpty

Omejitve: Veljavne vrednosti:

- 0, alias »Unknown«: vrednost atributa predstavlja neveljavno stanje postopka odbiranja in izločanja;
- 1, alias »Created«: vrednost atributa nastavi strežnik, ko uporabnik ustvari nov postopek odbiranja in izločanja;
- 2, alias »Preparing«: vrednost atributa nastavi strežnik v postopku izdelave vsebine za postopek odbiranja in izločanja;
- 3, alias »InReview«: vrednost atributa nastavi strežnik ob uspešni izdelavi vsebine za postopek odbiranja in izločanja;

- 4, alias »Completing«: vrednost atributa nastavi strežnik ob začetku izvajanja postopka odbiranja in izločanja;
- 5, alias »Completed«: vrednost atributa nastavi strežnik ob uspešno izvedenem postopku odbiranja in izločanja;
- 6, alias »Discarded«: vrednost atributa nastavi strežnik ob uspešnem zavrženju postopka odbiranja in izločanja;
- 7, alias »Failed«: vrednost atributa nastavi strežnik v primeru, da je pri izvedbi ali zavrženju prišlo do nepopravljive napake.

Uporaba: Postopek odbiranja in izločanja

Opis: Vrednost atributa predstavlja stanje postopka odbiranja in izločanja. Vrednost »Preparing« pomeni, da strežnik trenutno izdeluje vsebino za postopek odbiranja in izločanja. Vrednost »Completing« pomeni, da strežnik izvaja postopek odbiranja in izločanja.

Vrednost »InReview« pomeni, da je strežnik uspešno izdelal vsebino za postopek odbiranja in izločanja ter da jo uporabnik lahko pregleda, nastavi želene akcije ipd.
Spreminjanje postopka odbiranja in izločanja (npr. spremembe akcij, vnos komentarjev ali razlogov, ipd.) je dovoljeno samo, če je vrednost atributa postavljena na »Created« ali na »InReview«, v vseh ostalih primerih spreminjanje postopka odbiranja in izločanja ni dovoljeno.
V primeru vrednosti »Failed« se v atribut »sys:ret:rev:Message« zapiše sporočilo, zakaj izvedba ali izdelava postopka odbiranja in izločanja ni bila uspešna.

3.3 Entiteta

Entiteta je samostojni kontejner (angl.Container) podatkov in vsebin, ki zaokrožujejo logične celote informacije. Enolično jo identificiramo preko naslednjih treh različnih identifikatorjev:

- notranji ali sistemski (obvezen, samodejno določen ob ustvarjanju);
- klasifikacijska oznaka (obvezna, samodejno ali ročno določena ob ustvarjanju);
- zunanji (en ali več, neobvezen, zunanje določen in spremenljiv skozi celoten časovni cikel).

Vsako entiteto določajo:

- vrsta,oziroma tip
- klasifikacijska oznaka
- sistemski atributi (naziv, opis, podatki življenjskega cikla, status, avtor, ...)
- lastnosti specifične za entiteto.

Entiteta je abstrakten konstrukt, ki hrani podatke in vsebine objekta, ki ga hrani (npr. zadeva, fizičen dokument, skupina dokumentov, ...).
Različne vrste entitet so specializirane glede na vrste objektov,ki jih opisujejo ali hranijo in zaradi tega poleg sistemskih lastnosti, dobijo še objektu prilagojene lastnosti.
Obenem je entiteta tudi nosilec dostopnih pravic, ki uporabnikom določajo dovoljena in nedovoljena dejanja nad objekti, ki jih hranijo.
Vsaka entiteta ima svoj življenjski cikel, ki se beleži v njeni revizijski sledi.

3.3.1 Vrste

V strežniku IMiS®/ARChive Server so določene naslednje vrste entitet:

- razred
- zadeva in vsebovana zadeva
- dokument.

Razred

Razredi so namenjeni razvrščanju gradiva glede na njegovo vsebino,oziroma poslovne dejavnosti organizacije. So osnovni gradniki načrta razvrščanja gradiva.
Razredi lahko združujejo zadeve ali dokumente po:

- tipu vsebovanih dokumentov (npr. vsi računi se nahajajo v enem razredu, vse naročilnice pa v drugem);
- lastniku vsebovanih dokumentov (npr. kadrovski oddelek ima svoje dokumente v enem razredu, prodajnega oddelka pa v drugem).

Število razredov in njihova vsebina ni predpisana s strani strežnika IMiS®/ARChive Server. Upravitelj načrta razvrščanja gradiva jih lahko poljubno konfigurira glede na potrebe organizacije, ki arhivski strežnik uporablja.

Zadeva

Zadeva predstavlja skupino entitet (vsebovane zadeve, dokumenti), ki vsebinsko zaokrožujejo celoto. Predstavljajo dosje obravnavane zadeve (vsebinsko vprašanje, tema, naloga, projekt, ...) z vsemi pripadajočimi lastnostmi in vsebinami.
Je osnovna enota združevanja, evidentiranja, razvrščanja in arhiviranja dokumentov.

Dokument

Dokument predstavlja kontejner lastnosti in vsebine, ki jo hrani. En dokument lahko hrani več digitalnih vsebin (npr. besedilo, slika, video). Navadno je vsebovan v zadevah in podrejenih zadevah, lahko pa nastopa tudi samostojno v razredu. Predstavlja osnovno enoto arhivskega gradiva, ki hrani vsebine.

3.3.2 Hierarhija

Vsak elektronski arhiv je organiziran drevesno, ki se začenja z več debli (razredi na prvem nivoju) in vejami, ki jih predstavljajo posamezne entitete.
Končni razredi vsebujejo entitete vsebine (zadeva, vsebovana zadeva, dokument).
Strežnik IMiS®/ARChive Server ne omejuje globine hierarhije načrta razvrščanja gradiva.

Pri določanju načrta razvrščanja gradiva je potrebno upoštevati spodaj našteta pravila:

- Na prvem nivoju načrta razvrščanja gradiva moramo obvezno določiti enega ali več razredov. Onemogočeno je dodajanje zadev ali dokumentov v koren drevesa.
- Vsaka entiteta, ki je zmožna vsebovanja lahko vsebuje podrejene entitete ene vrste.
- Razred lahko vsebuje podrazrede.
- Končni razred lahko vsebuje zadeve ali dokumente.
- Zadeva lahko vsebuje podrejene zadeve.
- Končne zadeve lahko vsebujejo samo dokumente.
- Dokument ni zmožen vsebovanja podrejenih entitet.

Hierarhija entitet v načrtu razvrščanja gradiva obenem lahko določa tudi dostopne pravice vseh vsebovanih podrejenih entitet (glej poglavje 3.3.5 Dostopi), razen kadar so dostopne pravice natančneje določene v samih podrejenih entitetah (glej poglavje 3.3.5.2.5 Eksplicitna dovoljenja ali prepovedi). Istočasno hierarhija določa še stopnjo tajnosti (če ni eksplicitno določena; glej poglavje 3.3.5.1 Stopnje tajnosti) ter obveznost statusa entitete (sistemski atribut »sys:Status«).

Status je določen na:

- razredih
- zadevah in vsebovanih zadevah
- dokumentih uvrščenih neposredno pod razredi.

Če vrednost statusa ni eksplicitno nastavljena, potem se podeduje od nadrejene entitete.
To velja tudi za razrede na prvem nivoju, ki dobijo podedovano vrednost »Opened« v primeru, da nimajo eksplicitno nastavljene vrednosti.

3.3.3 Komponente

Entitete sestavljajo različne komponente, med njimi:

- shema sistemskih atributov
- shema specializiranih atributov, specifičnih za vrsto entitete
- shema atributov upravljanja fizičnega gradiva
- shema atributov prenesenega gradiva
- skladišče arhiviranih vsebin s pripadajočimi metapodatki o vsebinah
- komponente dostopnih pravic
- komponente elementov avtentičnosti
- roki hrambe.

Strukturo večini komponent določa predloga, ki je osnova za nastanek entitete (shema specializiranih atributov). Druge komponente so sistemske narave in so vedno prisotne, saj so ključne za obstoj in življenjski cikel entitet (shema sistemskih atributov, itd).
Entiteta je tudi nosilec njenih dostopnih pravic, revizijske sledi in elementov dokaza avtentičnosti.

3.3.4 Predloge

Predloge predpisujejo metapodatkovno shemo - zahtevane in dovoljene atribute.
Hkrati so osnova za izdelavo specializiranih predlog entitet istega tipa, ki podedujejo atributno shemo.
Vsaka predloga vsebuje vgrajene in vnaprej določene sistemske atribute.
Ti so potrebni za pravilno delovanje strežnika IMiS®/ARChive Server in jih ni mogoče spreminjati.
Njihovi nazivi prihajajo iz naslovnega področja »int:«, »sys:«, »trf:«, ...
Vse ostale atribute lahko administrator poljubno dodaja in briše iz predloge,
dokler ni ustvarjena prva entiteta. Potem so možnosti spremembe predloge omejene, omejitve so navedene v nadaljevanju.

Pri dodajanju atributa na predlogo lahko uporabnik z ustreznimi pravicami določi ali:

- je vrednost atributa v okviru neke entitete javnega značaja (angl. Public);
- ima lahko atribut več kot eno vrednost (angl. Multivalue);
- je atribut obvezen oziroma neobvezen (angl. Required);
- se vrednosti atributa po prvi shranitvi lahko spreminjajo (angl. ReadOnly);
- se vrednost atributa deduje iz nadrejene entitete (angl. Inherited). V kolikor v entiteti niso določene eksplicitne vrednosti, se te dedujejo iz prve nadrejene entitete, ki ima določene eksplicitne vrednosti;
- lahko uporabnik spreminja že zapisan atribut oz. dovoli samo dodajanje vsebine atributa brez brisanja že obstoječih vrednosti (angl. Append Only);
- se vrednost atributa v postopku zagotavljanja avtentičnosti hranjenega gradiva vključi v arhivski informacijski paket (angl. IncludeInAIP).

Brisanje predloge je dovoljeno samo v primeru, ko ne obstaja entiteta, ki bi po brisanju postala neveljavna.
Ostalim vrstam entitet (razred, zadeva, dokument) je potrebno dodeliti predlogo in morajo biti zgrajeni po pravilih, ki jih določa metapodatkovna shema predloge. Predloge lahko administrator ustvari na novo ali jih izvede iz kakšne druge predloge pod pogojem, da sta vrsti entitete isti.

Primer: Iz predloge »Račun« izpeljemo predlogi »Vhodni račun« in »Izhodni račun«. Vsaki predlogi posebej tako ni potrebno dodeljevati skupnih atributov.
Prav tako se vse bodoče spremembe predloge »Račun« samodejno upoštevajo v izpeljanih predlogah »Vhodni račun« in »Izhodni račun«.

3.3.5 Dostopi

Dostop do entitet in njihovo upravljanje je ključnega pomena, saj zagotavlja osnovna načela informacijske varnosti:

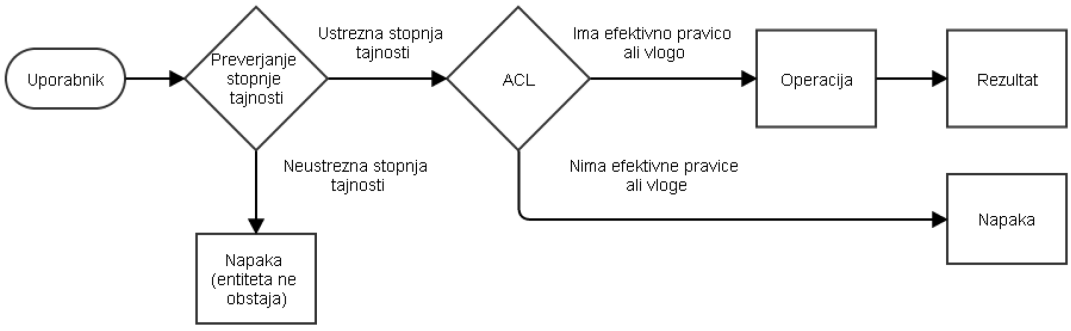
- celovitost (angl. Integrity): zagotavlja nespremenjenosti informacij;
- zaupnost (angl. Confidentiality): zagotavlja, da so informacije dostopne samo pooblaščenim osebam;
- razpoložljivost (angl. Availability): zagotavlja razpoložljivosti informacij pooblaščenim osebam.

Preverjanje pravic dostopa poteka v dveh korakih:

- preverjanje stopnje tajnosti (angl. Security Class);
- preverjanje dostopa do entitet (angl. Access Control Lists - ACL).

Stopnja tajnosti je predpogoj, da lahko uporabnik vidi obstoj entitet in izvaja dejanja nad entitetami. Administrator za vsakega uporabnika ali skupino določi do katerega nivoja tajnosti je avtoriziran.
Uporabnik mora imeti enako ali višjo stopnjo tajnosti, kot jo imajo entitete, ki so bodisi eksplicitno določene ali podedovane od nadrejene entitete. Če stopnja tajnosti uporabniku omogoča videti obstoj entitet, se nadalje preverja dostop do entitet.

Za preverjanje dostopov do entitet, uporablja strežnik IMiS®/ARChive Server koncept liste dostopnih pravic (angl. Access Control Lists - ACL). Za vsako operacijo na strežniku, ki jo izvaja uporabnik s pravicami preveri, ali jo ima pravico izvajati.
V primeru, da uporabnik nima pravice izvajati operacije, se operacija ne izvrši in strežnik vrne napako. Delovanje dostopa predstavlja naslednja slika.



Slika 3: Delovanje ACL

3.3.5.1 Stopnje tajnosti

Stopnja tajnosti je dodaten nivo varnosti dostopa do arhiviranega gradiva zaradi nevarnosti razkritja dokumentov nepooblaščenim osebam. Na nivoju vrste dokumentov administrator nastavi stopnjo tajnosti in za vsakega uporabnika (ali skupino) določi do katerega nivoja tajnosti je avtoriziran. V osnovnih nastavitvah so vse vrste dokumentov dostopne vsem uporabnikom (stopnja tajnosti je 0). Podobno kot pri dostopnih pravicah, se tudi stopnja tajnosti deduje po hierarhiji navzdol. V primeru, da ima entiteta v hierarhiji eksplicitno nastavljeno stopnjo tajnosti, le-ta prevlada nad podedovano vrednostjo.

Pri nastavljanju stopnje tajnosti obstajajo naslednja pravila:

- Največja stopnja tajnosti, ki jo lahko uporabnik nastavi za entiteto je enaka izračunani efektivni stopnji tajnosti.
- Uporabnik lahko nastavi samo večjo ali enako stopnjo tajnosti na entiteti, kot jo ima nadrejena entiteta v hierarhiji. Če nadrejena entiteta nima nastavljene stopnje tajnosti, potem lahko nastavi poljubno vrednost stopnje tajnosti, ki je manjša ali enaka uporabnikovi efektivni stopnji tajnosti.
- Pri dvigovanju se stopnja tajnosti zviša vsem podrejenim entitetam, ki imajo stopnjo tajnosti nižjo ali enako kot entiteta, ki ji dvigujemo stopnjo tajnosti (velja tako za eksplicitno nastavljene stopnje tajnosti kot podedovane).
- Pri spuščanju, se stopnja tajnosti spusti vsem podrejenim entitetam, ki imajo podedovano stopnjo tajnosti.

Primer 1:
Za primer vzemimo hierarhijo, ki jo predstavlja slika v poglavju 3.3.5.2.6 Dostopi – Efektivne pravice. Predpostavimo, da nobena od entitet nima eksplicitno nastavljene stopnje. Razredu nastavimo stopnjo tajnosti na »Restricted«. Zaradi dedovanja se vsem vsebovanim entitetam nastavi stopnja tajnosti na »Restricted«.

Primer 2:
Zadevi iz prejšnjega primera hočemo spustiti stopnjo tajnosti na »Unclassified«. Ker ima nadrejeni razred stopnjo tajnosti »Restricted«, spuščanje stopnje tajnosti ni mogoče.

Primer 3:
Zadevi nastavimo eksplicitno stopnjo tajnosti na »Restricted«, nakar nadrejenemu razredu odvzamemo stopnjo tajnosti. Zadeva in podrejeni dokumenti imajo natavljeno stopnjo tajnosti na »Restricted«.

Primer 4:
Razredu dvignemo stopnjo tajnosti na »Confidential«. Zadevi in podrejenim dokumentom se stopnja tajnosti dvigne na »Confidential«, saj imajo podrejene entitete nižjo stopnjo tajnosti kot entiteta, kateri zvišujemo stopnjo tajnosti.

Primer 5:
Zadevi dvignemo stopnjo tajnosti na »Secret«. Posledično se tudi stopnja tajnosti dokumentov dvigne na »Secret«. Nato razredu spustimo stopnjo tajnosti na »Unclassified«. Stopnja tajnosti zadeve in podrejenih dokumentov se ohrani, saj ima zadeva eksplicitno nastavljeno višjo stopnjo tajnosti kot razred, kateremu spuščamo stopnjo tajnosti.

Kot že omenjeno, ima uporabnik možnost nastavljanja največje stopnje tajnosti glede na njegovo izračunano efektivno stopnjo tajnosti. Podobno kot računanje efektivnih pravic, se računa tudi efektivna stopnja tajnosti za posameznega uporabnika, saj je le-ta odvisna od skupin, ki jim uporabnik pripada.
Efektivna stopnja tajnosti se izračuna na naslednji način:

- Če ima uporabnik eksplicitno nastavljeno stopnjo tajnosti, potem ta vrednost prevlada.
- Če uporabnik nima eksplicitno nastavljene stopnje tajnosti, potem se efektivna stopnja tajnosti računa na podlagi skupin, ki jim uporabnik pripada. Efektivna stopnja tajnosti je največja stopnja tajnosti vseh skupin, ki jim uporabnik pripada (ne glede na njihovo postavitev v hierarhiji entitet).

Primer 1:
Uporabnik ima eksplicitno nastavljeno stopnjo tajnosti na 1 (Unclassified). Hkrati pa je vsebovan v dveh skupinah, ki imata stopnjo tajnosti nastavljeno na 2 (Restricted) in 3 (Confidential). Efektivna stopnja tajnosti za tega uporabnika je tako 1, saj eksplicitno nastavljena stopnja tajnosti prevlada nad stopnjami tajnosti skupin.
Tak uporabnik lahko dostopa do entitet, ki so dostopne vsem uporabnikom (stopnja tajnosti 0) in entitet, ki imajo stopnjo tajnosti nastavljeno na 1 (Restricted).

Primer 2:
Vzemimo uporabnika iz prejšnjega primera in mu pobrišemo eksplicitno nastavljeno stopnjo tajnosti. Tako pridobi efektivno stopnjo tajnosti 3 (Confidential), saj prevlada največja stopnja tajnosti vseh skupin, ki jim pripada. Tako ima uporabnik dostop do entitet z največjo stopnjo tajnosti 3 ali manj.

3.3.5.2 Lista dostopnih pravic (ACL)

Uporabnik s pravico določanja dostopnih pravic na entiteti (angl. Change permissions) lahko dodeli listo dostopnih pravic uporabniku ali uporabniški skupini. Dostopne pravice razdelimo v dve skupini:

- dostopne pravice za entitete
- specializirane dostopne pravice za atribute.

Vsaki skupini pravic lahko uporabnik s pravico določi eksplicitno dovoljenje (angl. Allow) ali prepoved (angl. Deny), ki velja za posamezno pravico znotraj skupine in je lahko tudi časovno omejena. Eksplicitna dovoljenja skupaj s podedovanimi pravicami določajo efektivne pravice (angl. Effective rights), oziroma pravice uporabnika do zahtevane operacije na strežniku.

3.3.5.2.1 Dostopne pravice za entiteto

Za dostop do razredov, zadev in dokumentov se uporabljajo naslednje dostopne pravice:

- Read: pravica branja entitete.
  Uporabnik mora imeti to pravico, če hoče entiteto odpirati v načinu samo za branje (angl. Read-only mode).
- Write: pravica pisanja na entiteti.
  Za urejanje entitete mora imeti uporabnik poleg pravice pisanja tudi pravico branja (angl. Read-write mode).
- Move: pravica premika entitete v načrtu razvrščanja gradiva.
  Uporabniku je omogočeno premikanje entitete in njej podrejenih entitet v načrtu razvrščanja gradiva.
- Delete: pravica brisanja entitete.
  Uporabniku je omogočeno brisanje entitete in njenih komponent.
- Create entities: pravica ustvarjanja novih vsebovanih entitet.
  Uporabnikom pravica omogoča ustvarjanje novih vsebovanih entitet.
- Change permissions: pravica spreminjanja liste dostopnih pravic (ACL).
  Uporabniku je dovoljeno spreminjati dostopne pravice za entitete in metapodatke.
  Spreminjanje dostopnih pravic je mogoče samo, ko je entiteta odprta v načinu za urejanje.
- Change security class: pravica spreminjanja stopnje tajnosti.
  Uporabnikom, katerim je pravica dodeljena, lahko entiteti določajo inicialno stopnjo tajnosti, ko entiteta še ni shranjena, oziroma spreminjajo stopnjo tajnosti obstoječim entitetam. V kolikor uporabnik te pravice nima, entiteti ne more določati inicialne stopnje tajnosti (jo implicitno podeduje od nadrejene entitete) ali jo kasneje spreminjati.
- Change status: pravica spreminjanja statusa.
  Uporabnikom, katerim je pravica dodeljena, lahko spreminjajo status entitete.
  Privzeta vrednost je podedovana iz nadrejene entitete. Preostali eksplicitni vrednosti sta lahko »Odprto« (angl. Opened) ali »Zaprto« (angl. Closed).
  V primeru, da uporabnik zapre entiteto, hkrati zapre tudi vse podrejene entitete.
  V postopku odpiranja že zaprte entitete se status spremeni le neposredni entiteti, podrejene entitete ohranijo njihov status.
- Change retention: pravica spremembe veljavnosti rokov hrambe entitete.
  Uporabniki, katerim je pravica dodeljena, lahko spreminjajo veljavnost rokov hrambe za entiteto. Privzete vrednosti so podedovane iz nadrejenih entitet.

Naslednja tabela prikazuje operacije nad entitetami in zahtevane dostopne pravice za izvajanje le-teh.

Operacija	**Pravica**								
	Read	Write	Delete	Move	Change-permissions	Create-entities	Change security class	Change status	Change retention
Odpiranje entitete v načinu za branje	✓								
Urejanje entitete	✓	✓							
Brisanje entitete	✓		✓						
Urejanje ACL	✓	✓			✓				
Re-klasifikacija (izvorna entiteta)	✓			✓					
Re-klasifikacija (tarčna entiteta)						✓			
Ustvarjanje entitet (nadrejena entiteta)						✓			
Spreminjanje stopnje tajnosti	✓						✓		
Spreminjanje statusa	✓							✓	
Spreminjanje rokov hrambe									✓

Tabela 4: Tabela operacij in pravic

3.3.5.2.2 Dostopne pravice za metapodatke

Za dostop do metapodatkov entitete se uporabljajo naslednje dostopne pravice:

- Read: pravica branja metapodatkov
- Write: pravica spreminjanja vrednosti metapodatkov
- Delete: pravica brisanja metapodatkov
- Create: pravica dodeljevanja vrednosti metapodatkov na predhodno praznih metapodatkih.

Z dostopnimi pravicami za metapodatke strežnik dodatno omeji dostop in urejanje metapodatkov, ki niso javni. Tako lahko uporabniku s pravico branja na določeni entiteti prepreči branje posameznih metapodatkov z uporabo prepovedi »Read« pravice.
Če dostopne pravice za metapodatke niso določene, jih metapodatki prevzamejo iz dostopnih pravic pripadajoče entitete. Prepoved »Delete« pravice za brisanje metapodatkov ne zadrži brisanja entitete in njenih metapodatkov. Če ima uporabnik pravico brisanja entitete in hkrati nima pravice brisanja metapodatkov, potem ima pravica brisanja entitete prednost pred prepovedjo brisanja metapodatkov.

3.3.5.2.3 Izjeme

Metapodatki, ki so izjeme pri nastavljanju dovoljenj ali prepovedi so:

- javni metapodatki
- metapodatki, označeni »samo za branje«
- obvezni metapodatki
- posebni sistemski metapodatki.

Javni metapodatki ne vsebujejo zaupnih informacij, zato so izjema pri preverjanju pravic dostopa. Takim metapodatkom ne moremo nastaviti prepovedi branja, kakor tudi ne moremo nastaviti dovoljenj brez dostopne pravice »Read«. Metapodatke lahko uporabnik bere tudi, če nima pravice branja na entiteti, ki jih vsebuje, če mu to omogoča globalna varnostna nastavitev na strežniku IMiS®/ARChive Server.

Naslednja tabela prikazuje omejitev pravic vpogleda v javne metapodatke uporabnikov, glede na kombinacijo »Read« pravice na entiteti in globalne varnostne nastavitve.

Omejitev vpogleda javnih metapodatkov	**»Read« pravica na entiteti**	**Globalna varnostna nastavitev**
Uporabniku se ne prikaže nobena informacija, tako ni mogoče ugotoviti obstoja entitete (tudi če ima uporabnik zadostno stopnjo tajnosti).		
Uporabniku se prikažejo javni metapodatki. Uporabnik lahko potrdi obstoj entitete, ne more pa videti metapodatkov, ki niso javni.		✓
Če ima uporabnik pravico branja na entiteti, potem ima ne glede na globalno varnostno nastavitev tudi pravico branja javnih metapodatkov.	✓	N/A

Tabela 5: Omejitev pravic vpogleda v javne metapodatke

Metapodatkom, označenim »samo za branje« ne moremo nastaviti »Write« in »Delete« pravic. Obvezni metapodatki pa imajo eksplicitno nastavljeno »Create« pravico ter onemogočeno nastavljanje »Delete« pravice.
Poleg prej naštetih izjem, se posebno obravnavajo še naslednji sistemski metapodatki: sys:ExternalIds, sys:Opened, sys:Closed, sys:Creator. Tem metapodatkom je prepovedano nastavljanje kakršnih koli dovoljenj ali prepovedi.

3.3.5.2.4 Vloge

Vloga je skupek pravic, ki uporabniku omogočajo izvajanje določene operacije na strežniku.
Vnaprej so določene naslednje vloge:

- AuditLogQuery: vloga omogoča pridobivanje revizijske sledi;
- ImportExport: vloga omogoča uvoz in izvoz gradiva;
- ContentManagement: vloga omogoča izvajanje zahtev za indeksiranje in avtomatično pretvorbo vsebine;
- Reports: vloga omogoča:
  - prikaz sistemskih poročil o izvozu in uvozu;
  - prikaz sistemskih poročil o dostopih, zadevah, dokumentih, vsebini dokumentov in rokih hrambe;
  - tiskanje podatkov o razredih, zadevah in dokumentih;
  - tiskanje razredov (in zadev načrta razvrščanja gradiva.
- Review: vloga omogoča prikaz pregledov v postopku odbiranja in izločanja;
- AdminCodeListaRead: vloga omogoča pregled šifrantov;
- AdminCodeListUpdate: vloga omogoča dodajanje, brisanje ali spreminjanje šifranta;
- AdminCountersRead: vloga omogoča pregled določitev števcev;
- AdminCounterUpdate: vloga omogoča dodajanje, brisanje ali spreminjanje določitev števca;
- AdminDirectoryEntitiesRead: vloga omogoča pregled imeniških entitet;
- AdminDirectoryEntityUpdate: vloga omogoča dodajanje, brisanje ali spreminjanje imeniške entitete;
- AdminDirectoryGroupRead: vloga omogoča pregled članov imeniške skupine;
- AdminDirectoryGroupUpdate: vloga omogoča dodajanje ali brisanje članov imeniške skupine;
- AdminAttributesRead: vloga omogoča pregled atributov;

- AdminAttributeUpdate: vloga omogoča dodajanje, brisanje ali spreminjanje atributa;
- AdminTemplatesRead: vloga omogoča branje predlog entitet;
- AdminTemplateUpdate: vloga omogoča dodajanje, brisanje ali spreminjanje predlog entitete;
- AdminStorageProfilesRead: vloga omogoča branje strežniških profilov;
- AdminStorageProfileUpdate: vloga omogoča dodajanje, brisanje ali spreminjanje strežniškega profila;
- AdminStorageVolumesRead: vloga omogoča branje strežniških volumnov;
- AdminStorageVolumeUpdate: vloga omogoča dodajanje, brisanje ali spreminjanje strežniškega volumna;
- AdminACLRead: vloga omogoča branje dostopnih pravic;
- AdminACLUpdate: vloga omogoča dodajanje, brisanje ali spreminjanje dostopnih pravic;
- AdminLegacyArchiveRead: vloga omogoča branje strežniške konfiguracije za starejše odjemalce;
- AdminLegacyArchiveUpdate: vloga omogoča spreminjanje strežniške konfiguracija za starejše odjemalce;
- AdminArchiveSettingsRead: vloga omogoča branje splošnih nastavitev arhiva;
- AdminArchiveSettingsUpdate: vloga omogoča dodajanje, brisanje ali spreminjanje splošnih nastavitev arhiva;
- AdminAuditLogSettingsRead: vloga omogoča branje nastavitev revizijske sledi;
- AdminAuditLogSettingsUpdate: vloga omogoča dodajanje, brisanje ali spreminjanje nastavitev revizijske sledi;
- AdminRetentionRead: vloga omogoča branje politik hrambe;
- AdminRetentionUpdate: vloga omogoča dodajanje, brisanje ali spreminjanje politik hrambe;
- AdminContentSettingsRead: vloga omogoča branje strežniške konfiguracije za upravljanje z vsebino;
- AdminContentSettingsUpdate: vloga omogoča spreminjanje strežniške konfiguracije za upravljanje z vsebino;
- AdminLTANSSettingsRead: vloga omogoča branje strežniške konfiguracije za dolgoročno arhiviranje vsebin;
- AdminLTANSSettingsUpdate: vloga omogoča spreminjanje strežniške konfiguracije za dolgoročno arhiviranje vsebin;

- AdminAAASettingsRead: vloga omogoča branje strežniške konfiguracije za zunanjo avtentikacijo in sinhronizacijo uporabnikov:
- AdminAAASettingsUpdate: vloga omogoča spreminjanje strežniške konfiguracije za zunanjo avtentikacijo in sinhronizacijo uporabnikov;
- AdminSecuritySettingsRead: vloga omogoča branje shrambe digitalnih potrdil;
- AdminSecuritySettingsUpdate: vloga omogoča spreminjanje shrambe digitalnih potrdil;

3.3.5.2.5 Eksplicitna dovoljenja ali prepovedi

Z nastavljanjem eksplicitnih dovoljenj ali prepovedi upravljamo efektivne pravice uporabnika.
Vsaka sprememba dovoljenj in prepovedi se zabeleži v revizijsko sled.
Lastnosti dovoljenj ali prepovedi so naslednje:

- za vsako izmed skupin dostopnih pravic se lahko nastavijo dovoljenja ali prepovedi za uporabnika ali skupino;
- dovoljenja ali prepovedi se lahko časovno omejijo.

Časovne omejitve dovoljenj ali prepovedi lahko ločimo na:

- dovoljenja ali prepovedi brez časovne omejitve
- dovoljenja ali prepovedi z začetkom časovne omejitve
- dovoljenja ali prepovedi s koncem časovne omejitve
- dovoljenja ali prepovedi z začetkom in koncem časovne omejitve.

<u>Dovoljenja ali prepovedi brez časovne omejitve</u>
Standardni način uporabe dovoljenj ali prepovedi. Takšna dovoljenja in prepovedi veljajo vedno, ne glede na to, kdaj se uporabljajo pri izračunu efektivnih pravic.

<u>Dovoljenja ali prepovedi z začetkom časovne omejitve</u>
Dovoljenja ali prepovedi imajo določen začetek veljavnosti, nimajo pa določenega konca.
Takšna dovoljenja in prepovedi se upoštevajo pri izračunu efektivnih pravic v primeru,
da je trenutni datum večji ali enak datumu začetka veljavnosti dovoljenja ali prepovedi.

<u>Dovoljenja ali prepovedi s koncem časovne omejitve</u>
Dovoljenja ali prepovedi imajo določen konec veljavnosti, nimajo pa določenega začetka.
Takšna dovoljenja ali prepovedi se upoštevajo pri izračunu efektivnih pravic v primeru,
da je trenutni datum manjši ali enak datumu konca veljavnosti dovoljenja ali prepovedi.

Dovoljenja ali prepovedi z začetkom in koncem časovne omejitve

Dovoljenja ali prepovedi imajo določen začetek in konec veljavnosti.
Takšna dovoljenja ali prepovedi se upoštevajo pri izračunu efektivnih pravic v primeru, da je trenutni datum večji ali enak začetku veljavnosti, in manjši ali enak koncu veljavnosti dovoljenja ali prepovedi.

Eksplicitna dovoljenja ali prepovedi se avtomatično dedujejo na vse podrejene entitete v pripadajoči hierarhiji.

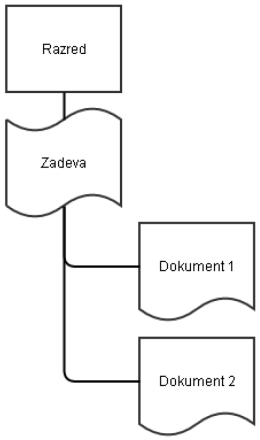
3.3.5.2.6 Efektivne pravice

Efektivne pravice so skupek podedovanih in eksplicitnih dovoljenj ali prepovedi, ki nam povedo, ali ima uporabnik pravico izvajati operacijo na strežniku IMiS®/ARChive Server.

Vrstni red upoštevanja dovoljenj ali prepovedi pri izračunu efektivnih pravic je naslednji:

1. eksplicitna prepoved (angl. Explicit deny rights)
2. eksplicitno dovoljenje (angl. Explicit allow rights)
3. podedovana prepoved (angl. Inherited deny rights)
4. podedovano dovoljenje (angl. Inherited allow rights).

Na izračun efektivnih pravic ne vpliva število skupin kjer je uporabnik naveden, ampak samo hierarhija le-teh v listah dostopnih pravic (ACL). Izračun efektivnih pravic za entitete predstavlja primer hierarhije, ki jo predstavlja slika v nadaljevanju.



Slika 4: Hierarhija z razredom, zadevo in dvema dokumentoma

Primer 1: Za določenega uporabnika želimo nastaviti pravice branja za celotno hierarhijo, ki jo predstavlja zgornja slika. V listi dostopnih pravic (ACL) razreda nastavimo eksplicitno dovoljenje branja entitete (»Read« pravica) za tega uporabnika.
Dedovanje po hierarhiji uporabniku omogoča, da ima efektivne pravice branja na vseh podrejenih entitetah razreda. Tako lahko uporabnik v našem primeru odpira razred, zadevo in oba dokumenta v načinu samo za branje.

Primer 2: Uporabniku iz primera 1 želimo dodati pravice urejanja zadeve in obeh dokumentov v zadevi. Za urejanje entitet sta potrebni pravici branja in pisanja (»Read« in »Write«). Pravico branja ima uporabnik že podedovano od razreda. Zato lahko v listi dostopnih pravic (ACL) zadeve nastavimo samo eksplicitno dovoljenje pisanja za tega uporabnika. Tako uporabniku nastavimo efektivne pravice branja in pisanja na zadevi ter na obeh dokumentih znotraj zadeve.

Primer 3: Iz primerov 1 in 2 izhaja, da ima uporabnik pravice branja vseh entitet iz hierarhije ter urejanja zadeve in obeh dokumentov v zadevi. Uporabniku želimo prepovedati urejanje dokumenta »Dokument 1«. V listi dostopnih pravic (ACL) dokumenta nastavimo eksplicitno prepoved pisanja.
Tako uporabniku na tem dokumentu prepovemo urejanje, saj ima vrstni red upoštevanja prepovedi pri izračunu efektivnih pravic prednost pred podedovanimi dovoljenji. Uporabnik ima tako še vedno pravico urejanja zadeve in dokumenta »Dokument 2«, nima pa pravice urejati dokumenta »Dokument 1«.

Primer 4: V primeru 3 smo uporabniku onemogočili urejanje dokumenta »Dokument 1«. V listi dostopnih pravic (ACL) dokumenta dodamo eksplicitno dovoljenje pisanja.
S tem ne vplivamo na izračun efektivnih pravic, saj ima eksplicitna prepoved pisanja prednost pred eksplicitnim dovoljenjem. Posledično ima uporabnik enake efektivne pravice kot v prejšnjem primeru.

Primer 5: Uporabniku v listi dostopnih pravic (ACL) zadeve dodamo eksplicitno dovoljenje za spreminjanje ACL (pravica »ChangeRights«). Uporabnik tako dobi efektivno pravico urejanja ACL na zadevi ter na dokumentu »Dokument 2«. Na dokumentu »Dokument 1« uporabnik še vedno ne more urejati ACL, saj mu eksplicitna prepoved pisanja to onemogoča.

Primer 6: Uporabniku iz primera 5 odvzamemo pravico spreminjanja liste dostopnih pravic (ACL). Hkrati mu želimo onemogočiti urejanje določenega metapodatka, ki se nahaja na zadevi in obeh dokumentih, ker ni javnega značaja. V ACL zadeve nastavimo eksplicitno prepoved pisanja metapodatka za uporabnika. S tem uporabniku preprečimo urejanje metapodatka kljub temu, da ima pravico urejanja zadeve in dokumenta »Dokument 2«.

Primer 7: Iz liste dostopnih pravic (ACL) razreda uporabniku odstranimo eksplicitno dovoljenje branja. Tako onemogočimo uporabniku odpiranje ter urejanje vseh entitet v hierarhiji.

Primer 8: Iz liste dostopnih pravic (ACL) vseh entitet v hierarhiji za uporabnika odstranimo vsa eksplicitna dovoljenja in prepovedi. V ACL razreda dodamo eksplicitna dovoljenja branja, pisanja in brisanja za entiteto ter eksplicitno prepoved brisanja za vse metapodatke v hierarhiji. S tem uporabniku omogočamo urejanje vseh entitet in metapodatkov v hierarhiji, onemogočamo pa brisanje le-teh. Kljub onemogočenem brisanju metapodatkov, pa lahko uporabnik izbriše celotno entiteto z metapodatki, saj ima pravica brisanja entitete prednost pred prepovedjo brisanja metapodatkov.

Primer 9: Eksplicitnim prepovedim brisanja metapodatkov iz »primera 8«nastavimo začetek časovne omejitve na datum 1.4.2015 (brez konca). Uporabnik tako izgubi efektivne pravice brisanja metapodatkov z dnem 1.4.2015, pred tem datum pa uporabnik lahko normalno briše metapodatke.
Če nastavimo konec časovne omejitve na 1.4.2015 (brez začetka), potem uporabnik do 1.4.2015 nima efektivne pravice brisanja metapodatkov, po 1.4.2015 pa jo pridobi, saj se prepoved ne upošteva več pri računanju efektivnih pravic, pravica brisanja pa se prenese iz entitete.

Primer 10: Iz list dostopnih pravic (ACL) vseh entitet v hierarhiji odstranimo vse eksplicitne prepovedi in dovoljenja, ter nastavimo naslednje ACL vrednosti za uporabnika:

- *v ACL razreda nastavimo eksplicitno dovoljenje branja entitete, ki je časovno omejeno od 1.4.2015 do 15.4.2015;*
- *v ACL zadeve nastavimo eksplicitno dovoljenje pisanja in ustvarjanja podrejenih entitet, ki je časovno omejeno od 6.4.2015 do 12.4.2015;*
- *v ACL dokumenta »Dokument 1« nastavimo eksplicitno prepoved pisanja, ki je časovno omejeno od 8.4.2015 do 12.4.2015.*

Glede na nastavljene vrednosti ACL ima uporabnik naslednje časovno omejene pravice:

- *med 1.4.2015 in 15.4.2015 ima uporabnik pravico odpiranja vseh entitet v hierarhiji;*
- *med 6.4.2015 in 12.4.2015 ima uporabnik pravico urejanja zadeve in dokumenta »Dokument 2« ter ustvarjanja podrejenih entitet v zadevi;*
- *urejanje dokumenta »Dokument 1« je dovoljeno uporabniku med 6.4.2015 in 8.4.2015, saj z 8.4.2015 začne veljati eksplicitna prepoved pisanja dokumenta, ki ima prednost pri računanju efektivnih pravic in posledično izniči efektivno pravico pisanja;*
- *pred 1.4.2015 in po 15.4.2015 uporabnik nima pravice odpiranja in urejanja entitet v hierarhiji. Še vedno pa ima pravico branja javnih metapodatkov, če mu globalna varnostna nastavitev to dopušča.*

3.3.6 Identifikatorji

Strežnik IMiS®/ARChive Server pozna tri načine identificiranja posameznih entitet.
V vseh načinih identificiranja identifikator določa natančno eno entiteto, oziroma nobene entitete v primeru arhivu neznanega identifikatorja. Identifikator nikoli ne more določati več entitet.

3.3.6.1 Notranji identifikatorji

Notranji identifikator entitete je samodejno generiran niz dolžine 24 ali 32 bajtov (dolžina je odvisna od zahteve odjemalca). Vsebina niza je odvisna tudi od identifikatorja arhiva, ki naj bi bil unikaten za vse arhive. S tem je notranji identifikator posameznih entitet tudi globalno unikaten. Pri generiranju niza strežnik IMiS®/ARChive Server uporablja šifriranje po standardu AES-256, kar zagotavlja zanemarljivo majhno verjetnost zadetka pravilnega identifikatorja v primeru naključnega ugibanja identifikatorjev.
Pri arhivu, ki vsebuje na primer 100.000.000 entitet, je verjetnost, da bo naključno generiran identifikator pravilen, samo $1 : 1{,}593 \times 10^{50}$ pri identifikatorju dolžine 24 bajtov in $1 : 8{,}636 \times 10^{70}$ pri identifikatorjih dolžine 32 bajtov.

Za shranjevanje identifikatorjev, kjer binarni način shranjevanja ni mogoč, strežnik omogoča tri načine kodiranja notranjih identifikatorjev:

- Šestnajstiško kodiranje, kjer je identifikator predstavljen z nizom znakov iz nabora 0-9, 'a'-'f'. Tak niz je dolg 48 znakov za identifikator velikosti 24 bajtov in 64 znakov za identifikator velikosti 32 bajtov.
- Base64 kodiranje, kjer je identifikator predstavljen z nizom znakov iz nabora velikih in malih črk angleške abecede (52 znakov), števil (0-9) in znakov '-' ter '_'.
  Tak niz je dolg 32 znakov za identifikatorje velikosti 24 bajtov in 43 znakov za identifikator velikosti 32 bajtov.
- Base85 kodiranje, kjer je identifikator predstavljen z nizom znakov iz nabora base64 in dodatnih 21 znakov: !#$%&()*+;<=>?@^`{|}~.
  Tako predstavljen notranji identifikator ima dolžino 30 znakov za identifikatorje velikosti 24 bajtov in dolžino 40 znakov za identifikatorje velikosti 32 bajtov.

Način kodiranja notranjih identifikatorjev v svojih zahtevkih določa odjemalec glede na njegove zahteve podatkovnega modela baze, kamor jih bo shranjeval.

3.3.6.2 Zunanji identifikatorji

Zunanji identifikatorji so nizi poljubnih znakov dolžine največ 100 znakov, ki nastajajo neodvisno od strežniškega okolja IMiS®/ARChive Server. Strežnik omogoča asociacijo katerekoli entitete s poljubnim številom zunanjih identifikatorjev, pod pogojem, da v istem arhivu še nobena entiteta ni asociirana z enakim identifikatorjem. Asociacija entitete z zunanjimi identifikatorju ni obvezna.

3.3.6.3 Klasifikacijska oznaka

Zaradi možnosti hitrejšega upravljanja z entitetami v načrtu razvrščanja gradiva in večje preglednosti, označujemo entitete (razredi, zadeve, dokumenti) v načrtu razvrščanja gradiv. Strežnik IMiS®/ARChive Server samodejno dodeljuje klasifikacijske oznake entitetam v načrtu razvrščanja gradiva glede na položaj razredov in zadev v hierarhiji. Klasifikacijske oznake so enolično določene in se dodeljujejo ob namestitvi ali kasneje pri upravljanju z načrtom razvrščanja gradiva.
V primeru spremembe položaja razreda v načrtu razvrščanja gradiva se vsem entitetam uvrščenim pod ta razred določi nova klasifikacijska oznaka, ki odraža nov položaj v hierarhiji. Nova klasifikacijska oznaka postane nemudoma veljavna za vse entitete uvrščene v ta razred.

Kadar novonastali entiteti strežnik zaradi pomanjkljive ali namerno določene konfiguracije ne more samodejno dodeliti klasifikacijske oznake, mora klasifikacijsko oznako ob nastajanju nove entitete določiti uporabnik.
Tudi ta klasifikacijska oznaka mora biti znotraj nadrejene entitete enolično določena, saj v nasprotnem primeru strežnik zavrne zahtevek za nastanek nove entitete z napako.
Do zavrnitve zahtevka pride tudi v primeru, ko klasifikacijska oznaka ni določena s strani uporabnika.
V primeru premeščanja gradiva znotraj arhiva (reklasifikacija), se morajo vsem premaknjenim entitetam spremeniti klasifikacijske oznake, da ustrezajo novemu položaju znotraj arhiva. V tem primeru ročno dodeljevanje klasifikacijskih oznak ni mogoče in strežnik z napako zavrne zahtevek za premik, če za kakšno od premaknjenih entitet ni možno samodejno dodeliti klasifikacijske oznake.

3.3.6.3.1 Kanonična oblika (angl. Fully Qualified Classification Code – FQCC)

Primer: C=01^C=02^F=2014-01^D=0001

Kanonična oblika polne klasifikacijske oznake se uporablja predvsem pri komunikaciji med strežnikom IMiS®/ARChive Server in odjemalci, uporabniki pa take oblike ne vidijo pogosto. Kanonična oblika polne klasifikacijske oznake je sestavljena iz komponent, kjer posamezna komponenta predstavlja lastno/delno klasifikacijsko oznako entitete iz hierarhije, ki ji izbrana entiteta pripada.
Komponente so med seboj vedno ločene z znakom »^«. Posamezna komponenta je sestavljena iz dveh delov. Prvi del je enoznakovna oznaka za tip entitete: C za razred, F za zadevo in D za dokument. Drugi del komponente je dejanska vrednost klasifikacijske oznake, dela pa sta ločena z znakom '='.

3.3.6.3.2 Javna klasifikacijska oznaka (angl. Public Classification Code – PCC)

Primer: 01.02-2014-01/0001 (ekvivalent primeru kanonične oblike zgoraj)

Javna klasifikacijska oznaka je oblika, ki uporabnikom predstavlja polno klasifikacijsko oznako. Sestavljena je iz prav toliko komponent kot kanonična oblika, posamezna komponenta pa vsebuje le dejansko vrednost klasifikacijske oznake. Med komponentami so nizi znakov, ki jih administrator pri nastavitvi produkta določi za ločilne znake klasifikacijskih oznak in sicer za vsak tip entitet posebej.

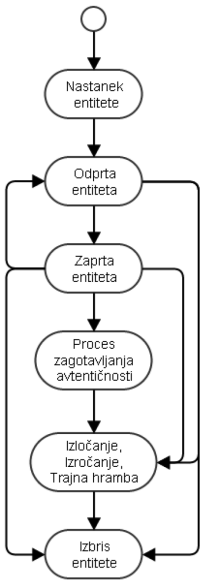
V tem primeru je pred komponentami, ki predstavljajo razrede, znak pika (.), pred komponentami, ki predstavljajo zadeve znak minus (-) in pred zadnjo komponento, ki predstavlja dokument pa je znak poševnica (/).

3.3.7 Življenjski cikel

Ločimo dva različna življenjska cikla entitet. Prvi opisuje spremembo statusa, drugi pa zgolj življenjski cikel instance entitete znotraj uporabniške seje. Vse navedeno v nadaljevanju velja za tipe entitet: razred, zadeva in dokument, razen kjer je posebej naveden tip entitete.

3.3.7.1 Status entitete

Ob nastanku dobi entiteta podedovano vrednost »Odprto« (angl. Opened; za podrobnosti glej poglavje 3.2.2 Hierarhija). Uporabnik lahko poljubno nastavlja status s spreminjanjem vrednosti atributa. Po prvem shranjevanju entitete je spreminjanje vrednosti atributa onemogočeno. Uporabnik lahko spremeni status z izvajanjem akcije za spremembo statusa.
V statusu »Odprto« (angl. Opened) uporabnik izvaja operacije nad entitetami, ki omogočajo spreminjanje njenih atributov in dodajanje podrejenih entitet (npr. vlaganje novih dokumentov v zadevo).
Uporabnik spremeni status v »Zaprto« (angl. Closed) šele, ko je povsem prepričan, da entitete ne bo več potrebno spreminjati in pod njo uvrščati novih entitet. Vsebine zaprtih entitet uporabnik ne more spreminjati. Na zaprti entiteti in podrejenih entitetah (glej poglavje 3.3.7.1.2 Status »Closed«) lahko uporabnik sproži postopek zagotavljanja avtentičnosti (glej poglavje 3.6.1 Predpogoji). Ta omogoča izdelavo in vzdrževanje dokaznih elementov avtentičnosti, ki bi v primeru kasnejših sprememb entitete v hrambi postali neveljavni.
Izbris entitete je mogoč v katerikoli fazi življenjskega cikla. Brisanje pred postopkom izločanja je izreden dogodek in naj ne bi bil del rednega delovnega procesa v organizaciji. Uporabljen naj bi bil zgolj v primeru zmotnega vnosa entitete in je namenjen popravljanju napak.



Slika 5: Življenjski cikel entitete

3.3.7.1.1 Status »Opened«

Status »Opened« povzroči, da je entiteto mogoče odpreti v načinu za pisanje, ter da je dovoljeno dodajanje novih podrejenih entitet (npr. vlaganje novih dokumentov v zadevo). Uporabnik odpre zaprto entiteto tako, da spremeni status samo dotični entiteti, status vseh podrejenih entitet pa ostane zaprt.

Odpiranje zaprte entitete je možno v primeru da:

- je entiteta, ki jo želi odpreti zaprta
- ima nadrejene entitete odprte.

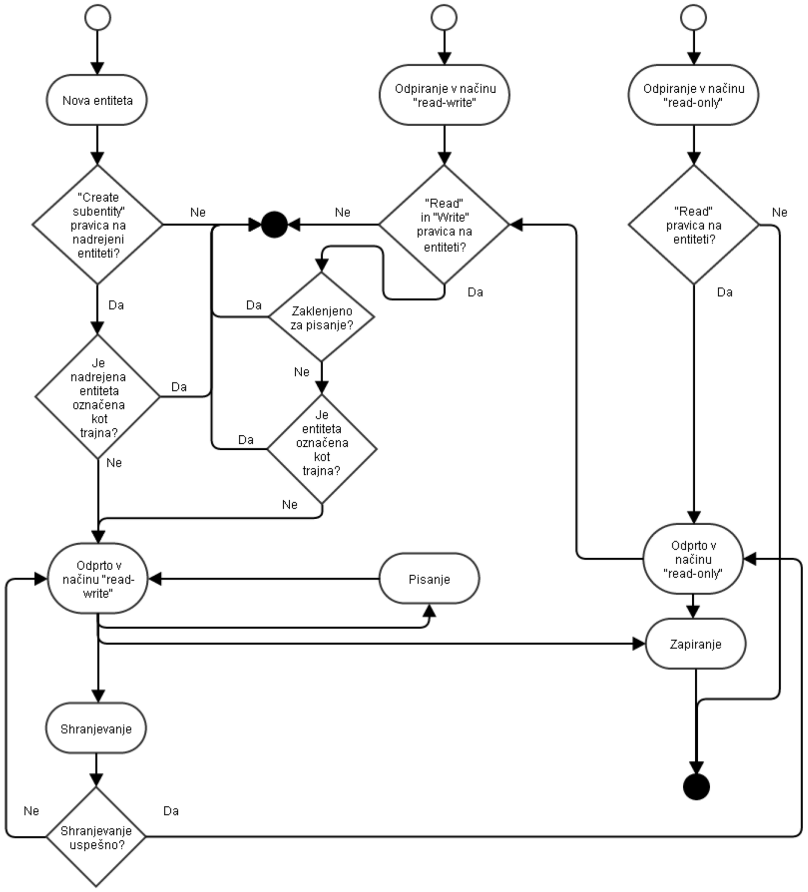
3.3.7.1.2 Status »Closed«

V statusu »Closed« je mogoče entiteto odpreti zgolj v načinu za branje. Spreminjanje atributov in dodajanje podrejenih entitet je onemogočeno. V kolikor je strežnik IMiS®/ARChive Server nastavljen za zagotavljanje avtentičnosti gradiva, strežnik v seznam entitet, primernih za generiranje dokaznih elementov avtentičnosti, takšne entitete vključi. Zapiranje odprte entitete povzroči zapiranje vseh odprtih podrejenih entitet s statusom.
V primeru, da so zaprte entitete (ter njihove vsebovane entitete) vključene v postopek zagotavljanja avtentičnosti, se za vse spremenjene entitete ponovno generira arhivski informacijski paket (AIP) ter sproži postopek zagotavljanja avtentičnosti (glej poglavje 3.6.1 Predpogoji).

3.3.7.2 Instanca entitete

Instanca entitete je njena reprezentacija v delovnem spominu strežnika IMiS®/ARChive Server.
Pri odpiranju entitete je potrebno najprej preveriti ali je entiteta že v predpomnilniku.
V predpomnilniku je vselej, kadar je uporabljena v drugi seji.
Če entitete še ni v predpomnilniku, se naloži iz podatkovne baze, drugače se uporabi že obstoječa instanca, ki je skupna vsem sejam. Ta model omogoča učinkovito izrabo virov strežnika in hitrejše odzive na zahteve po odpiranju entitet, saj je uporaba že naloženih instanc neprimerno hitrejša od prenašanja iz podatkovne baze.
V primeru dostopa »za pisanje« se v pomnilniku naredi nova kopija entitete,
ki ni dostopna nobeni drugi seji. Če ima katera druga seja že odprto svojo kopijo entitete (entiteta je odprta v načinu za pisanje v drugi seji), se dostop zavrne. Prav tako se dostop zavrne, če je entiteta označena kot trajna.

Ostale seje dostopajo do instance, iz katere je bila kopija za pisanje ustvarjena;
ta za ostale seje ostane živa in nespremenljiva. Istočasno je dovoljena zgolj ena kopija »za pisanje«.
Posledično, nobena sprememba entitete ni vidna v drugih sejah, dokler se ne izvede postopek shranitve instance, odprte za pisanje. Slednji preveri konsistentnost vnesenih metapodatkov, shrani spremembe v podatkovno bazo in zamenja instanco v predpomnilniku. Po shranitvi, vse nove operacije odpiranja entitete vračajo spremenjeno instanco. Seje,
ki imajo entiteto odprto pred shranitvijo nove instance, novega stanja v obstoječi instanci ne vidijo (model deluje po principu nespremenljivega stanja v časovni točki odpiranja).



Slika 6: Življenjski cikel instance entitete v delovnem spominu strežnika

3.3.7.3 Nova entiteta

Ne-shranjena instanca entitete nastane kot posledica zahtevka odjemalca, ki mora obvezno vsebovati:

- veljavni enolični identifikator nadrejene entitete
- veljavni enolični identifikator predloge, ki ne sme biti sistemska ali interna.

Nadalje se preveri, če so izpolnjeni naslednji pogoji:

- pravica dostopa do nadrejene entitete iz naslova stopnje tajnosti;
- uporabnik ima pravico ustvarjanja novih podrejenih entitet na nadrejeni entiteti;
- status nadrejene entitete ne sme biti »Closed«;
- nadrejena entiteta ne sme biti označena za trajno hrambo (glej poglavje 3.7 Odbiranje in izločanje);
- enolični identifikator predloge mora biti vsebovan v spisku dovoljenih entitet na tem delu načrtu razvrščanja gradiva.

Če katerikoli od zgornjih pogojev ni izpolnjen, strežnik IMiS®/ARChive Server vrne odgovor z napako. Sicer se v delovnem pomnilniku naredi nova instanca entitete, ki pa se v tej fazi še ne zapiše v podatkovno bazo. Instanca je dostopna samo v seji uporabnika, ki je objekt ustvaril.
Če naredimo poizvedbo po novi entiteti iz druge seje, le-ta ni vidna. Tip entitete določa uporabljena predloga, ki je ni mogoče spreminjati. V tej fazi klasifikacijska oznaka nove entitete ni določena. Entiteta se fizično shrani na strežniku po klicu metode za shranjevanje.
Po uspešnem klicu postane entiteta dosegljiva drugim sejam.

3.3.7.4 Odpiranje entitete za »branje in pisanje« ali »samo za branje«

Zahtevek za odpiranje entitete pri strežniku IMiS®/ARChive Server izvede odjemalec.
Ta določi ali entiteto odpira v načinu samo za branje (»RO«) ali pa v načinu za branje in pisanje – spreminjanje (»RW«).
Strežnik odpre zahtevano instanco entitete (glej poglavje 3.3.7.2 Instanca entitete) in preveri dostopne pravice. Če stopnja tajnosti entitete dovoljuje dostop in ima uporabnik potrebne dostopne pravice, mu strežnik vrne:

- referenco na odprto instanco entitete;
- identifikator njene nadrejene entitete (starša);
- predlogo, s katero je bila ustvarjena;
- več sistemskih metapodatkov o entiteti (datum ustvarjanja, zadnje spremembe, efektivne pravice uporabnika nad entiteto, ...);
- seznam predlog, s katerimi je možno ustvariti podrejene entitete.

Vsi naknadni klici se morajo sklicevati na dano referenco.
Dogodek odpiranja entitete povzroči zapis dogodka v revizijski sledi.

3.3.7.5 Branje vsebine entitete

Zahtevek za branje podatkov iz entitete je zahteva strežniku IMiS®/ARChive Server za branje komponent entitete. Entiteta mora biti odprta ali v načinu »samo za branje« ali pa v načinu »za branje in pisanje«. Zahtevek mora vsebovati vsaj:

- referenco na instanco entitete
- obseg branja podatkov.

Obseg branja podatkov vsebuje navodila strežniku, kaj naj v svojem odgovoru na zahtevo vrne. Zahteva lahko obsega zahtevo za:

- metapodatke (vsi, javni, navedeni);
- dostopne pravice (zapisi dostopnih pravic entitete, zapisi dostopnih pravic atributov ali oboji);
- dokazila o avtentičnosti in celovitosti entitete (AIP in ERS).

V odgovoru odjemalcu strežnik vrne vse vrednosti zahtevanih atributov, z izjemo vrednosti do katerih uporabnik nima pravice dostopa.
V primeru, da uporabnik nima dostopnih pravic, IMiS®/ARChive Server ne javi napake vendar mu vrednosti takih metapodatkov ne vrne.
Dogodek branja vsebine entitete ne povzroči zapisa v revizijsko sled.

3.3.7.6 Spreminjanje vsebine entitete

Zahtevek za zapis sprememb komponent instance entitete je zahteva strežniku IMiS®/ARChive Server, da si v svojo instanco entitete v spominu začasno shrani spremembe metapodatkov iz zahtevka. Odjemalec lahko pošlje več zahtevkov za spreminjanje na entiteti, ki je odprta za pisanje. Spremembe se ne shranijo v podatkovno bazo dokler strežnik ne prejme zahtevka za shranjevanje entitete.
Zahtevek vsebuje:

- referenco na instanco entitete (obvezno);
- seznam specializiranih sistemskih atributov in njihovih vrednosti;
- seznam atributov in njihovih vrednosti (vse atribute entitete razen specializiranih sistemskih atributov);
- seznam spremenjenih dostopnih pravic (dodanih, spremenjenih, izbrisanih).

V kolikor uporabnik spreminja atribut, ki vsebuje več vrednosti, je potrebno zapisati vse vrednosti, tako spremenjene kot tiste, ki ostajajo iste. Pri zapisovanju teh vrednosti se delno preveri tudi veljavnost vrednosti atributov (glej poglavje 3.2.2 Parametri atributov).
Na strežniku se lahko pošlje več zaporednih zahtevkov za spreminjanje na entiteti odprti za pisanje.
V odgovoru na zahtevek za spreminjanje strežnik odjemalcu odgovori pritrdilno ali vrne napako, da spremembe ni mogoče izvesti. Entiteta se fizično shrani na strežniku po klicu metode za shranjevanje. Po uspešnem klicu postanejo spremembe entitete dosegljive drugim sejam.

Dogodek spreminjanja entitete ne povzroči zapisa v revizijsko sled, morebitne spremembe se zabeležijo v času shranitve, saj se pri spreminjanju spreminja le instanca entitete v delovnem spominu in te spremembe niso trajne.

3.3.7.7 Shranitev

Zahtevek za shranitev je zahteva strežniku IMiS®/ARChive Server za shranjevanje entitete.
Entiteta, ki je odprta za pisanje, se fizično zapiše v podatkovno bazo,
kopija zaščitena za pisanje pa se naloži v predpomnilnik (glej poglavje 3.3.7.2 Instanca entitete).

Zahtevek mora vsebovati vsaj:

- referenco na instanco entitete.

Obenem se preveri drugi del veljavnost vrednosti atributov (glej poglavje 3.2.2 Parametri atributov). Razlog za dvofazno preverjanje je, da veljavnost vseh atributov ni mogoče preverjati pri shranjevanju. Tipični primer je preverjanje prisotnosti obveznih atributov,
ki v posamičnem klicu za spreminjanje instance ni mogoč. V postopku shranjevanja se preverijo tudi efektivni roki hrambe na entiteti, če entiteta predstavlja razred, zadevo ali dokument pod razredom.
V primeru, da entiteta nima efektivnega roka hrambe shranjevanje ni uspešno.
Po shranitvi instance entitete ostane entiteta odprta v načinu »samo za branje«, odjemalcu pa se vrne tudi unikatni notranji identifikator entitete, katerega si lahko odjemalec oziroma aplikacija, ki ga uporablja, shrani v podatkovne zbirke tretjih aplikacij za kasnejši neposredni priklic entitete. Po uspešni shranitvi so spremembe entitete vidne na vsaki instanci,
ki je bila odprta po shranjevanju. Instance odprte pred shranitvijo, ostanejo nespremenjene v njihovemu celotnemu življenjskemu ciklu.

Shranitev povzroči zapis naslednjih dogodkov v revizijsko sled:

- »Sprememba vrednosti atributa«: v kolikor pride do spremembe enega ali več sistemskih ali specializiranih atributov;
- »Sprememba liste dostopnih pravic«: v kolikor pride do spremembe enega ali več zapisov dostopnih pravic;
- »Sprememba atributa fizičnega gradiva«: v kolikor pride do spremembe enega ali več atributov upravljanja s fizičnim gradivom;
- »Entiteta shranjena«.

3.3.7.8 Zapiranje

Zahtevek za zapiranje instance entitete je zahteva strežniku IMiS®/ARChive Server, da zniža število referenc na instanco entitete za 1 in v kolikor je števec referenc padel na 0, zavrže instanco iz predpomnilnika.
Klic za zapiranje se izvede tudi za vse instance entitet neke seje ob njenem zapiranju na katerih ni bil eksplicitno klican zahtevek za zapiranje instance entitete.
Zahtevek mora vsebovati vsaj:

- referenco na instanco entitete.

Po zaprtju entitete postane referenca na instanco entitete neveljavna.
Če je bila zaprta ne-shranjena instanca entitete, odprta »za pisanje«, so spremembe nepovratno izgubljene. Zapiranje ne povzroči zapisa v revizijsko sled.

3.3.7.9 Premik

Premik entitete se sproži z zahtevkom za premik. Gre za uvrstitev entitete v drug del načrta razvrščanja gradiva, govorimo lahko tudi o reklasifikaciji.
Zahtevek mora vsebovati vsaj:

- enolično oznako entitete, ki jo želimo premakniti
- enolično oznako entitete pod katero želimo entiteto iz prejšnje točke uvrstiti
- razlog za premik.

Pri premiku strežnik IMiS®/ARChive Server preverja naslednje predpogoje:

- pravica dostopa do entitete iz naslova stopnje tajnosti;
- pravica spreminjanja entitete, ki jo premikamo (dodajanje vrednosti sistemskih atributov);
- pravica premika entitete, ki jo premikamo;
- pravica dodajanja novih podrejenih entitet pod entiteto, kamor jo uvrščamo (nova lokacija);
- tarčna entiteta, kamor uvrščamo entiteto, ki se premika, mora dovoljevati ustvarjanje podrejenih entitet s predlogo, s katero je bila izdelana entiteta, ki se premika;
- tarčna entiteta, kamor uvrščamo entiteto, ne sme biti zaprta.

Če katerikoli od zgornjih predpogojev ni izpolnjen, se operacija premika zavrne.
Skupaj z označeno entiteto se premaknejo tudi vse njene podrejene entitete.
Vsi zgoraj omenjeni predpogoji se preverjajo tudi na podrejenih entitetah.

Klasifikacijske oznake premaknjenih entitet se izračunajo na novo glede na pravila,
ki veljajo na novi lokaciji (glej poglavje 3.4.1 Klasifikacijske oznake) in nimajo povezave s starimi oznakami. Uporabnik, ki je sprožil premik, mora obvezno navesti tudi razlog.
Dogodek premika se zabeleži v revizijski sledi premaknjene entitete, hkrati pa pride do zapisa dogodka premika na vseh podrejenih entitetah, ki se z njo premikajo.
Obseg podatkov v vsebini sporočila v revizijski sledi je različna za entiteto, ki se premika in njej podrejene entitete.

3.3.7.10 Sprememba stopnje tajnosti

Spremembo stopnje tajnosti odjemalec sproži z zahtevkom za spremembo stopnje tajnosti.
Zahtevek mora vsebovati vsaj:

- enolično oznako entitete
- novo stopnjo tajnosti
- razlog za spremembo.

Pred spremembo stopnje tajnosti strežnik IMiS®/ARChive Server preverja naslednje predpogoje:

- pravica dostopa do entitete iz naslova stopnje tajnosti;
- pravica uporabnika do spremembe stopnje tajnosti na dotični entiteti;
- pravica do uporabe stopnje tajnosti, ki ustreza uporabnikovi, torej je manjša ali enaka uporabnikovi;
- entiteta, ki ji spreminjamo stopnjo tajnosti, ne sme biti zaprta.

Strežnik najprej preveri ali je operacije spremembe stopnje tajnosti izvedljiva. Stopnja tajnosti ne more biti višja od podedovane s strani nadrejene entitete. V primeru dviga stopnje tajnosti, dvigne stopnjo tajnosti entiteti in njej podrejenim entitetam z enako stopnjo tajnosti, v kolikor imajo podrejene entitete nižjo stopnjo tajnosti, jih pusti nedotaknjene.
V primeru nižanja stopnje tajnosti spremeni stopnjo tajnosti vsem podrejenim entitetam, ki eksplicitno določajo stopnjo tajnosti, na novo raven.

Strežnik ob spremembi stopnje tajnosti samodejno dopolni atribute sprememb stopenj tajnosti na entiteti, kateri neposredno določamo novo stopnjo tajnosti:

- »sys:SecurityClassChangeReason«: razlog za spremembo stopnje tajnosti posreduje uporabnik kot obvezen podatek ob dejanju spremembe stopnje tajnosti;
- »sys:SecurityClassChangeAgent«: uporabnik, ki je spremembo stopnje tajnosti izvedel;
- »sys:SecurityClassChangeDateTime«: datum in čas spremembe stopnje tajnosti;
- »sys:SecurityClassChangeFrom«: vrednost efektivne stopnje tajnosti entitete pred spremembo, lahko je eksplicitna ali podedovana;
- »sys:SecurityClassChangeTo«: vrednost efektivne stopnje tajnosti entitete po spremembi, lahko je eksplicitna ali podedovana.

Vsaka sprememba stopnje tajnosti iz naslova neposredne spremembe ali spremembe kot posledice podrejenosti entiteti, ki se ji stopnja tajnosti spreminja, se zabeleži v revizijsko sled vsakokratne entitete kot dogodek spremembe stopnje tajnosti s prejšnjo vrednostjo, novo vrednostjo in razlogom za spremembo.

3.3.7.11 Brisanje

Izbris entitete se odjemalec sproži z zahtevkom za izbris entitete. Zahtevek mora vsebovati vsaj:

- enolično oznako entitete;
- v kolikor sistemski atribut »sys:Description« entitete nima vrednosti, mu je vrednost možno dodeliti ob izbrisu, saj je opis entitete ob izbrisu obvezen podatek;
- razlog za izbris.

Pred izbrisom strežnik IMiS®/ARChive Server preverja naslednje predpogoje:

- pravica uporabnika do izbrisa entitete;
- prisotnost vseh obveznih metapodatkov: preveri tudi sistemski atribut »sys:Description«, ki v primeru izbrisa postane obvezen;
- preveri se sistemski atribut »sys:Significance«, če je vrednost atributa nastavljena na 1 (»Vital«) ali 2 (»Permanent«), potem je brisanje entitete onemogočeno.

Način izbrisa je na strežniku mogoče nastaviti tako, da izbris:

- ohrani celovitost entitete, torej ne pride do izbrisa katere koli komponente entitete (*specifikacija Moreq2 9.3.1*);
- povzroči izbris vseh metapodatkov in vsebin razen tistih, ki so obvezni za ohranitev konsistentnosti izbrisane entitete (*specifikacija Moreq2 9.3.2*).

V obeh primerih se entiteta premakne v sistemski razred izbrisov s klasifikacijsko oznako »C=sys ^C=Trash^C=Deleted«, kjer se zbirajo vsi izbrisi.
Posledično se entiteta vsem uporabnikom »skrije«.

Vsaka izbrisana entiteta ne glede na nastavitev obsega izbrisa ohrani naslednje sistemske atribute:

- »sys:Title«: Naslov entitete.
- »sys:Description«: Opis entitete (prej neobvezen postane v izbrisanih entitetah obvezen). V primeru, da uporabnik poda opis entitete pred izbrisom, se entiteta odpre v načinu za spreminjanje, izvede se popravek atributa in shrani entiteta. Vse spremembe se zabeležijo v revizijski sledi.

Dodatno strežnik samodejno doda naslednje sistemske atribute izbrisane entitete:

- »int:Template«: izvirna predloga s katero je bila entiteta ustvarjena (uporabniku je skrita);
- »int:ParentId«: notranji identifikator nadrejene entitete, kamor je bila izbrisana entiteta uvrščena (uporabniku skrita);
- »sys:del:Reason«: razlog za izbris posreduje uporabnik kot obvezen podatek ob izbrisu;
- »sys:del:Agent«: uporabnik, ki je izbris izvedel;
- »sys:del:DateTime«: datum in čas izbrisa;
- »sys:del:ClassificationCode«: klasifikacijska oznaka entitete pred izbrisom;
- »sys:del:Reference«: referenca na preneseno entiteto – atribut je v primeru brisane entitete prazen, uporablja se samo za prenesene entitete v okviru odbiranja in izločanja (glej poglavje 3.7 Odbiranje in izločanje).

Opozorilo: Operacija izbrisa je nepovratna.

V primeru, da se opis entitete posreduje v času izbrisa, pride pred izbrisom do beleženja naslednjih dogodkov v revizijski sledi:

- odpiranje entitete v načinu za spreminjanje
- sprememba sistemskega atributa »sys:Description«
- shranitev entitete.

Dogodek izbrisa se zabeleži v revizijski sledi izbrisane entitete.

3.3.7.12 Sprememba statusa

Spremembo statusa odjemalec sproži z zahtevkom za spremembo statusa.
Zahtevek mora vsebovati:

- enolično oznako entitete
- novo vrednost statusa
- razlog za spremembo (neobvezno).

Pred spremembo stopnje tajnosti strežnik IMiS®/ARChive Server preverja naslednje predpogoje:

- pravica uporabnika do spremembe statusa na dotični entiteti in vsebovanih entitetah;
- entiteta, kateri se spreminja status mora imeti nadrejene entitete odprte.

Spremembo statusa lahko delimo v naslednje akcije:

- odpiranje zaprte entitete
- zapiranje odprte entitete
- nastavljanje podedovane vrednosti na odprti entiteti.

Odpiranje zaprte entitete: odpiranje zaprte entitete povzroči, da se odpre samo dotična entiteta, njene podrejene entitete pa ostanejo zaprte. V sistemski atribut »sys:Status« se zapiše nov status, v »sys:Opened« se zapiše trenutni datum in čas odprtja, vrednost »sys:Closed« pa se izbriše. Dogodek odprtja se zapiše v revizijsko sled odprte entitete.

Zapiranje odprte entitete: zapiranje odprte entitete povzroči, da se zapre dotična in vse podrejene odprte entitete. Vsem zaprtim entitetam se nastavi »sys:Closed« na trenutni datum in čas ter v »sys:Status« zapiše nova vrednost statusa (samo na dotični entiteti, ker podrejene entitete to vrednost podedujejo). V revizijsko sled vsake zaprte entitete se zapiše dogodek o spremembi statusa.

Nastavljanje podedovane vrednosti: v primeru, da ima odprta entiteta eksplicitno nastavljeno vrednost za status, potem ji s to akcijo nastavimo podedovano vrednost nadrejene entitete. Ker se vsebinsko vrednost statusa za dotično entiteto ni spremenila se tudi akcija ne zapiše v revizijsko sled.

3.3.8 Revizijska sled

Revizijska sled je nespremenljiv kronološki zapis dostopov, poizvedb in sprememb na strežniku IMiS®/ARChive Server. Revizijska sled vsebuje vsaj informacijo o uporabniku, času in dejanju v zvezi s katerim koli dokumentom, zbirko ali razredom načrta razvrščanja gradiva.
Hkrati je dokumentiran zapis o izvajanju določenih postopkov.
Revizijska sled je popolnoma nespremenljiva v svojem celotnem življenjskem ciklu
in iz tega vidika zaščitena pred dovoljenimi in nedovoljenimi posegi.

Omogoča beleženje sprememb in pregled nad izvajanjem postopkov.
Namenjena je ugotavljanju izvajanja aktivnosti nad arhiviranimi objekti.
Podatki revizijske sledi se shranjujejo skupaj z arhiviranim gradivom v strežniku.
Za vsako uporabnikovo sejo se v revizijski sledi zabeleži tudi začetek in konec »Revizijske seje« (angl. Audit Session). Vsi dogodki vsebujejo referenco na to sejo.

Podatki, ki se beležijo v revizijski sledi, sledijo določbam Zakona o varstvu osebnih podatkov. Namen zakona je preprečevati nezakonite in neupravičene posege v zasebnost posameznika pri obdelavi osebnih podatkov, njihovem varovanju in uporabi.

3.3.8.1 Seje

Z odprtjem seje na strežniku IMiS®/ARChive Server se obenem odpre tudi seja revizijske sledi. Pri odprtju slednje se v podatkovno bazo zapišejo naslednji podatki:

- uporabniški račun uporabnika, ki se je prijavil;
- ime (angl. Hostname) računalnika iz katerega je bila seja vzpostavljena;
- datum in čas začetka seje;
- interni omrežni naslov (notranji IP naslov mrežnega vmesnika iz katerega je bila seja vzpostavljena, posreduje odjemalec v avtentikacijskih podatkih);
- javni omrežni naslov (omrežni naslov odjemalca kot ga vidi strežnik);
- datum ter čas zaključka seje;
- razlog zaključka seje:
  - seja ni bila zaprta (0)
  - normalen zaključek seje (1)
  - nepričakovan zaključek seje s strani odjemalca (2)
  - iztek dovoljene neaktivnosti (angl. Time-out) (3)
  - nepravilna avtorizacija (4)
  - nepravilni zahtevek (5).

Po zapisu teh podatkov seji strežnik dodeli enolični identifikator, ki se ne spreminja v času veljavnosti seje in se uporablja za zapis vsakega dogodka, ki je bil zaznan in sprožen v okviru te seje; postane povezovalni podatek med podatki o seji in vseh dogodkov te seje.

3.3.8.2 Dogodki

Vsak dostop do strežnika se zabeleži kot dogodek revizijske sledi.
V vsak zapis dogodka v revizijsko sled se zabeležijo vsaj naslednji parametri:

- enolični identifikator seje
- vrsta dogodka
- datum in čas dogodka.

Revizijska sled dovoljuje shranitev naslednjih neobveznih parametrov, ki jih posreduje odjemalec ob dejanju:

- razlog za dejanje: lahko je parametriziran s »printf« oblikovanim sporočilom (http://en.wikipedia.org/wiki/Printf_format_string);
- parametri razloga za dejanje: v kolikor je razlog parametriziran, se vsebina parametrov pri izpisu dejanja združi z razlogom za dejanje v enotno sporočilo.

V revizijski sledi se samodejno beležijo naslednji dogodki:

Nova entiteta [1]

Dogodek se zapiše v revizijsko sled vsakokrat, ko uporabnik ustvari novo entiteto (glej poglavje 3.3.7.3 Nova entiteta). Ker potrebujemo enolično oznako entitete, ki pa v tej fazi še ni na voljo (dodeli se ob shranjevanju), se dogodek fizično zapiše v podatkovno bazo šele ob shranjevanju s časom shranitve, istočasno kot dogodek »Entiteta shranjena«.

Razlog za tak model beleženja tega dogodka je tudi v tem, da entiteta pred shranitvijo dejansko ne obstaja in na strežniku fizično nastane ob prvi shranitvi.
V poročilo revizijske sledi se zabeleži:

- enolična oznaka entitete
- razlog/sporočilo uporabnika ob ustvaritvi entitete (neobvezno)
- parametri razloga/sporočila (neobvezno, oziroma obvezno, če je prisoten parametriziran razlog/sporočilo).

Odpiranje entitete v načinu za branje [2]

Dogodek se zapiše v revizijsko sled vsakokrat, ko uporabnik odpre entiteto v načinu za branje (glej poglavje 3.3.7.4 Odpiranje entitete za »branje in pisanje« ali »samo za branje«).
V poročilo revizijske sledi se zabeleži:

- enolična oznaka entitete;
- razlog/sporočilo uporabnika za odpiranje entitete (neobvezno);
- parametri razloga/sporočila (neobvezno, oziroma obvezno, če je prisoten parametriziran razlog/sporočilo).

Odpiranje entitete v načinu za branje in pisanje [3]

Dogodek se zapiše v revizijsko sled vsakokrat, ko uporabnik odpre entiteto v načinu za pisanje (glej poglavje 3.3.7.4 Odpiranje entitete za »branje in pisanje« ali »samo za branje«).

V poročilo revizijske sledi se zabeleži:

- enolična oznaka entitete;
- razlog/sporočilo uporabnika za odpiranje entitete (neobvezno);
- parametri razloga/sporočila (neobvezno, oziroma obvezno, če je prisoten parametriziran razlog/sporočilo).

Entiteta shranjena [4]

Dogodek se zapiše v revizijsko sled, ko se entiteta shrani (glej poglavje 3.3.7.7 Shranitev entitete).

V poročilo revizijske sledi se zabeleži:

- enolična oznaka entitete;
- razlog/sporočilo uporabnika za shranitev entitete (neobvezno);
- parametri razloga/sporočila (neobvezno, oziroma obvezno, če je prisoten parametriziran razlog/sporočilo).

Premik entitete [5]

Dogodek se zapiše v revizijsko sled pri premiku (glej poglavje 3.3.7.9 Premik).

Za vsako premaknjeno entiteto (pri premiku veje entitet jih je lahko več) se dogodek zabeleži v njeno revizijsko sled. V poročilo revizijske sledi se zabeleži:

- status entitete;
- stara polna klasifikacijska oznaka;
- nova polna klasifikacijska oznaka;
- vrednosti vseh vsebovanih atributov;
- razlog/sporočilo uporabnika za premik entitete (neobvezno).

Brisanje entitete [6]

Dogodek se zapiše v revizijsko sled pri izbrisu entitete (glej poglavje 3.3.7.11 Brisanje).

Obvezni podatek o razlogu za izbris se poleg zabeleženja v za to namenjeni atribut zabeleži tudi v revizijski sledi.

V poročilo revizijske sledi se zabeleži:

- enolična oznaka entitete;
- razlog/sporočilo uporabnika za izbris entitete (neobvezno);
- parametri razloga/sporočila (neobvezno, oziroma obvezno, če je prisoten parametriziran razlog/sporočilo).

Poizvedba revizijske sledi [7]

Dogodek se zapiše v revizijsko sled ob vsakokratni poizvedbi v revizijsko sled.

V poročilo revizijske sledi se zabeleži:

- iskalni niz, ki ga strežnik oblikuje na podlagi uporabnikove poizvedbe.

Sprememba nastavitev revizijske sledi [8]

Dogodek se zapiše v revizijsko sled ob zaznavi strežnika, da je prišlo do spremembe nastavitev revizijske sledi. Ker se nastavitev spremembe revizijske sledi lahko izvede le ob zaustavljeni storitvi, se to preverjanje istovetnosti prejšnjih in novih nastavitev revizijske sledi zgodi ob zagonu.

V poročilo revizijske sledi je zabeleženo katera nastavitev se je spremenila in vse nastavitve. Spremenjene nastavitve so označene s »*«.

Primer: V sporočilu revizijske sledi se zabeleži:

- *»Enabled state changed to on*«;*
- *»Global required audit log parameters settings changed. UserName: on* ComputerName: on, PrivateAddress: on, Reason: on«;*
- *»Events settings changed. AuditLog.Query: on*, Entity.Create: on*, Entity.OpenReadOnly: on*, Entity.OpenReadWrite: on*, Entity.Update: on*, Entity.Move: on*, Entity.Delete: on*, Entity.ACLChange: on*, Entity.PropertiesChange: on*, Entity.PhysicalRecordsManagementChange: on*, Entity.SecurityClassChange: on*, Entity.StatusChange: on*, Entity.Disposed: on*, Entity.Permanent: on*, Entity.Transferred: on*, Entity.Reviewed: on*, Content.OpenReadOnly: on*, Content.OpenReadWrite: on*, Content.Create: on*, Content.Delete: on*, Content.Update: on*, Content.MetadataChange: on*«.*

Z beleženjem dogodka preprečujemo zlorabe, kjer bi administrator z dostopom do konfiguracije strežnika lahko začasno konfiguracijo strežnika spremenil.

Primer: izključitev revizijske sledi za čas ene poizvedbe.

Sprememba vrednosti atributa [9]

Dogodek se zapiše v revizijsko sled ob shranjevanju entitete in sicer ob spremembi vrednosti enega ali več atributov (glej poglavje 3.3.7.6 Spreminjanje vsebine entitete in poglavje 3.3.7.7 Shranitev entitete). Atributi upravljanja s fizičnim gradivom so zavedeni v svojem dogodku in ta dogodek ne beleži.

V poročilo revizijske sledi se zabeleži:

- enolična oznaka entitete
- seznam vseh spremenjenih atributov.

Sprememba liste dostopnih pravic [10]

Dogodek se zapiše v revizijsko sled ob shranjevanje entitete in sicer ob spremembi liste dostopnih pravic(glej poglavje 3.3.7.6 Spreminjanje vsebine entitete in poglavje 3.3.7.7 Shranitev entitete). Takrat strežnik preveri in primerja obstoječo in novo listo dostopnih pravic ter ugotovi razlike.

V kolikor je prišlo do spremembe, se v revizijsko sled zabeležita obe listi: stara in nova.
Če je bil dodan kakšen nov vpis v listi se zapiše samo ta, saj stare vrednosti ni.
V primeru izbrisa zapisa iz liste dostopnih pravic se zapiše zgolj stara, saj nove vrednosti ni.
Enolična oznaka atributa se zapiše kadar se spremeni lista dostopnih pravic za atribut.

V poročilo revizijske sledi se zabeleži:

- enolična oznaka entitete imenika – uporabnik ali skupina;
- staro stanje zapisa v listi dostopnih pravic;
- nova stanje zapisa v listi dostopnih pravic;
- enolična oznaka atributa – v kolikor gre za zapis v listi dostopnih pravic za atribut.

Sprememba atributov upravljanja s fizičnim gradivom [11]

Dogodek se zapiše v revizijsko sled ob shranjevanju entitete (glej poglavje 3.3.7.6 Spreminjanje vsebine entitete in poglavje 3.3.7.7 Shranitev entitete). Zapiše se samo v primeru, ko je bil spremenjen vsaj en atribut upravljanja s fizičnim gradivom.

V poročilo revizijske sledi se zabeleži:

- enolična oznaka entitete;
- vsi spremenjeni in nespremenjeni atributi upravljanja s fizičnim gradivom in njihove vrednosti pred in po spremembi.

Sprememba stopnje tajnosti [12]

Dogodek se zapiše v revizijsko sled ob spremembi stopnje tajnosti (glej poglavje 3.3.5.1 Stopnje tajnosti).

V poročilo revizijske sledi se zabeleži:

- stara stopnja tajnosti
- nova stopnja tajnosti
- komentar.

Sprememba statusa entitete [13]

Dogodek se zapiše v revizijsko sled ob spremembi statusa entitete.

V poročilo revizijske sledi se zabeleži:

- nova vrednost statusa
- razlog (neobvezno).

Sprememba politik hrambe in zadržanj v postopku odbiranja in izločanja [14]

Dogodek se zapiše v revizijsko sled ob spremembi efektivnih politik hrambe ter dodanih ali odvzetih zadržanj entitet v postopku odbiranja in izločanja.

V poročilo revizijske sledi se zapišejo nazivi:

- dodanih politik hrambe
- odvzetih politik hrambe
- dodanih zadržanj
- odvzetih zadržanj.

Odpiranje vsebin v načinu za branje [17]

Dogodek se zapiše v revizijsko sled ob odpiranju vsebine v načinu za branje (za podrobnosti o vsebinah glej poglavje 3.2.5 Sistemski atributi - sys:Content).

V poročilo revizijske sledi se zabeleži:

- identifikator vsebine
- opis vsebine (če je prisoten)
- razlog (neobvezno).

Odpiranje vsebin v načinu za urejanje [18]

Dogodek se zapiše v revizijsko sled ob odpiranju vsebine v načinu za urejanje.

V poročilo revizijske sledi se zabeleži:

- identifikator vsebine
- opis vsebine (če je prisoten)
- razlog (neobvezno).

Ustvarjanje vsebine [19]

Dogodek se zapiše v revizijsko sled ob ustvarjanju vsebine.

V poročilo revizijske sledi se zabeleži:

- identifikator vsebine
- opis vsebine (če je prisoten)
- razlog (neobvezno).

Brisanje vsebine [20]

Dogodek se zapiše v revizijsko sled ob brisanju vsebine.

V poročilo revizijske sledi se beleži:

- identifikator vsebine
- opis vsebine (če je prisoten ob izbrisu)
- razlog (neobvezno).

Shranjevanje vsebine [21]

Dogodek se zapiše v revizijsko sled ob shranjevanju spremenjene vsebine.

V poročilo revizijske sledi se zabeleži:

- identifikator vsebine
- opis vsebine (če je prisoten)
- razlog (neobvezno).

Sprememba metapodatkov vsebine [22]

Dogodek se zapiše v revizijsko sled ob spremembi metapodatkov vsebine (sprememba opisa).

V poročilo revizijske sledi se zabeleži:

- identifikator vsebine
- nov opis vsebine (če je prisoten)
- razlog (neobvezno).

Izločanje entitete [23]

Dogodek se zapiše v revizijsko sled ob izločanju entitete v postopku odbiranja in izločanja.

V poročilo revizijske sledi se zabeleži:

- razlog izločanja
- komentar (neobvezno).

Trajna hramba entitete [24]

Dogodek se zapiše v revizijsko sled ob označevanju entitete za trajno v postopku odbiranja in izločanja.

V poročilo revizijske sledi se zabeleži:

- razlog za trajno hrambo
- komentar (neobvezno).

Izročanje entitete [25]

Dogodek se zapiše v revizijsko sled ob izročanju entitete v postopku odbiranja in izločanja.

V poročilo revizijske sledi se zabeleži:

- razlog izročanja
- komentar (neobvezno).

Izpuščeno za naslednji pregled [26]

Dogodek se zapiše v revizijsko sled ob izpuščanju entitete v postopku odbiranja in izločanja.

V poročilo revizijske sledi se zabeleži:

- razlog izpustitve
- komentar (neobvezno).

3.3.8.3 Pravica vpogleda v revizijsko sled

Uporabnik lahko izvaja poizvedbe revizijske sledi v kolikor ima pravico vpogleda v revizijsko sled. Uporabnik ima lahko vpogled v celotno revizijsko sled ali pa ga nima. Pravico pridobi z dodelitvijo vloge »RevizijskaSled« (angl. AuditLogQuery). Rezultat vpogleda je poročilo v XML datoteki, ki je opisano v poglavju 3.3.8.5 Format poročila.

3.3.8.4 Poizvedba

Uporabnik lahko pri poizvedbi uporabi naslednje iskalne parametre in tako omeji število rezultatov na dogodke, ki ga dejansko zanimajo:

- razpon datumov dogodkov
- razpon omrežnih (IP) naslovov (npr. od 192.168.1.1 do 192.168.1.99)
- spisek omrežnih (IP) naslovov
- spisek uporabniških imen
- spisek imen računalnikov, ki so bili uporabljani za dostop do sistema
- seznam šifriranih enoličnih identifikatorjev entitete.

V primeru, da so iskalni parametri premalo selektivni in bi bilo prikazanih preveč rezultatov, strežnik v odgovoru vrne napako. V tem primeru je priporočljivo ponoviti poizvedbo z bolj natančno določenimi parametri iskanja. Uporabnik najlažje omeji razpon rezultatov s čim manjšim časovnim obdobjem dogodkov.
Iskalne parametre lahko med seboj tudi kombinira, vendar obstajajo omejitve.
Ne glede na ostale parametre lahko vedno išče po datumu poizvedbe in po seznamu šifriranih enoličnih identifikatorjev entitete.

Dodatno lahko izbere en parameter iz naslednjega seznama:

- razpon omrežnih (IP) naslovov
- seznam omrežnih (IP) naslovov
- seznam uporabniških imen
- spisek imen računalnikov, iz katerih se je izvajal dostop do sistema.

Rezultat poizvedbe je poročilo v XML datoteki, ki je opisano v poglavju 3.3.8.5 Format poročila v nadaljevanju.

3.3.8.5 Format poročila

Poročilo, izdelano na podlagi podatkov iz revizijske sledi, je ključno pri rekonstrukciji dogodkov, ki spremljajo neko arhivirano gradivo v celotnem življenjskem ciklu – od nastanka do izvedbe revizijskega pregleda. Izostanek revizijske sledi praktično onemogoči verodostojen revizijski pregled pooblaščene osebe, ki lahko temelji na podlagi subjektivnih ocen okoliščin in dogodkov. Kot rezultat vpogleda v revizijsko sled posreduje odjemalcu, ki je vpogled opravil, XML datoteko. Ta je sestavljena po določilih XSD sheme priložene strežniku IMiS®/ARChive Server. XSD shema je dostopna tudi na spletnem naslovu: http://www.imis.si/imisarc/auditlog.xsd.

Primer: XML zapis vsebuje rezultat vpogleda v revizijsko sled

```
<?xmlversion="1.0" encoding="UTF-8"?>
<auditlog.query.resultsetxsi:schemaLocation="http://www.imis.si/imisarc
http://www.imis.si/imisarc/auditlog.xsd" xmlns="http://www.imis.si/imisarc"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<sessions>
<!--Audit query sessions.-->
<sessionid="3" closureReason="2" address="192.168.92.77"
internal_address="192.168.92.77" dateTimeOpened="2014-04-23T14:51:38Z"
dateTimeClosed="2014-04-23T14:54:00Z"username="jnovak"
computerName="novak-pc"/>
<sessionid="4"closureReason="2" address="192.168.92.23"
internal_address="192.168.92.23" dateTimeOpened="2014-04-23T14:54:25Z"
dateTimeClosed="2014-04-23T15:33:10Z" username="fkovac"
computerName="kovac-pc"/>
<sessionid="23" closureReason="0" address="192.168.92.77"
internal_address="192.168.92.77" dateTimeOpened="2014-04-24T10:01:27Z"
username="jnovak" computerName="novak-pc"/>
</sessions>
<events>
<!--Audit query events.-->
<!--Sort compare function is QuerySorter::CompareSESS_TS-->
<eventseq="0" sessionId="23" type="2" dateTime="2014-04-24T10:01:47Z" classificationCode="C=01"
context="0D=R}rSc^&amp;lt;N3ZQRdCGJqtf&amp;amp;z2iB`~aWgCI+MeIYx"/>
<eventseq="1" sessionId="3" type="1" dateTime="2014-04-23T14:51:43Z" classificationCode="C=01"
context="0D=R}rSc^&amp;lt;N3ZQRdCGJqtf&amp;amp;z2iB`~aWgCI+MeIYx"/>
<eventseq="2" sessionId="3" type="4" dateTime="2014-04-23T14:51:43Z" classificationCode="C=01"
context="0D=R}rSc^&amp;lt;N3ZQRdCGJqtf&amp;amp;z2iB`~aWgCI+MeIYx"/>
<eventseq="3" sessionId="3" type="10" dateTime="2014-04-23T14:51:43Z" classificationCode="C=01"
context="0D=R}rSc^&amp;lt;N3ZQRdCGJqtf&amp;amp;z2iB`~aWgCI+MeIYx" message="Values for the following
properties have changed:sys:Creator, sys:Opened, sys:Owner, sys:Status, sys:Title"/>
<eventseq="4" sessionId="4" type="2" dateTime="2014-04-23T14:54:26Z" classificationCode="C=01"
context="0D=R}rSc^&amp;lt;N3ZQRdCGJqtf&amp;amp;z2iB`~aWgCI+MeIYx"/>
</events>
</auditlog.query.resultset>
```

XML datoteko sestavljata dva (2) sklopa:

- »sessions«
- »events«.

3.3.8.5.1 Podatki o sejah: sklop »Sessions«

Skop »sessions« vsebuje seznam vseh sej, znotraj katerih so bili sproženi iskani dogodki.

Seznam podatkov seje:

- enolični identifikator seje (Id);
- razlog zaključka seje (closureReason), seznam možnih razlogov zaključka sej je na voljo v pripadajoči shemi na spletnem naslovu: http://www.imis.si/imisarc/auditlog.xsd;
- javni omrežni (IP) naslov (address);
- privatni omrežni (IP) naslov (internal_address);
- ime računalnika (computerName);
- čas vzpostavitve seje (dateTimeOpened);
- čas zaključka seje (dateTimeClosed);
- enolični identifikator uporabniškega računa - uporabnik (username).

V primeru, da seja še ni zaključena, je vrednost razloga zaključka seje enaka nič,
čas zaključka seje pa je nedoločen. To se zgodi v primeru, če je seja še aktivna,
ali pa (teoretičen primer), če je med sejo prišlo do nenadzorovane zaustavitve strežnika.

3.3.8.5.2 Podatki o dogodkih: sklop »Events«

Skop »events« vsebuje seznam iskanih dogodkov in njihovih podatkov.

Seznam podatkov dogodka:

- sekvenca dogodka v seznamu (seq);
- identifikator seje (sessionId): referenca na sejo, v okviru katere se je dogodek sprožil;
- vrsta dogodka (type);
- čas dogodka (dateTime);
- klasifikacijska oznaka pripadajočega objekta (classificationCode);
- šifrirani enolični identifikator entitete (context);
- poročilo/razlog za dejanje (message).

3.3.9 Roki hrambe

Roki hrambe predstavljajo časovni okvir v katerem mora arhivski sistem hraniti entitete. Po pretečenem časovnem okviru se v postopku odbiranja in izločanja odloča, kaj se bo s hranjenimi entitetami zgodilo.

Vsak rok hrambe vsebuje poleg ostalih obveznih atributov tudi naslednja atributa, ki sta ključna za njeno delovanje v pripravi postopka odbiranja in izločanja:

- atribut »sys:ret:pol:Trigger«: predstavlja iskalni niz s katerim določimo časovni okvir roka hrambe (glej poglavje 3.5.2 Pravila iskalnega niza);
- atribut »sys:ret:pol:Action«: predstavlja privzeto akcijo za entitete, ki bodo postale predmet odbiranja in izločanja po preteku roka hrambe.

Vsaka entiteta v načrtu razvrščanja gradiva (razen dokumenta v zadevi) mora imeti določen vsaj en rok hrambe. Z vzpostavitvijo povezav med entitetami in politikami hrambe nadziramo efektivne roke hrambe ter s tem posredno postopek priprave za odbiranje in izločanje. Posebno vlogo pri postopu priprave in izvedbe odbiranja in izločanja ima t.i. zadržanje. Zadržanje omogoča, da se iz postopka priprave za odbiranje in izločanje izločijo vse entitete, ki imajo povezavo na vsaj eno zadržanje. Pri postopku izvedbe se za takšne entitete izbrana akcija ne izvede. Zadržanja se tako kot politike hrambe upravlja preko povezav, opisanih v nadaljevanju.

3.3.9.1 Upravljanje s povezavami zadržanj

Zadržanje povežemo na entitete kadar jih želimo zadržati iz postopka odbiranja in izločanja. Zadržanje lahko povežemo na vse vrste entitet, na katere lahko povežemo tudi politike hrambe. Zadržanje velja za entiteto na katero je povezano in za vse vsebovane entitete, ne glede na efektivne roke hrambe. Zadržanje velja do preklica, oziroma dokler povezava ni odstranjena.

Primer: Za primer vzemimo zadevo z vsebovano zadevo »F=2015-00011^F=00001«, ki je v postopku odločanja. Zaradi izjemnega dogodka (npr. tožbe na sodišču) je zadeva »F=2015-00011« ključen material, ki se v času trajanja postopka ne sme uničiti. V tem primeru vežemo zadržanje na zadevo »F=2015-00011«. S tem poskrbimo, da bo zadeva z vsebovano zadevo izločena iz postopka izvedbe odbiranja in izločanja. Prav tako zadeva z vsebovano zadevo ne bo predmet postopka priprave za odbiranje in izločanje, dokler bo povezava na zadržanje obstajala.

3.3.9.2 Upravljanje s povezavami politik hrambe

Pri povezavah politik hrambe na entitete lahko določimo naslednje parametre:

- ali je politika omogočena ali onemogočena
- na katere vrste entitet vpliva politika hrambe (razred, zadeva, dokument uvrščen neposredno pod razred).

S temi parametri nadziramo efektivnost politik hrambe za posamezne entitete.
Efektivne politike hrambe so skupek eksplicitnih in podedovanih politik hrambe, ki nam povedo, katere politike imajo efektiven vpliv na postopek odbiranja in izločanja posameznih entitet.

Vrstni red upoštevanja povezav politik hrambe pri izračunu efektivnosti je naslednji:

1. eksplicitne povezave
2. podedovane povezave.

Efektivni roki hrambe za določeno entiteto se izračunajo na naslednji način:

- Iz celotne veje entitet (vseh nadrejenih entitet) se pregledajo povezave na politike hrambe. Ob upoštevanju hierarhije se preveri, katere politike hrambe vplivajo na trenutno vrsto entitete, za katero se izračunavajo. Rezultat so podedovani roki hrambe.
- Za dotično entiteto se pregledajo eksplicitne povezave na politike hrambe, ki se združijo s podedovanimi politikami hrambe. Rezultat so efektivni roki hrambe, ki veljajo za to entiteto.

Primeri:

Za primer vzemimo konfiguracijo politik hrambe, ki jo prikazuje spodnja tabela.

Politika hrambe	Časovni okvir
5 let	5 let od datuma ustvarjanja entitete
10 let	10 let od datuma ustvarjanja entitete

Tabela 6: Primer konfiguracije politik hrambe

Imamo naslednje drevo entitet v načrtu razvrščanja gradiva:
C=3300
C=3300^C=3301
C=3300^C=3301^F=2015-00100
C=3300^C=3301^F=2015-00101
C=3300^C=3301^F=2015-00102
C=3300^C=3302
C=3300^C=3302^D=1000
C=3300^C=3302^D=1001

Glavni razred »C=3300« vsebuje razreda »C=3301« in »C=3302«. Vsebovani razred »C=3301« vsebuje zadeve »F=2015-00100«, »F=2015-00101« in »F=2015-00102«. V vsebovanem razredu »C=3302« se nahajata dokumenta »D=1000« in »D=1001«.

Primer 1: Na glavnem razredu »C=3300« imamo vezano politiko hrambe »5 let«, ki je omogočena za razrede, zadeve in dokumente uvrščene neposredno pod razrede. S to konfiguracijo pokrijemo vse entitete v drevesu, saj se politika hrambe deduje na vse vrste entitet v drevesu.

Primer 2: Konfiguracijo iz prejšnjega primera spremenimo tako, da politiko hrambe »5 let« omogočimo za razrede in zadeve. Taka konfiguracija je neveljavna, saj se politika hrambe deduje na vse vrste entitet v drevesu razen za dokumente pod razredom, ki pa nimajo eksplicitnih povezav na politike hrambe.

Primer 3: Na vsebovani razred »C=3302« dodamo eksplicitno povezavo za politiko hrambe »10 let«, in jo omogočimo za dokumente, hkrati pa na razredu »C=3300« omogočimo politiko hrambe »5 let« za razrede in zadeve. Taka konfiguracija je ustrezna, saj velja politika hrambe »5 let« za vse razrede in zadeve, politika hrambe »10 let« pa velja samo za dokumente v vsebovanem razredu »C=3302«.

Primer 4: Na razred »C=3300« povežemo politiko hrambe »10 let« ter jo omogočimo za razrede in dokumente. Na vsebovan razred »C=3301« povežemo politiko hrambe »5 let«, in jo omogočimo za zadeve. Takšna konfiguracija določa, da se bodo zadeve odbirale po politiki hrambe »5 let«, razredi in vsebovani razredi z dokumenti pa po politiki hrambe »10 let«.

Primer 5: Na razred »C=3300« povežemo politiko »10 let« in jo omogočimo za razrede in dokumente. Na isti razred povežemo še politiko »5 let« in jo omogočimo za zadeve. Taka konfiguracija nam omogoča odbiranje vseh zadev znotraj razreda po politiki hrambe »5 let«, odbiranje razredov in dokumentov pod njimi pa po politiki hrambe »10 let«.

Primer 6: Na razred »C=3300« povežemo politiko hrambe »10 let«, in jo omogočimo za razrede, zadeve in dokumente. Na vsebovani razred »C=3301« povežemo politiko hrambe »10 let« in jo onemogočimo za razrede in zadeve, hkrati pa povežemo politiko hrambe »5 let« in jo omogočimo za razrede in zadeve. S takšno konfiguracijo lahko odbiramo razrede, zadeve in dokumente pod razredi s politiko hrambe »10 let« razen vsebovanega razreda »C=3301«, ki ima politiko hrambe »10 let« onemogočeno, ima pa omogočeno politiko hrambe »5 let«, ki je omogočena za razrede in zadeve.

Primer 7: Na razred »C=3300« vežemo politiko hrambe »10 let« in jo omogočimo za razrede, zadeve in dokumente. Nato na vsebovani razred »C=3301« vežemo politiko hrambe »5 let« in jo omogočimo za zadeve. Konfiguracija je sicer veljavna, saj imajo vse vrste entitet v drevesu efektivno politiko hrambe. Vendar pa so zadeve v vsebovanem razredu »C=3301« v konfliktu, saj imajo podedovani obe politiki hrambe (politika »10 let« se deduje od glavnega razreda »C=3300«, politika »5 let« pa se deduje od vsebovanega razreda »C=3301«). V primeru, da se bo pripravljalo odbiranje in izločanje po eni ali obeh politikah hrambe, bodo zadeve zmeraj prisotne zaradi konflikta.

Primer 8: Na razred »C=3300« vežemo politiko hrambe »10 let«, in jo omogočimo za razrede, zadeve in dokumente. Na zadevo »F=2015-00100« povežemo politiko »10 let« in jo onemogočimo za zadeve, hkrati pa na isto zadevo povežemo politiko hrambe »5 let« in jo omogočimo za zadeve. Enako naredimo še za zadevi »F=2015-00101« in »F=2015-00102«. S tako konfiguracijo lahko odbiramo zadeve po politiki hrambe »5 let«, razrede in dokumente pod razredi pa po politiki hrambe »10 let«.

Primer 9: Na razred »C=3300« vežemo politiko hrambe »10 let« in jo omogočimo za razrede, zadeve in dokumente. Na vsebovani razred »C=3302« vežemo politiko hrambe »10 let« in jo onemogočimo za dokumente. Konfiguracija je neveljavna, saj dokumenti v vsebovanem razredu nimajo efektivnih politik hrambe. Če pa na oba dokumenta v vsebovanem razredu vežemo recimo politiko hrambe »5 let« in jo omogočimo za dokumente, potem lahko na vsebovanem razredu »C=3302« onemogočimo politiko hrambe »10 let« za dokumente, saj imajo vsi dokumenti v vsebovanem razredu vsaj eno efektivno politiko hrambe.

Primer 10: V drevesu entitet imamo naslednjo konfiguracijo: na razred »C=3300« je vezana politika »10 let«, ki je omogočena za razrede. Na vsebovani razred »C=3301« je vezana politika »5 let«, ki velja za zadeve, na vsebovani razred »C=3302« je tudi vezana politika »5 let«, ki pa velja za dokumente. Želeli bi, da bi politika »10 let« veljala za vse entitete v drevesu entitet. Najlažji način je, da najprej omogočimo na razredu »C=3300« politiko »10 let« za razrede, zadeve in dokumente. S to potezo sicer naredimo konflikt za zadeve in dokumente, vendar pa, če odstranimo povezavi na politike na vsebovanih razredih »C=3301« in »C=3302«, rešimo nastali konflikt, hkrati pa nam konfiguracija politike na razredu »C=3300« pokrije celotno drevo entitet s politiko hrambe »10 let«.

3.3.10 Upravljanje z vsebinami

Vsebina je poleg metapodatkov pomemben del celote entitete. Poleg ustvarjanja, spreminjanja in brisanja vsebine lahko z vsebino upravljamo tudi na naslednje načine:

- samodejno pretvarjanje vsebine;
- izvajanje drugih akcij nad vsebino.

3.3.10.1 Samodejno pretvarjanje vsebin

Samodejno pretvarjanje vsebin je postopek, kjer se za določen tip vsebine ustvari več reprezentacij iste vsebine, vendar v različnih formatih. Tako lahko pretvarjamo BMP slike v TIFF format, ali pa MS Word dokumente v PDF/A. Vse samodejno ustvarjene vsebine so odvisne od originalne vsebine, zato za njih velja nekaj omejitev:

- brisanje vsebine, ki ima odvisne vsebine ni dovoljeno, razen če se eksplicitno izbrišejo tudi vse odvisne vsebine,
- premikanje in urejanje odvisne vsebine ni dovoljeno, razen če se vsebine ne odklopi (angl. Detach) ter jo naredi samostojno.

Samodejno pretvarjanje vsebin se izvaja s pomočjo vtičnikov, ki jih s konfiguracijo povežemo v verige. Verige se nato izvajajo na strežniku ter ustvarjajo odvisne vsebine.

Primer konfiguracijske verige:

```
<ns1:Converter xsi:type="ns1:ConverterSettings">
  <ns1:Id xsi:type="xsd:string">c6d25a2e-f1ea-40ba-9267-39e4c6b72ef1</ns1:Id>
  <ns1:SourceOperation xsi:type="ns1:ContentSourceOperation">None</ns1:SourceOperation>
  <ns1:Filter xsi:type="ns1:ContentTypeFilter" disposition="Include">
    <ns1:ContentType xsi:type="xsd:string">image/bmp</ns1:ContentType>
  </ns1:Filter>
  <ns1:Output queueForFTI="true" nextConverterId="c3287e71-068a-4ec5-9277-985303f810de"
descriptionExpression="%DESC_BASE%-
%NOW_HOUR%:%NOW_MINUTE%:%NOW_SECOND%.jpg">image/jpeg</ns1:Output>
  <ns1:Provider xsi:type="ns1:ServiceProvider" id="ImageMagic-BMP" type="Plugin"
driver="libiaconvim.so.1">
    <ns1:Arguments
xsi:type="xsd:string">&lt;WorkDirectory>/iarc/work/conversion&lt;/WorkDirectory></ns1:Arguments>
  </ns1:Provider>
</ns1:Converter>
```

```
<ns1:Converter xsi:type="ns1:ConverterSettings">
  <ns1:Id xsi:type="xsd:string">c3287e71-068a-4ec5-9277-985303f810de</ns1:Id>
  <ns1:SourceOperation xsi:type="ns1:ContentSourceOperation">None</ns1:SourceOperation>
  <ns1:Filter xsi:type="ns1:ContentTypeFilter" disposition="Include"/>
  <ns1:Scope xsi:type="ns1:EntityId" type="ClassificationCode">C=61</ns1:Scope>
  <ns1:Output queueForFTI="true">image/tiff</ns1:Output>
  <ns1:Provider xsi:type="ns1:ServiceProvider" id="ImageMagic-TIFF" type="Plugin"
driver="libiaconvim.so.1">
    <ns1:Arguments
xsi:type="xsd:string">&lt;WorkDirectory>/iarc/work/conversion&lt;/WorkDirectory></ns1:Arguments>
  </ns1:Provider>
</ns1:Converter>
```

Veriga je sestavljena iz dveh vtičnikov. Prvi vtičnik (id: c6d25a2e-f1ea-40ba-9267-39e4c6b72ef1) najprej pretvori BMP v JPEG, nato pa drugi vtičnik pretvori JPEG v TIFF. Rezultat so tri med seboj odvisne datoteke (JPEG je reprezentacija BMP, TIFF je reprezentacija JPEG).

V nadaljevanju sledi podroben opis posameznih etiket in atributov konfiguracijskega XML.

Etiketa »Id«: Etiketa vsebuje unikaten identifikator vtičnika.

Etiketa »SourceOperation«: Vrednost etikete predstavlja operacijo, ki se bo izvedla po uspešni pretvorbi na izvorni vsebini. Naslednja tabela prikazuje veljavne vrednosti z njihovimi opisi.

Vrednost	Opis
NONE	Izvorna vsebina ostane nedotaknjena.
DELETE	Izvorna vsebina se izbriše.
REPLACE	Izvorna vsebina se nadomesti z vsebino, ki je bila na novo ustvarjena.

Tabela 7: Vrednosti etikete "SourceOperation"

Etiketa »Filter«: Vrednost etikete predstavlja vhodni seznam MIME tipov, katere lahko vtičnik pretvori (vrednost atributa »disposition« je »Include«) oziroma za katere vtičnik nima podpore (vrednost atributa »disposition« je »Exclude«). Če je vrednost »disposition« atributa »Include« in je seznam prazen, potem tak vtičnik ni konfiguriran kot vhodni vtičnik v verigi.

Etiketa »Output«: Vrednost etikete predstavlja MIME tip pretvorjeve vsebine. Naslednja tabela prikazuje atribute etikete ter njihove opise.

Atribut	Opis
queueForFTI	Vrednost »True« pomeni, da se bo novo ustvarjena vsebina indeksirala za kasnejše iskanje po vsebini. Vrednost »False« pomeni, da se vsebina ne bo indeksirala.
nextConverterId	Vrednost je unikatni identifikator vtičnika, ki bo naslednji izvajal pretvarjanje vsebine. Če atribut ni prisoten se pretvorbena veriga konča.
descriptionExpression	Vrednost predstavlja opis vsebine, ki jo bo vtičnik ustvaril. Če atribut ni prisoten je opis enak opisu izvorne vsebine. Rezervirane besede za opis so naslednje: • %PAGE_COUNT% - beseda se nadomesti s številom strani pretvorjene vsebine; • %DESC_BASE% - beseda se nadomesti z opisom izvorne vsebine brez končnice (za končnico se šteje zapis za zadnjo piko v izvornem opisu); • %DESC_EXT% - beseda se nadomesti s končnico (pika je vključena) iz izvornega opisa; • %DESC% - beseda se nadomesti z izvornim opisom vsebine; • %NOW_YEAR% - beseda se nadomesti z letom trenutnega datuma pretvorbe; • %NOW_MONTH% - beseda se nadomesti z mesecem trenutnega datuma pretvorbe; • %NOW_DAY% - beseda se nadomesti z dnevom trenutnega datuma pretvorbe; • %NOW_HOUR% - beseda se nadomesti z uro trenutnega časa pretvorbe; • %NOW_MINUTE% - beseda se nadomesti z minuto trenutnega časa pretvorbe; • %NOW_SECOND% - beseda se nadomesti s sekundo trenutnega časa pretvorbe.

Tabela 8: Atributi etikete "Output"

Etiketa »Provider«: Vrednost etikete so argumenti, ki jih posamezen vtičnik potrebuje za pretvarjanje vsebine. Naslednja tabela prikazuje atribute etikete ter njihove opise.

Atributi	Opis
id	Identifikator vtičnika
type	Tip vtičnika – vrednost mora biti »Plugin«
driver	Ime datoteke, kjer se vtičnik nahaja

Tabela 9: Atributi etikete "Provider"

3.3.10.2 Izvajanje drugih operacij

Poleg prej opisanih operacij lahko nad vsebino entitete izvajamo še:

- premikanje vsebine (angl. Move);
- odklapljanje odvisne vsebine (angl. Detach);
- izvajanje zahtev za indeksiranje (angl. Index) in samodejno pretvorbo vsebin (angl. Convert).

Premikanje vsebine: Vsebino lahko premikamo med entitetami. Pogoj je, da je vsebina neodvisna, ter da se premik izvaja med atributi istega tipa. Izvorna entiteta mora biti odprta v načinu za urejanje. Uporabnik mora imeti na tarčnih entitetah pravico urejanja, saj se tarčne entitete ob premiku vsebine implicitno odprejo v načinu za urejanje. Premik vsebine se dejansko izvede ob shranitvi izvorne entitete.

Odklapljanje odvisne vsebine: Odvisno vsebino, ki je rezultat avtomatične pretvorbe lahko odklopimo. S tem dobimo neodvisno vsebino, za katero veljajo vse lastnosti kot za druge neodvisne vsebine.

Izvajanje zahtev za indeksiranje vsebine: Za vsako vsebino (tako odvisno kot neodvisno) se lahko izvaja zahteva za indeksiranje. Entiteta, ki vsebino vsebuje mora biti odprtva v načinu za branje. Uporabnik pa mora imeti »ContentManagement« vlogo, drugače je zahteva za indeksiranje vsebine zavrnjena.

Izvajanje zahtev za samodejno pretvorbo vsebin: Zahtevo za samodejno pretvorbo vsebin lahko izvaja uporbnik z vlogo »ContentManagement«, vsebina mora biti neodvisna, entiteta pa mora biti odprta v načinu samo za branje. V nasprotnem primeru se zahteva za avtomatsko pretvorbo vsebine zavrne.

3.4 Razvrščanje

Razvrščanje (klasifikacija) dokumentacije je bil nepogrešljiv pripomoček že pri klasičnem upravljanju, oziroma hrambi dokumentacije v papirni obliki.
Enako velja tudi za elektronsko upravljanje in elektronsko hrambo. Za potrebe razvrščanja je potrebno v strežniku IMiS®/ARChive Server predhodno vzpostaviti načrt razvrščanja dokumentarnega gradiva (klasifikacijski načrt), ki omogoča:

- strukturiranje, razčlenjevanje in razvrščanje dokumentacije po vsebini, pristojnostih, dejavnostih ter poslovnih in strokovnih funkcijah;
- določanje rokov hrambe dokumentarnega gradiva;
- določanje arhivskega gradiva, oblike dokumentacije in strukture metapodatkov, ki so dodani zadevam in dokumentom, vključno z vodenjem evidence o fizičnem gradivu.

Strežnik omogoča vzpostavitev načrta razvrščanja gradiva poljubnih, praktično neomejenih dimenzij. Pri tem ne omejuje števila nivojev razredov, kakor tudi ne števila podrazredov posameznega razreda. Število nivojev razredov v posameznih delih arhiva je lahko različno. Načrt razvrščanja gradiva predstavlja osnovo za nadzor dostopa do posameznih delov arhiva in je v arhivu predstavljen s hierarhijo entitet vrsta razred.

3.4.1 Klasifikacijske oznake

Vsaka entiteta v arhivu ima svojo polno klasifikacijsko oznako, ki je unikatna za celoten arhiv. Dodeljena je ob nastanku entitete in je nespremenljiva, razen v primeru premika entitete v arhivu (reklasifikacija).

Polna klasifikacijska oznaka razreda je sestavljena iz polne klasifikacijske oznake nadrejenega razreda. Tej je dodan del oznake, ki je entiteti lasten in unikaten med vsemi entitetami, ki imajo isto nadrejeno entiteto. Opcijsko je lahko dodano tudi poljubno ločilo,
ki je enotno za cel arhiv in ga določi administrator arhiva v nastavitvah strežnika
(primer: ».«).

Polna klasifikacijska oznaka zadeve je sestavljena iz polne klasifikacijske oznake razreda kamor je uvrščena. Tej je dodana tudi unikatna oznaka zadeve znotraj tega razreda. Med obema komponentama lahko opcijsko nastopi ločilo, ki ga določi administrator arhiva in je lahko različno od ločila, ki nastopa med oznakami nivojev razreda. Tega določi administrator arhiva v nastavitvah strežnika (primer: »–«).

Polna klasifikacijska oznaka dokumenta je sestavljena iz polne klasifikacijske oznake zadeve ali razreda, ki ji/mu dokument pripada. Tej je dodana unikatna oznaka samega dokumenta znotraj nadrejene entitete. Pred to oznako lahko opcijsko nastopi tudi ločilo, ki ga določi administrator arhiva in je namenjeno ločevanju komponente dokumenta v polnih klasifikacijski oznakah. Tega določi administrator arhiva v nastavitvah strežnika (primer: »/«).

Strežnik IMiS®/ARChive Server samodejno določa klasifikacijske oznake ob dodajanju novih entitet vseh vrst (glej poglavje 3.3.6.3 Klasifikacijska oznaka).
Če samodejno določanje klasifikacijskih oznak za novo entiteto ni določeno (nastavitev posamične entitete za njene podrejene entitete), strežnik zavrne zahtevo za nastanek nove entitete. To stori v primeru, če zahtevku ni dodana vrednost klasifikacijske oznake, ali če podana klasifikacijska oznaka ni unikatna znotraj novi entiteti nadrejene entitete.

3.4.2 Nastavitve načrta razvrščanja gradiva

Podroben in praktično dokončen načrt razvrščanja gradiva mora biti praviloma določen ob uvodni konfiguraciji strežnika IMiS®/ARChive Server, oziroma pred začetkom njegove uporabe. Administrator v hierarhijo načrta razvrščanja gradiva doda vse potrebne razrede.
Za dodajanje razredov je potrebno predhodno:

- ustvariti predloge za nove razrede (glej poglavje 3.3.4 Predloge);
- nastaviti samodejno generiranje klasifikacijskih oznak (po potrebi).

Pred nastankom zadev in dokumentov je potrebno pripraviti:

- predloge za zadeve v končnih razredih (razredi, ki nimajo vsebovanih razredov);
- predloge za dokumente, ki bodo nastajali v zadevah ali neposredno v razredih.

S predlogami določimo:

- metapodatke, ki so za posamezne entitete obvezni;
- metapodatke, ki so dovoljeni za vnos;
- metapodatke, kjer je za isto entiteto možno vnesti več vrednosti;
- druge lastnosti entitet.

Vsaka predloga lahko določa, katere predloge je mogoče uporabiti za nastanek podrejenih entitet v entiteti, ki je nastala z uporabo te predloge. Dovoljene predloge za nastanek podrejenih entitet je mogoče določiti tudi neposredno na že obstoječih entitetah arhiva.
Tako razredi, ki so na zadnjem nivoju razredov v arhivu navadno vsebujejo zadeve, izjemoma pa lahko vsebujejo tudi samostojne dokumente.
Vsak razred vedno lahko vsebuje samo entitete enega tipa, torej samo podrazrede ali samo zadeve ali samo dokumente.

Natančna priprava načrta razvrščanja gradiva je pred uporabo strežnika zelo pomembna. Strežnik IMiS®/ARChive Server namreč ne dovoljuje zamenjave predloge za že nastale in shranjene entitete. Obenem ne dovoljuje sprememb predlog,
ki so že uporabljene za nastanek entitet, razen dodajanja metapodatkov v predlogo.

Po začetku uporabe strežnika, ko se v njem že nahajajo zadeve in/ali dokumenti pa so možne naslednje operacije, ki vplivajo na načrt razvrščanja gradiva:

- Nove razrede na kateri koli nivo je v arhiv mogoče dodajati kadarkoli.
  Novim razredom bodo klasifikacijske oznake dodeljene samodejno ali ročno, odvisno od nastavitev samodejnega dodeljevanja klasifikacijskih oznak.
- Brisanje razredov je mogoče, ko razred ne vsebuje nobene entitete, torej v primeru,
  ko je popolnoma prazen.
- Dodajanje predlog, ki določajo metapodatke podrejenih entitet. Mogoče jih je dodajati na določeno mesto v hierarhiji.
- Premikanje entitet je mogoče skupaj s celotnim drevesom entitet, ki ga premikajoča se entiteta vsebuje (glej poglavje 3.4.3 Premikanje gradiva v načrtu razvrščanja gradiva).

Opozorilo: Vse naštete operacije so mogoče le, če uporabniku to dovoljujejo dostopne pravice.
Tako je administratorjem dovoljeno upravljanje z dostopnimi pravicami kadarkoli po nastanku hierarhije razvrščanja.

Uporabnikom, ki jim dostopne pravice to omogočajo, je kadarkoli po nastanku razredov omogočeno tudi:

- spreminjanje metapodatkov razredov;
- zapiranje razredov, kar onemogoči:
  - dodajanje novih entitet v zaprt razred ali v hierarhijo pod njim
  - spreminjanje vsebine arhiva kjerkoli v hierarhiji pod zaprtim razredom, vključno z metapodatki razreda samega.

3.4.3 Premikanje gradiva v načrtu razvrščanja (reklasifikacija)

Kljub skrbnemu načrtovanju razvrščanja gradiva, prihaja v življenjski dobi elektronskega arhiva do potrebe po spremembah načrta razvrščanja gradiva.
Vse večkrat pa prihaja do potreb po spremembi klasifikacijske oznake določene zadeve ali dokumenta. V ta namen strežnik IMiS®/ARChive Server ponuja možnost premikanja entitete katerega koli tipa na drugo mesto znotraj arhiva.
Premikanje entitete, ki vsebuje hierarhijo drugih entitet, povzroči premik celotne hierarhije na drugo mesto.
Za uspešno premikanje entitete v arhivu morajo biti izpolnjeni naslednji pogoji:

- uporabnik mora imeti pravico branja entitete, ki se premika (glej poglavje 3.3.5.2.1 Dostopne pravice za entiteto);
- uporabnik mora imeti pravico premika entitete, ki se premika (glej poglavje 3.3.5.2.1 Dostopne pravice za entiteto);
- uporabnik mora imeti pravico ustvarjanja novih entitet na mestu, kamor premika izbrano entiteto (glej poglavje 3.3.5.2.1 Dostopne pravice za entiteto);
- predloga premikajoče se entitete mora ustrezati predlogam, ki so določene kot dovoljene na mestu, kamor uporabnik entiteto premika;
- premikajoči se entiteti nadrejena entiteta ne sme biti zaprta;
- hierarhija kamor premika entiteto ne sme biti zaprta;
- omogočeno mora biti samodejno dodeljevanje klasifikacijskih oznak za vse entitete v hierarhiji pod premikajočo se entiteto;
- stopnja tajnosti (angl. Security class) premikajoče se entitete mora biti manjša ali enaka stopnji tajnosti entitete, kamor uporabnik entiteto premika.

Ob uspešnem premeščanju entitete oziroma hierarhije entitet, so vsem premaknjenim entitetam dodeljene nove klasifikacijske oznake, ki ustrezajo pravilom označevanja na novi lokaciji.
Krovni entiteti premaknjenih entitet so poleg tega dodani metapodatki:

- »sys:MoveReason«: razlog za premik entitete, ki ga obvezno posreduje uporabnik ob premiku;
- »sys:MoveAgent«: uporabnik, ki je premik izvedel;
- »sys:MoveDateTime«: datum in čas premika (konca);
- »sys:MoveClassificationCode«: polna klasifikacijska oznaka entitete pred premikom.

Za podrobnejši pomen naštetih metapodatkov glej poglavje 3.2.5 Sistemski atributi.
Za vse premaknjene entitete se v revizijsko sled zabeleži dogodek opisan v poglavju 3.3.7.9 Premik.

V revizijsko sled se ob dogodku premika entitete in njej podrejenih entitet vnese še sporočilo ob premiku, ki vsebuje naslednje informacije:

- trenutni status entitete
- stara in nova vrednost polne klasifikacijske oznake entitete
- razlog za premik, ki ga je ob premiku navedel uporabnik, ki je izvedel premik
- vrednosti vseh metapodatkov, ki jih je ob premiku entiteta vsebovala.

3.5 Iskanje

Ena najpomembnejših funkcionalnosti strežnika IMiS®/ARChive Server je zmožnost iskanja arhiviranega gradiva ter prikaz zadev in dokumentov glede na pravice dostopov.
Uporabnik lahko išče vse vrste entitet glede na vrednosti posameznih metapodatkov, ključnih besed in/ali glede na vsebino polnega besedila v arhivu shranjenih dokumentov.
Opcijsko lahko vključimo tudi rekurzivna iskanja in iskanja po podedovanih vrednostih v primeru atributov, ki imajo glede na svoje nastavitve vrednosti podedovane (npr. stopnja tajnosti).

3.5.1 Varnost in zaščita podatkov pri iskanju

Rezultati iskanja so lahko samo razredi, zadeve ali dokumenti do katerih ima uporabnik pravico dostopa, bodisi iz naslova stopnje tajnosti, bodisi kot seznamov dostopnih pravic (ACL). Entitete, do katerih uporabnik nima pravice dostopov, ostanejo uporabniku zakrite,
čeprav ustrezajo iskalnim pogojem.

Ostale pravice za operacije nad entitetami, ki so rezultat iskanja, so enake pravicam pri običajnem pregledovanju arhiva.

Primer: Če uporabnik nima pravice za odpiranje entitete v bralskem načinu, bo v rezultatih iskanja videl samo njen obstoj in njene javne metapodatke.

3.5.2 Pravila iskalnega niza

Enostaven pogoj je osnovni del iskalnih nizov. Strežnik IMiS®/ARChive Server pozna dve vrsti enostavnih pogojev:

- pogoji za iskanje po metapodatkih
- pogoji za iskanje po polnem besedilu dokumentov.

Iskalni niz je sestavljen iz najmanj enega enostavnega pogoja.
Zahtevnejši iskalni nizi so sestavljeni iz več enostavnih pogojev, ki so v nizu združeni z logičnimi operacijami (IN, ALI, izključujoči ALI in negacija).
Zaporedje računanja logičnih operacij lahko dodatno določimo z uporabo oklepajev.

3.5.2.1 Enostavni pogoji za iskanje po metapodatkih

Strežnik IMiS®/ARChive Server omogoča iskanje po vseh metapodatkih, za katere je glede na nastavitve sistema omogočeno iskanje. Iskanje je na voljo za metapodatke vseh tipov.

Enostaven pogoj za iskanje po metapodatkih sestavljajo tri komponente:

- Ime metapodatka za katerega velja pogoj. V pogoju je ime metapodatka vedno navedeno v oglatih oklepajih. Primer: [Znesek].
- Operacija primerjanja. Poznane operacije za primerjanje v enostavnih pogojih so: manjše (<), manjše ali enako (<=), enako (= ali ==), različno (<> ali !=), večje ali enako (>=) in večje (>).
- Vrednost za katero velja operacija primerjanja. Vse vrednosti so lahko navedene v dvojnih narekovajih. Če vrednost vsebuje preslednice, je uporaba dvojnih narekovajev obvezna (npr. »Spodnja Kungota«).

  Znak za ločevanje celega dela od decimalk v decimalnih vrednostih je pika. Datumske in časovne vrednosti morajo biti navedene v XML notaciji. Pri datumskih in časovnih vrednostih lahko uporabimo štiri posebne izraze, ki nadomestijo celotno vrednost ali enega od njenih delov.

  @NOW@ cela vrednost bo sestavljena iz trenutnega datuma in časa

Primer: [ZacetekSeje] < @NOW@

@TODAY@ datum v vrednosti bo enak današnjemu datumu
Primer: [CasPrejema] < @TODAY@T08:00+02:00

@YEAR@ leto v vrednosti bo enako trenutnemu letu
@MONTH@ mesec v vrednosti bo enak trenutnemu mesecu
Primer: [CasFotografije] < @YEAR@@MONTH@-15T20:00+02:00

Posebna vrednost za pogoje iskanja je izraz NULL. To vrednost je mogoče uporabiti za pogojevanje obstoja vrednosti določenega metapodatka.
Pri uporabi vrednosti NULL in pri metapodatkih logičnega tipa je možna samo uporaba primerjalnih operatorjev enako (== ali =) in različno (<> ali !=).

- Opcijsko lahko za imenom metapodatka navedemo aritmetično operacijo seštevanja ali odštevanja (+ ali -), katere drugi operand je konstantna vrednost.
  Pri znakovnih in logičnih metapodatkih ta opcija ni na voljo.
  Za datumske metapodatke je predpisana posebna oblika konstantne vrednosti, sestavljena iz najmanj ene ali vseh treh komponent, ki predstavljajo leta, mesece in dneve. Posamezne komponente so sestavljene iz celega števila in črke, ki predstavlja datumsko enoto kot izhaja iz naslednjih primerov:
  - za leta uporabimo črko y ali Y (npr. 15y)
  - za mesece m ali M (npr. 3m)
  - za dneve d ali D (npr. 60d).

Primeri: enostavni pogoji za iskanje po metapodatkih:
[Znesek] - 40.5 >= 250
[Avtor]="Janez Novak"
[ZacetekSeje] + 1y3m <= 2014-05-07T11:05+02:00
[KonecSeje] + 7d <= @NOW@
[Avtor]!=NULL

3.5.2.2 Enostavni pogoji za iskanje po polnem besedilu dokumentov

Strežnik IMiS®/ARChive Server omogoča iskanje po polnem besedilu dokumentov, za katere je glede na nastavitve strežnika omogočeno iskanje. Možnost iskanja je na voljo za datoteke vseh razširjenih tipov, katerih vsebina je lahko besedilo.

Enostavni pogoji za iskanje po polnem besedilu dokumentov so v iskalnih nizih navedeni v zavitih oklepajih: {"Janez Novak"}.Znotraj zavitih oklepajev je enostaven pogoj za iskanje po polnem besedilu z uporabo logičnih operacij mogoče razširiti na kompleksnejši pogoj za iskanje po polnem besedilu. Podprte logične operacije v pogojih za iskanje po polnem besedilu,
navedene po vrstnem redu izračunavanja, so:

- negacija: NOT
- in: AND
- ali: OR.

Znotraj pogojev za iskanje po polnem besedilu je z uporabo navadnih oklepajev možno spremeniti vrstni red izračunavanja operacij.

Primer: {"Novak" AND "Janez" OR "Franc"}
najde vsa besedila, kjer nastopata obe besedi »Novak« in »Janez« in besedila kjer nastopa beseda »Franc«.

Primer: {"Novak" AND ("Janez" OR "Franc")}
najde vsa besedila, kjer obvezno nastopa beseda "Novak" v kombinaciji z besedo »Janez« ali besedo »Franc«.

3.5.2.3 Logične operacije v iskalnih nizih

Enostavni pogoji so v kompleksnejše iskalne nize povezani z logičnimi operacijami.
Strežnik IMiS®/ARChive Server podpira naslednje logične operacije, ki so navedene po vrstnem redu izračunavanja:

- negacija: NOT ali !
- in: AND ali &
- izključujoči ali: XOR ali ^
- ali: OR ali |.

Primer: Izraz:[A]>3 OR [A]=3 AND [B]!=NULL
je enakovreden izrazu: NOT [B]== NULL & [A]==3 | [A]>3

V obeh primerih se najprej izračuna operacija »in«, nato pa še operacija »ali«.
V drugem primeru se pred »IN« izračuna še negacija »NOT«.

Z uporabo navadnih oklepajev je mogoče spremeniti privzeti vrstni red izračuna logičnih operacij. Zgornji iskalni izraz lahko z uporabo oklepajev postane povsem nekaj drugega in posledično bo rezultat iskanja popolnoma drugačen.
Primer: Uporaba oklepajev povzroči, da se operacija »ALI« izračuna pred operacijo »IN«:([A]>3 OR [A]=3) AND [B]!=NULL

V istem iskalnem nizu je možna hkratna uporaba obeh tipov enostavnih pogojev.

Primer: Izraz:[Povrsina] >= 12 AND {"Janez AND Novak"}
je povsem pravilen.

Priporočljivo je, da se pogoji za iskanje po polnem besedilu združijo v en izraz znotraj zavitih oklepajev, če je to le mogoče.

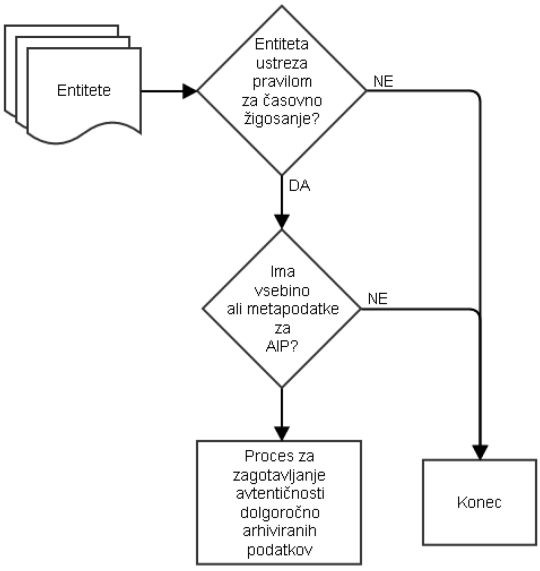
Primer: Izraz:[Povrsina] >= 12 AND {"Janez"} AND {"Novak"}
bo na primer vrnil enake rezultate kot izraz:
[Povrsina] >= 12 and {"Janez" AND "Novak"}
Drugi izraz bo iskanje izvedel nekoliko hitreje kot prvi.

3.6 Avtentičnost

3.6.1 Predpogoji

Skozi življenjski cikel entitet se lahko spreminjajo njeni metapodatki in vsebina.
Predpogoj za zagotavljanje avtentičnosti entitet je njihova nespremenljivost.
Strežnik IMiS®/ARChive Server za matematično preverjanje nespremenljivosti uporablja priporočila delovne skupine LTANS (angl. Long-Term Archive and Notary Services), ki predpisujejo »de-facto« standarde postopkov dolgoročnega arhiviranja vsebin.
V postopek se vključujejo entitete po pravilih, ki so opisana v nadaljevanju (glej poglavje 3.6.7 Pravila). Priporočljivo je, da so pravila napisana tako, da se v postopek avtentičnosti vključujejo samo entitete, ki so nespremenljive (npr. zaprte entitete). Ker se lahko vsebina entitet tudi kasneje spreminja, se vedno znova ustvarjajo tudi dokazila za tako entiteto.
Kot dodatno mora imeti entiteta vsaj en metapodatek, ki označuje, da je predmet arhivskega informacijskega paketa (angl. Archival Information Package – AIP).
AIP je XML reprezentacija vsebine entitete in njenih metapodatkov in je podrobneje opisan v nadaljevanju.

Avtentičnost AIP in posledično entitete dokazuje pripadajoča sintaksa evidenčnih podatkov (ERS), ki se ustvari v postopku zagotavljanja avtentičnosti dolgoročno arhiviranih podatkov. Spodnja slika prikazuje postopek umeščanja entitet, ki ustrezajo predpogojem v postopku za zagotavljanje avtentičnosti dolgoročno arhiviranih podatkov.



Slika 7: Umeščanje entitete v postopku za zagotavljanje avtentičnosti dolgoročno arhiviranih podatkov

3.6.2 Koncept

Eden izmed ključnih konceptov zagotavljanja varnega elektronskega arhiva je zagotavljanje avtentičnosti arhiviranega gradiva skozi ves čas njegove hrambe.
To pomeni, da je potrebno ohraniti izvirne lastnosti dokumenta glede na kontekst, strukturo in njegovo vsebino. Bistveno je, da se od trenutka arhiviranja dokumenta vsi njegovi deli (struktura, vsebina, metapodatki, ...) potrebni za zagotovitev avtentičnosti, ne spremenijo več.

Avtentičnost elektronsko arhiviranega gradiva lahko zagotovimo z uporabo naslednjih funkcionalnosti:

- prstni odtis (angl. Hash - #)
- elektronski podpis z digitalnim potrdilom (angl. Digital signature)
- časovni žig (angl. Timestamp).

3.6.2.1 Prstni odtis

Prstni odtis je rezultat enosmerne zgoščevalne funkcije, ki za vhodni podatek vzame blok podatkov (npr. niz zlogov binarne vsebine dokumenta, izvorno besedilo dokumenta) in izračuna prstni odtis fiksne dolžine. Minimalna sprememba vhodnih podatkov se odraža v veliki spremembi vrednosti prstnega odtisa.

Lastnosti idealne zgoščevalne funkcije so naslednje:

- za vsak vhodni blok podatkov je preprosto izračunati prstni odtis;
- iz prstnega odtisa je nemogoče ugotoviti vhodni blok podatkov (zgoščevalna funkcija je enosmerna);
- nemogoče je spremeniti vhodni blok podatkov na tak način, da bo prstni odtis nespremenjen;
- teoretično je nemogoče, da bi prišlo do sovpadanja zgoščenih vrednosti različnih vhodnih blokov (angl. Collision).

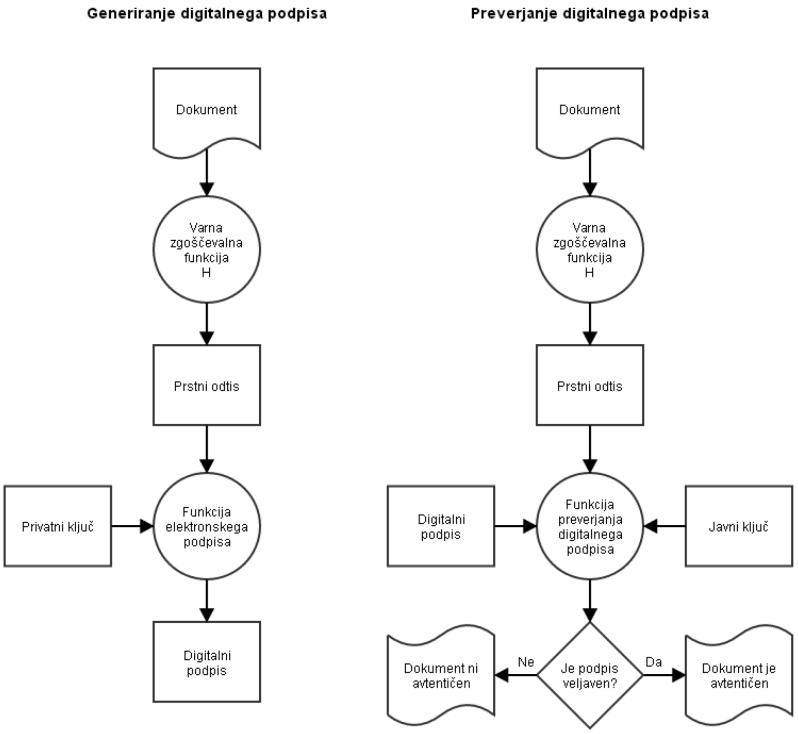
Z razvojem tehnologije in večanjem računalniške moči, se veča tudi možnost sovpadanja zgoščenih vrednosti različnih vhodnih blokov, kar oslabi varnost zgoščevalne funkcije.
Tako danes velja MD5 zgoščevalna funkcija za zlomljivo in ni več varna za uporabo.

Slika 8: Delovanje zgoščevalne funkcije

3.6.2.2 Elektronski podpis z digitalnim potrdilom

Elektronski podpis temelji na infrastrukturi javnih ključev (angl. Public Key Infrastructure – PKI) in matematično zagotavlja avtentičnost elektronskega dokumenta.
Shema elektronskega podpisa temelji na treh algoritmih:

- ustvarjanje ključev: ustvarita se privatni in javni ključ, ki sta ključna za izdelavo in preverjanje podpisa;
- algoritem za podpisovanje: na podlagi privatnega ključa ter dokumenta se ustvari elektronski podpis;
- algoritem za preverjanje podpisa: na podlagi elektronskega podpisa in javnega ključa se preveri avtentičnost dokumenta.



Slika 9: Generiranje in preverjanje elektronskega podpisa

Veljavnost elektronskega podpisa je pogojena z veljavnostjo digitalnega potrdila.
Glede na 32. člen Uredbe za elektronsko poslovanje in elektronsko podpisovanje omejeno na največ 5 let. Za ohranjanje avtentičnosti dokumenta je potrebno dokument podpisati z novim digitalnim potrdilom pred iztekom veljavnosti starega.

3.6.2.3 Časovni žig

Postopek časovnega žigosanja je ekvivalenten elektronskemu podpisovanju, le da je vključen tudi izdajatelj varnih časovnih žigov (angl. Time Stamping Authority – TSA).
Slednji elektronskemu podpisu doda časovno komponento, ki enolično določa čas elektronskega podpisa dokumenta. Enako, kot pri elektronskemu podpisu, je tudi veljavnost časovnega žiga omejena z veljavnostjo digitalnega potrdila,
ki ga je potrebno za zagotavljanje avtentičnosti dokumenta podaljševati.
Časovni žig se uporablja za podaljševanje veljavnosti elektronskih podpisov.

Gre za praktično rešitev zlasti v primeru, ko je vsebino podpisalo več oseb, ki jim bo poteklo digitalno potrdilo. Brez časovnega žiga bi morali za zagotovitev avtentičnosti vsebine vsi podpisniki, ki jim je digitalno potrdilo preteklo še enkrat elektronsko podpisati vsebino.

Na ta način časovni žig zagotavlja avtentičnost vsebine tudi, ko so digitalna potrdila podpisnikov že pretečena.

Vsi zgoraj našteti načini (prstni odtis, elektronski podpis z digitalnim potrdilom, časovni žig) imajo pomanjkljivosti, tako tehnološke (prstni odtis) kakor tudi časovne (veljavnost digitalnega potrdila). Zato uporaba posameznega načina ni primerna za zagotavljanje dolgoročne avtentičnosti dokumenta.
Tako obstaja kar nekaj standardov, ki za dolgoročno zagotavljanje avtentičnosti dokumentov uporabljajo kombinacijo prstnih odtisov, elektronskih podpisov in časovnih žigov:

- standardi ETSI;
- sintaksa evidenčnih podatkov (angl. Evidence Record Syntax – ERS);
- okolje revidiranega nadzora (angl. Auditing Control Environment – ACE);
- servis celovitosti vsebine (angl. Content Integrity Service – CIS).

Strežnik IMiS®/ARChive Server za dolgoročno zagotavljanje avtentičnosti dokumentov uporablja ERS standard v njegovi XML obliki (XMLERS, RFC 6283) opisan v nadaljevanju.

3.6.3 Shramba digitalnih potrdil

Za delo z digitalnimi potrdili (certifikati) uporablja strežnik IMiS®/ARChive Server shrambo digitalnih potrdil (angl. Certificate Store). Temelji na odprtokodni knjižnici OpenSSL.
Shramba digitalnih potrdil omogoča:

- dodajanje digitalnih potrdil
- omogočanje in onemogočanje digitalnih potrdil
- iskanje digitalnih potrdil
- iskanje po seznamih preklicanih digitalnih potrdil.

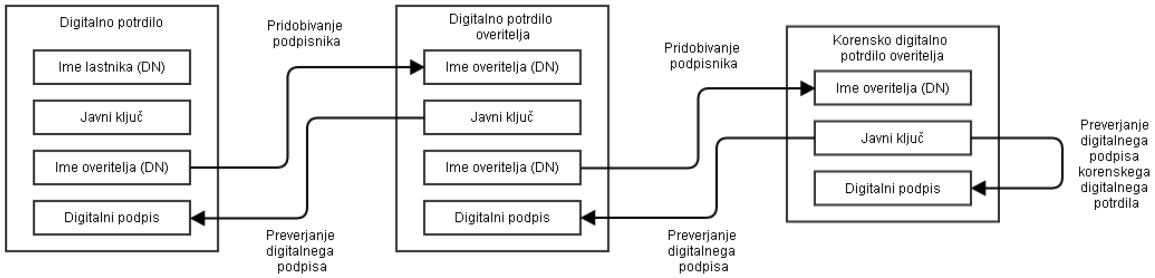
Za hitrejše pridobivanje digitalnih potrdil in aktualnih informacij o preklicih je v shrambi izvedeno predpomnenje (angl. Caching) trenutno uporabljenih verig digitalnih potrdil (angl. Certificate chain) in informacij o preklicih.

3.6.3.1 Veriga digitalnih potrdil

Veriga digitalnih potrdil je seznam digitalnih potrdil, ki dokazujejo avtentičnost posameznega digitalnega potrdila. Vsako digitalno potrdilo v seznamu je podpisano s privatnim ključem naslednjega digitalnega potrdila v tem seznamu.

Na koncu seznama je korensko digitalno potrdilo overitelja (angl. Certificate authority root certificate), ki je samo sebi podpisnik (angl. Self-signed certificate).
Tako verigo imenujemo veriga zaupanja (angl. Chain of trust) in zagotavlja avtentičnost vseh digitalnih potrdil v njej.

Spodnja slika prikazuje primer verige s tremi digitalnimi potrdili ter postopek preverjanja posameznih digitalnih potrdil.



Slika 10: Veriga digitalnih potrdil

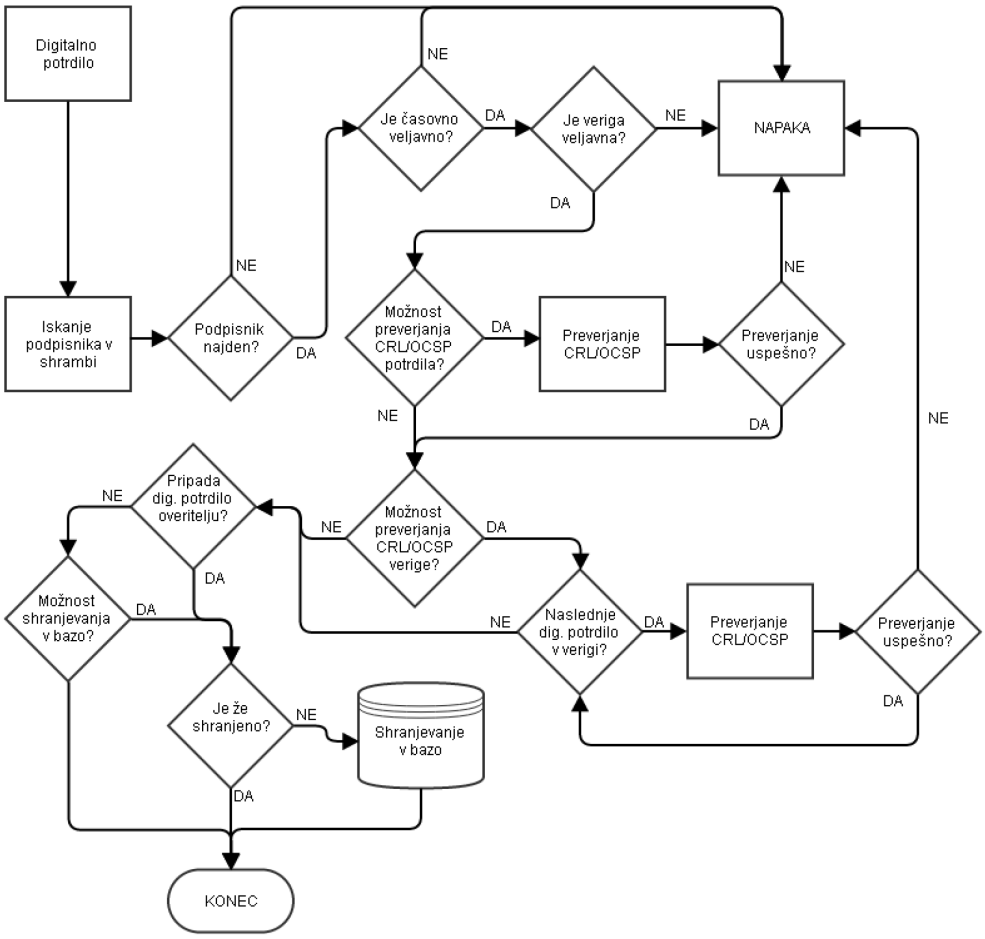
3.6.3.2 Dodajanje digitalnih potrdil

Dodajanje digitalnih potrdil v shrambo je uspešno, če so zadoščeni naslednji pogoji:

- digitalno potrdilo mora biti veljavno v času dodajanja v shrambo (ne sme biti pretečeno);
- shramba mora vsebovati vsa predhodna digitalna potrdila, ki tvorijo verigo zaupanja;
- vsa predhodna digitalna potrdila morajo biti omogočena;
- v primeru, da želimo pri dodajanju digitalnega potrdila preveriti informacije o preklicu le tega ali informacije o preklicih vseh digitalnih potrdil, je dodajanje uspešno le v primeru preverjanja digitalnih potrdil, da le-ta niso preklicana.

Obvezna pogoja za dodajanje digitalnega potrdila v shrambo sta:

- digitalno potrdilo je bilo veljavno v času dodajanja;
- v shrambi so že predhodna digitalna potrdila (in so omogočena), ki tvorijo verigo zaupanja.



Slika 11: Postopek dodajanja digitalnega potrdila v shrambo

3.6.3.3 Omogočanje in onemogočanje digitalnih potrdil

Z omogočanjem in onemogočanjem digitalnih potrdil nadziramo, katerim digitalnim potrdilom zaupamo in katerim ne. Z onemogočanjem posameznih digitalnih potrdil avtomatsko onemogočimo celotno vejo, katero sestavlja onemogočeno digitalno potrdilo. Če se v veji nahaja onemogočen certifikat, potem je taka veja neveljavna oziroma se tak certifikat v veji ne upošteva.

3.6.3.4 Iskanje digitalnih potrdil

Digitalna potrdila lahko iščemo po shrambi po njegovem prstnem odtisu (uporablja se zgoščevalni algoritem SHA-512) ali po njegovem 32-bitnem unikatnem identifikatorju, ki ga dobi ob shranjevanju v podatkovno bazo. Digitalna potrdila, ki niso shranjena v bazi, tega identifikatorja nimajo.

3.6.3.5 Iskanje informacij o preklicih

Iskanje informacij o preklicih je možno le po 64-bitnem unikatnem identifikatorju, ki ga strežnik dodeli v primeru shranjevanja v bazo. Shranjevanje informacij o preklicih se izvaja pri preverjanju časovnega žiga v postopku zagotavljanja avtentičnosti dolgoročno arhiviranih podatkov (glej poglavje 3.6.4 ERS in poglavje 3.6.6 Časovno žigosanje).
Vsako digitalno potrdilo ima t.i. podaljške (angl. Certificate extensions), ki so podrobneje opisani v specifikaciji RFC 5280 (http://tools.ietf.org/html/rfc5280).
Podaljški vsebujejo informacije o distribucijskih točkah, kjer se dobijo informacije o:

- preklicih digitalnega potrdila
- informacije o tipu digitalnega potrdila
- informacije o namenski uporabi digitalnega potrdila
- ...

Strežnik pri obdelavi digitalnih potrdil, poleg informacij o distribucijskih točkah, uporablja še podaljške imenovane osnovne omejitve (angl. Basic constraint) in razširjena uporaba ključa (angl. Extended key usage).

3.6.3.6 Osnovne omejitve

Podaljšek vsebuje informacije o tem, ali digitalno potrdilo pripada overitelju ali ne. Informacija se zapiše v parameter »mCACertificate« digitalnega potrdila in ima lahko vrednosti »1« ali »0«. Vrednost »1« pomeni, da digitalno potrdilo pripada overitelju, »0« pa pomeni da ne.

3.6.3.7 Razširjena uporaba ključa

Podaljšek vsebuje parametre z informacijami o namenu uporabe digitalnega potrdila.

Parameter: »mServerAuthentication«
Veljavne vrednosti: »1« ali »0«
Opis: Če ima vrednost »1«, potem se digitalno potrdilo lahko uporablja za overjanje strežnika pri vzpostavljanju šifrirane povezave med strežnikom in odjemalcem.

Parameter: »mClientAuthentication«
Veljavne vrednosti: »1« ali »0«
Opis: Če ima vrednost »1«, potem se digitalno potrdilo lahko uporablja za overjanje odjemalca pri vzpostavljanju šifrirane povezave med strežnikom in odjemalcem.

Parameter: »mEmailProtection«
Veljavne vrednosti: »1« ali »0«
Opis: Če ima vrednost »1«, potem se digitalno potrdilo lahko uporablja za podpisovanje elektronske pošte.

Parameter: »mCodeSigning«
Veljavne vrednosti: »1« ali »0«
Opis: Če ima vrednost »1«, potem se digitalno potrdilo lahko uporablja za podpisovanje izvršljive kode (angl. Executable code).

Parameter: »mMicrosoftServerGatedCrypto«
Veljavne vrednosti: »1« ali »0«
Opis: Parameter se ne uporablja

Parameter: »mNetscapeServerGatedCrypto«
Veljavne vrednosti: »1« ali »0«
Opis: Parameter se ne uporablja

Parameter: »mOcspSign«
Veljavne vrednosti: »1« ali »0«
Opis: Če ima vrednost »1«, potem se digitalno potrdilo lahko uporablja za podpisovanje OCSP odgovora strežnika, ki posreduje informacije o preklicih digitalnih potrdil.

Parameter: »mTimestamp«
Veljavne vrednosti: »1« ali »0«
Opis: Če ima vrednost »1«, potem se digitalno potrdilo lahko uporablja za časovno žigosanje.

Parameter: »mDvcs«
Veljavne vrednosti: »1« ali »0«
Opis: Če ima vrednost »1«, potem se digitalno potrdilo lahko uporablja za podpisovanje komponent v DVCS protokolu (protokol opisuje specifikacija RFC 3029 - http://tools.ietf.org/html/rfc3029.

3.6.3.8 Kriptografska razširitev

S kriptografsko razširitvijo JCA (angl. Java Cryptography Extension) je omogočena uporaba shrambe digitalnih potrdil tudi v vtičnikih (vtičniki za imeniške storitve, vtičniki za časovno žigosanje, ...). S tem je omogočena enotna uporaba shrambe digitalnih potrdil v vseh komponentah strežnika IMiS®/ARChive Server. Shrambo digitalnega potrdila v vtičnikih omogočimo tako, da v konfiguracijo vtičnika dodamo rezervirano besedo »IAKS«, ki predstavlja shrambo digitalnega potrdila. Naslednji zapis predstavlja konfiguracijo vtičnika, ki uporablja strežniško shrambo digitalnih potrdil za varno povezovanje na LDAP strežnik.

```
<Arguments>
  <Class>com.imis.imisarc.server.aaa.impl.ActiveDirectory</Class>
  <LdapURL>ldaps://pot.do.streznika</LdapURL>
  ...
  <SSLTSType>IAKS</SSLTSType>
  ...
</Arguments>
```

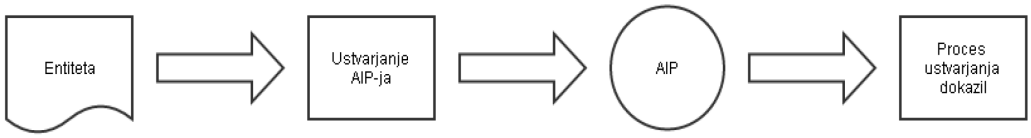
3.6.4 ERS

Sintaksa evidenčnih podatkov (angl. Evidence Record Syntax - ERS) je standard, ki opisuje sistem za zagotavljanje avtentičnosti dolgoročno arhiviranega gradiva. Predpisuje kako naj se dokazila ustvarjajo, podaljšujejo in kako naj bodo strukturirana, da bodo nedvoumno dokazovala avtentičnost arhiviranega gradiva od trenutka, ko je to vstopilo v postopek dolgoročnega zagotavljanja avtentičnosti.

V strežniku IMiS®/ARChive Server je izveden ERS v XML formatu po specifikaciji RFC 6283 (https://tools.ietf.org/html/rfc6283).
Ključna postopka za dolgoročno zagotavljanje avtentičnosti arhiviranega gradiva sta:

- ustvarjanje dokazil
- podaljševanje dokazil.

Pred pričetkom postopka ustvarjanja dokazil je potrebno najprej izbrati vsebino in metapodatke, ki jih bo sintaksa evidenčnih podatkov ščitila, oziroma dokazovala, da so le-ti avtentični.
To se naredi z ustvarjanjem arhivskega informacijskega paketa (AIP) za vsako entiteto, ki ustreza predpogojem za zagotavljanje avtentičnosti dolgoročno shranjenih podatkov (glej poglavje 3.6.1 Predpogoji).

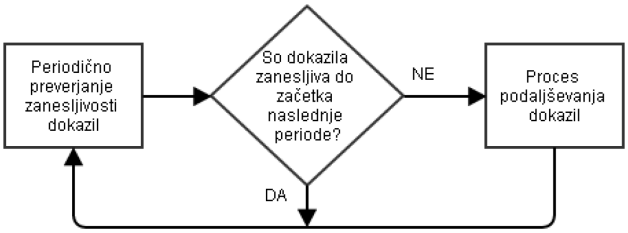


Slika 12: Ustvarjanje AIP za postopek ustvarjanja dokazil

Dokazila za dokazovanje avtentičnosti (prstni odtis, elektronski podpis, časovni žig) imajo omejeno življenjsko dobo. Veljavnost elektronskega podpisa in časovnega žiga sta omejena s časovno veljavnostjo digitalnega potrdila, ki ju je ustvaril.
Zato je potrebno dokazila podaljševati in sicer z generiranjem novega elektronskega podpisa ali časovnega žiga. To imenujemo tudi postopek enostavnega podaljševanja dokazil.

S časom se varnost algoritma za ustvarjanje prstnega odtisa zmanjša, saj se poveča verjetnost sovpadanja zgoščenih vrednosti. Zato je potrebno algoritem nadomestiti z novejšim. S tem zagotovimo, da dokazila ne izgubijo zanesljivosti, kljub slabljenju matematičnih algoritmov, ki jih ustvarjajo. To imenujemo tudi postopek kompleksnega podaljševanja dokazil.

Ustvarjena dokazila je potrebno periodično preverjati ter jih pred izgubo zanesljivosti podaljšati, kot je prikazano na spodnji sliki.

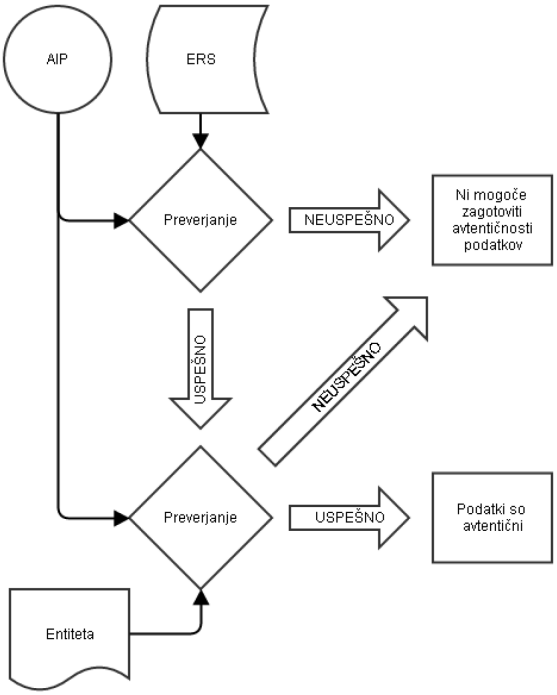


Slika 13: Podaljševanje zanesljivosti dokazil

Preverjanje avtentičnosti arhiviranega gradiva poteka v dveh stopnjah:

- preverjanje AIP s podatki v ERS
- preverjanje vsebine in metapodatkov entitete z vrednostmi v AIP.

Če sta obe preverjanji uspešni pomeni, da so vsebina entitete in metapodatki ostali nespremenjeni skozi ves čas hrambe. Če je katerokoli od preverjanj neuspešno pomeni, da ni zagotovila, da so podatki avtentični.



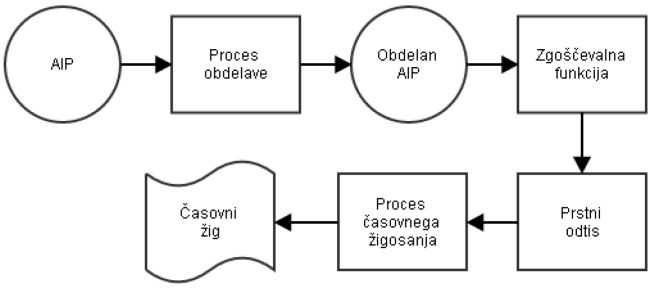
Slika 14: Primer preverjanja avtentičnosti podatkov

3.6.4.1 Postopek ustvarjanja dokazil

Za vsako entiteto, ki pride v postopek zagotavljanja avtentičnosti dolgoročno arhiviranih podatkov se ustvari arhivski informacijski paket (AIP). Tak paket strežnik IMiS®/ARChive Server obdela glede na informacije, ki se nahajajo v glavi paketa (glej poglavje 3.6.5 AIP).

Za obdelan paket se nato izračuna prstni odtis, ki je osnova za časovno žigosanje.

Časovni žig nam nedvoumno dokazuje, da je prstni odtis (ter posledično AIP, kateremu pripada) obstajal pred časom, navedenem v časovnem žigu.



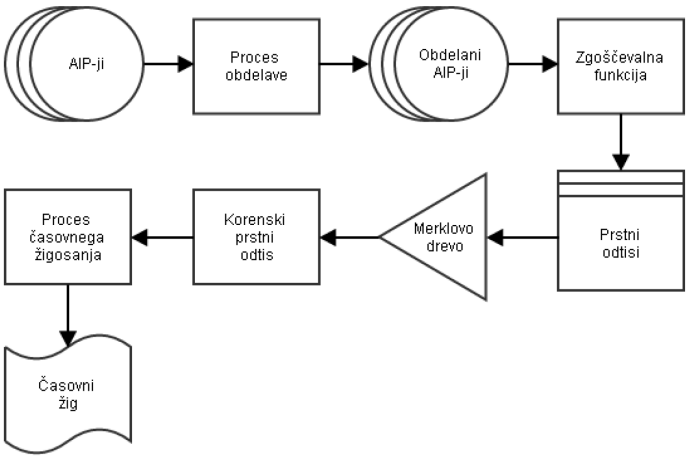
Slika 15: Postopek časovnega žigosanja posameznega AIP-ja

Ker je postopek časovnega žigosanja časovno precej zahteven, istočasno pa večina ponudnikov časovnega žigosanja svojo storitev zaračunava, je žigosanje posameznih prstnih odtisov neracionalno.

Zato se večinoma uporablja paketno časovno žigosanje, ki se izvaja v naslednjih korakih:

- za vse entitete, za katere je potrebno ustvariti dokazila o verodostojnosti se ustvarijo AIP;
- vse ustvarjene AIP strežnik IMiS®/ARChive Server obdela, ter se za njih ustvari prstne odtise;
- iz izračunanih prstnih odtisov se zgradi drevo prstnih odtisov (angl. Hash tree) – za časovno žigosanje strežnik uporablja Merklovo drevo (angl. Merkle tree) (glej poglavje 3.6.4.3 Merklovo drevo);
- iz Merklovega drevesa se izračuna korenski prstni odtis, ki ščiti celotno drevo prstnih odtisov;
- korenski prstni odtis se časovno žigosa, tako se ustvari dokazilo o obstoju vseh prstnih odtisov (ter pripadajočih AIP) v Merklovem drevesu pred časom, navedenim v časovnem žigu.

Z uporabo Merklovega drevesa lahko paketno žigosamo večje število entitet.
S tem racionaliziramo postopek ustvarjanja dokazil. Ko so dokazila enkrat ustvarjena, je potrebno poskrbeti za njihovo zanesljivost s podaljšanjem le-teh.



Slika 16: Postopek časovnega žigosanja z uporabo Merklovega drevesa

3.6.4.2 Postopek podaljševanja dokazil

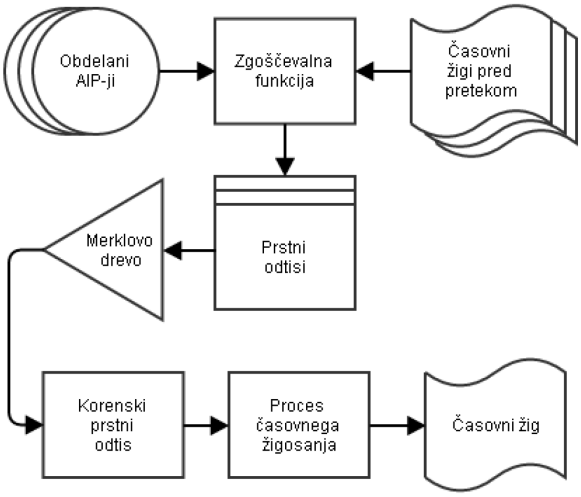
Vsa ustvarjena dokazila je potrebno pred izgubo zanesljivosti podaljšati, saj drugače ne morejo zagotavljati avtentičnosti dolgoročno arhiviranih podatkov.
Podaljševanje dokazil lahko ločimo na dva načina:

- enostavno podaljševanje
- kompleksno podaljševanje.

3.6.4.2.1 Enostavno podaljševanje

Enostavno podaljševanje se izvede v primeru, ko je digitalno potrdilo, ki je ustvarilo časovni žig pred iztekom. Tak časovni žig enostavno podaljšamo tako, da iz njega izračunamo prstni odtis ter ga časovno žigosamo. Tako nov časovni žig podaljša zanesljivost starega kljub temu, da bodisi digitalno potrdilo starega časovnega žiga preteče. Temu pravimo veriga časovnih žigov (angl. Timestamp chain).

Zaradi neracionalnosti podaljševanja posameznih časovnih žigov, se večinoma za podaljševanje uporablja Merklovo drevo po enakem postopku kot pri paketni obdelavi AIP.
Postopek enostavnega podaljševanja združimo s postopekom ustvarjanja dokazil.
Uporabimo Merklovo drevo, ki mu skozi postopek zgoščevanja dodamo prstne odtise AIP in časovne žige pred potekom veljavnosti.
S časovnim žigosanjem korenskega prstnega odtisa Merklovega drevesa tako ustvarimo dokazilo za obstoj AIP ter hkrati podaljšamo veljavnost časovnih žigov pred pretekom.



Slika 17: Postopek enostavnega podaljševanja v kombinaciji z ustvarjanjem dokazil

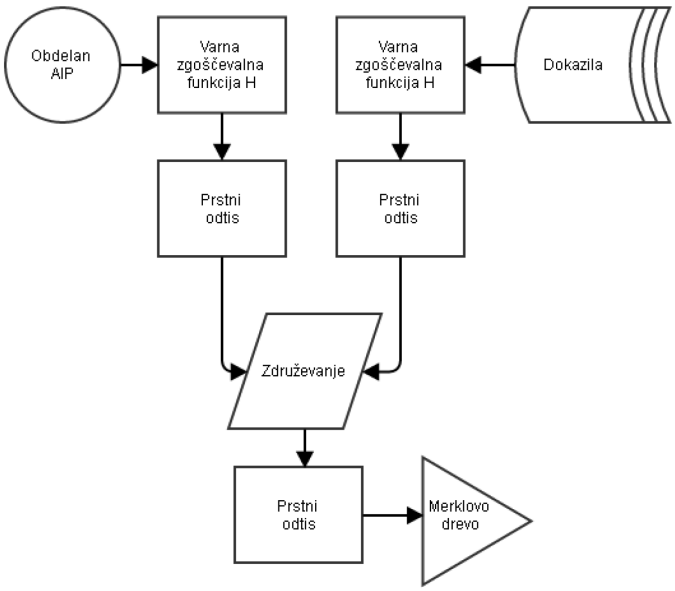
3.6.4.2.2 Kompleksno podaljševanje

Kompleksno podaljševanje se izvede takrat, ko se napove zmanjšanje varnosti zgoščevalnega algoritma, s katerim so bili narejeni prstni odtisi AIP.

V tem primeru je potrebno, preden zgoščevalni algoritem postane šibek izvesti podaljševanje po naslednjih korakih:

1. izberemo novo varno zgoščevalno funkcijo;
2. izračunamo nov prstni odtis za arhivski informacijski paket (AIP);
3. izračunamo nov prstni odtis iz vseh dokazil, ki pripadajo AIP iz točke 2.;
4. prstna odtisa iz točke 2 in 3 združimo, njun skupni prstni odtis pa dodamo v Merklovo drevo (glej poglavje 3.6.4.3 Merklovo drevo);
5. za vsak arhivski informacijski paket (AIP), za katerega je bil v postopku ustvarjanja dokazil uporabljen šibek algoritem je potrebno izvesti operacije, naštete v točkah 2, 3 in 4.;
6. iz Merklovega drevesa se izračuna korenski prstni odtis;
7. korenski prstni odtis se časovno žigosa.

S kompleksnim podaljševanjem zagotovimo zanesljivost AIP in vseh pripadajočih dokazil.
Z novim časovnim žigom začnemo novo verigo časovnih žigov, ki se jih podaljšuje z enostavnim podaljševanjem, dokler velja za varno nova zgoščevalna funkcija, uporabljena pri zadnjem kompleksnem podaljševanju.



Slika 18: Del postopeka kompleksnega podaljševanja, opisanega v točkah 2,3 in 4

3.6.4.3 Merklovo drevo

Merklovo drevo je drevesno urejena podatkovna struktura, pri katerem so vozlišča drevesa prstni odtisi zgoščevalne funkcije enakega tipa.

Lastnosti drevesa so:

- Vsebuje samo prstne odtise zgoščevalnih funkcij enakega tipa (drevo ne more vsebovati prstnih odtisov narejenih na primer z MD5 in SHA1 zgoščevalnim algoritmom).
- Prstni odtisi se v drevo dodajajo kot končna vozlišča (angl. Leaf node).
  Gre za vozlišča, brez podrejenih vozlišč.
- Dodajanje in odvzemanje prstnih odtisov je mogoče le, ko drevo še ni zgrajeno.
- Korenski prstni odtis je odtis, izračunan pri gradnji drevesa in ščiti celotno drevo.
- Zgrajeno drevo lahko zmanjšamo samo na vozlišča, ki so nujno potrebna za izračun korenskega prstnega odtisa iz posameznega končnega vozlišča.

Na strežniku IMiS®/ARChive Server je izvedeno Merklovo drevo z uporabo dvojiškega drevesa. Dvojiško drevo je podatkovna struktura z vozliščem, ki ima lahko največ dve podrejeni vozlišči.

3.6.4.3.1 Gradnja Merklovega drevesa

Gradnja drevesa je postopek gradnje vozlišč od končnih vozlišč do korenskega vozlišča.
Gradnja vozlišča ali skupine poteka na naslednji način:

- vzamemo dva prstna odtisa (vrednosti iz končnih vozlišč);
- binarne vrednosti prstnih odtisov razvrstimo po velikosti v naraščajočem vrstnem redu;
- razvrščene vrednosti združimo;
- iz združenih vrednosti prstnih odtisov izračunamo nov prstni odtis istega tipa,
  ki predstavlja novo vozlišče v drevesu.

Ta postopek ponavljamo dokler imamo na voljo več kot eno vozlišče. Ko nam ostane samo še eno vozlišče pomeni, da je drevo zgrajeno. Vozlišče, ki je ostalo pa predstavlja korenski prstni odtis.

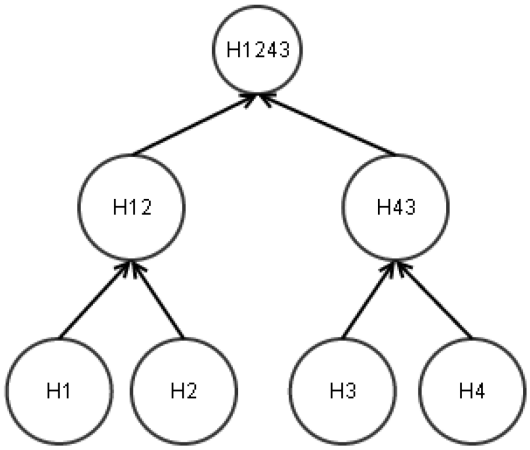
Primer: Merklovo drevo, ki ima štiri končna vozlišča s prstnimi odtisi, označena s H1, H2, H3 in H4. Predpostavimo, da za vrednosti prstnih odtisov veljajo naslednje relacije:

- *binarna vrednost prstnega odtisa H1 je manjša od H2 (H1 < H2)*
- *binarna vrednost prstnega odtisa H3 je večja od H4 (H3 > H4).*

Na podlagi teh relacij izračunamo novi vozlišči H12 in H43 po naslednjih formulah:

- *H12 = HASH(H1 || H2) – ker velja H1 < H2, sortiranje ohrani vrstni red, zato oba prstna odtisa samo združimo (operacija ||).*
- *H43 = HASH(H4 || H3) – ker je H3 večji kot H4, sortiranje obrne vrstni red prstnih odtisov.*

Za vozlišči H12 in H43 predpostavimo, da velja naslednja relacija: H12 < H43.
Tako iz teh dveh vozlišč izračunamo zadnje vozlišče: H1243 = HASH (H12 || H43).
Ker je H1243 zadnje vozlišče v drevesu pomeni, da je drevo zgrajeno, vrednost vozlišča pa je korenski prstni odtis.



Slika 19: Primer Merklovega drevesa

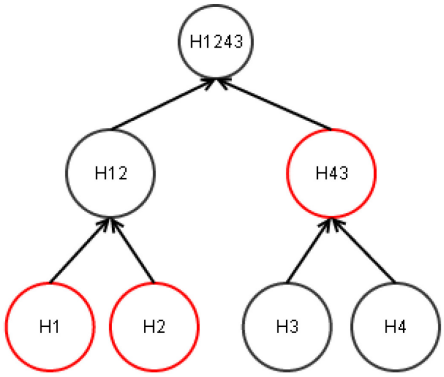
3.6.4.3.2 Reducirano Merklovo drevo

Zgrajeno Merklovo drevo lahko zreduciramo na število vozlišč, ki so nujno potrebna za izračun korenskega prstnega odtisa.
Reduciranje poteka na naslednji način:

- Izberemo končno vozlišče, za katerega hočemo reducirano Merklovo drevo.
- Za izbrano vozlišče pogledamo njegovo neposredno nadrejeno vozlišče in za to vozlišče vzamemo vsa neposredna podrejena vozlišča (kamor spada tudi na začetku izbrano vozlišče). Vrednosti prstnih odtisov v vozliščih razvrstimo naraščajoče (enako kot pri gradnji vozlišča).
- Iskanje nadaljujemo za vsa neposredno nadrejena vozlišča, dokler jih ne zmanjka (kar pomeni, da smo prišli do korenskega vozlišča). Vsa dobljena vozlišča razvrstimo naraščajoče.
- Neposredno nadrejenih vozlišč ne vključujemo v reducirano drevo, saj se jih da izračunati po pravilu gradnje Merklovega drevesa.

Primer: Reducirano Merklovo drevo, ki smo ga uporabili pri gradnji drevesa.
Izberemo končno vozlišče H1. Neposredno nadrejeno vozlišče je H12, njegovi podrejeni vozlišči pa sta H1 in H2. Obe vozlišči vzamemo ter ju razvrstimo po velikosti njihovih prstnih odtisov (velja relacija H1 < H2).
Vozlišču H12 je neposredno nadrejeno vozlišče H1243, njegovi podrejeni vozlišči pa sta H12 in H43.
Ker se da vozlišče H12 izračunati iz vozlišč H1 in H2 ga izpustimo in v reducirano drevo vključimo samo vozlišče H43. Vozlišče H1243 nima neposredno nadrejenega vozlišča, zato se reduciranje drevesa za vozlišče H1 konča.

Reducirano drevo tako vsebuje vozliča v naslednjem vrstnem redu: H1, H2, H43.
Kot je razvidno iz spodnje slike, je reducirano drevo za vozlišče H2 enako, saj imata H1 in H2 isto nadrejeno vozlišče H12.



Slika 20: Primer reduciranega drevesa za vozlišči H1 in H2

Primer izračuna korenskega prstnega odtisa:
Korenski prstni odtis se iz reduciranega drevesa izračuna na naslednji način:
H1243 = HASH(HASH(H1 || H2) || H43).
Postopek izračuna je naslednji:

- *H12 = HASH(H1 || H2) – izračun vmesnega vozlišča H12;*
- *SORT(H12, H43) = [H12, H43] – rezultat sortiranja H12 in H43 (H12 < H43);*
- *H1243 = HASH(H12 || H43) – izračun korenskega prstne odtisa.*

Vzemimo za primer še reducirano drevo za vozlišči H3 in H4, ki vsebuje vozlišča v naslednjem vrstnem redu: H4, H3, H12.
Korenski prstni odtis izračunamo na naslednji način:
H1243 = HASH(H12 || HASH(H4 || H3)).

Postopek je naslednji:

- *H43 = HASH(H4 || H3) – izračun vmesnega vozlišča H43;*
- *SORT(H43, H12) = [H12, H43] – rezultat sortiranja H43 in H12 (H12 < H43);*
- *H1243 = HASH(H12 || H43) – izračun korenskega prstnega odtisa.*

3.6.4.4 Sintaksa

Kot je bilo že omenjeno ima strežnik IMiS®/ARChive Server izveden ERS v XML obliki.

3.6.4.4.1 Etiketa »EvidenceRecord«

Korenski element XML je etiketa »EvidenceRecord«, ki vsebuje elemente iz spodnje tabele.

Ime XML elementa	Tip XML elementa	Obveznost
Version	Atribut	DA
ArchiveTimeStampSequence	Etiketa	DA

Tabela 10: XML elementi etikete »EvidenceRecord«

Atribut »Version«

Vrednost atributa je fiksna vrednost »1.0« in predstavlja verzijo ERS.

Etiketa »ArchiveTimeStampSequence«

Vsebina etikete je zaporedje vseh verig arhivskih časovnih žigov (angl. Archive Timestamp chain), ki prikazuje ustvarjanje in podaljševanje ustvarjenih dokazil za avtentičnost dolgoročno hranjenih entitet (glej poglavje 3.6.4.1 Postopek ustvarjanja dokazil in poglavje 3.6.4.2 Postopek podaljševanja dokazil).

Vsako zaporedje časovnih žigov je označeno z etiketo »ArchiveTimeStampChain«.

Primer: XML zapis, ki predstavlja ERS z dvema verigama časovnih žigov.

```
<EvidenceRecord xmlns="urn:ietf:params:xml:ns:ers" version="1.0">
<ArchiveTimeStampSequence>
<ArchiveTimeStampChain order="1">...</ArchiveTimeStampChain>
<ArchiveTimeStampChain order="2">...<ArchiveTimeStampChain>
</ArchiveTimeStampSequence>
</EvidenceRecord>
```

3.6.4.4.2 Etiketa »ArchiveTimeStampChain«

Etiketa vsebuje verigo arhivskih časovnih žigov, ki si bili ustvarjeni v postopku ustvarjanja in postopku podaljševanja z istim tipom zgoščevalne funkcije.

Elementi etikete so opisani v spodnji tabeli.

Ime XML elementa	Tip XML elementa	Obveznost
Order	Atribut	DA
DigestMethod	Etiketa	NE
CanonicalizationMethod	Etiketa	DA
ArchiveTimeStamp	Etiketa	DA

Tabela 11: XML elementi etikete »ArchiveTimeStampChain«

Atribut »Order«

Vrednost atributa predstavlja vrstni red zaporedja verig časovnih žigov.
Pri ustvarjanju prvega arhivskega časovnega žiga, se začne graditi veriga časovnih žigov z vrednostjo atributa »1«. V to verigo se nato dodajajo vsi arhivski časovni žigi,
ki so bili ustvarjeni z enostavnim podaljševanjem.
Za vsak postopek kompleksnega podaljševanja se ustvari nova veriga arhivskih časovnih žigov, z vrednostjo atributa »2«, »3« itn. (glej poglavje 3.6.4.5 Enostavno podaljševanje in poglavje 3.6.4.6 Kompleksno podaljševanje).

Etiketa »DigestMethod«

Etiketa vsebuje obvezen atribut »Algorithm«. Vrednost atributa je URI,
ki predpisuje zgoščevalni algoritem uporabljen v verigi. Etiketa ni obvezna, saj se informacija o zgoščevalnem algoritmu nahaja tudi v posameznem časovnem žigu.

Etiketa »CanonicalizationMethod«

Etiketa vsebuje atribut »Algorithm«. Njegova vrednost je enotni označevalnik vira (angl. Uniform resource identifier – URI), ki predpisuje algoritem za pretvorbo verige v normalizirano.

3.6.4.4.2.1 Etiketa »ArchiveTimeStamp«

Arhivski časovni žig vsebuje časovni žig (z vsemi pripadajočimi podatki, ki so potrebni za preverjanje le-tega) in reducirano Merklovo drevo, če se je časovno žigosalo večje število AIP.
Elementi etikete so opisani v spodnji tabeli.

Primer: XML zapis prikazuje verigo, ki vsebuje tri arhivske časovne žige, generirane z zgoščevalnim algoritmom SHA1.

```
<ArchiveTimeStampChain order="1">
<DigestMethod algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
<CanonicalizationMethod algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
<ArchiveTimeStamp order="1">...</ArchiveTimeStamp>
<ArchiveTimeStamp order="2">...</ArchiveTimeStamp>
<ArchiveTimeStamp order="3">...</ArchiveTimeStamp>
</ArchiveTimeStampChain>
```

Ime XML elementa	Tip XML elementa	Obveznost
Order	Atribut	DA
HashTree	Etiketa	NE
TimeStamp	Etiketa	DA

Tabela 12: XML elementi etikete »ArchiveTimeStamp«

Atribut »Order«

Vrednost atributa predstavlja vrstni red arhivskega časovnega žiga znotraj verige. Arhivski časovni žig z vrednostjo »1« je bil tako ustvarjen v postopku ustvarjanja dokazil, vsi ostali arhivski časovni žigi pa v postopku enostavnega podaljševanja.

Etiketa »HashTree«

V primeru, da je etiketa prisotna, le-ta predstavlja reducirano Merklovo drevo, nad katerim je bil narejen pripadajoči arhivski časovni žig. V primeru, da etiketa ni prisotna, arhivski časovni žig pripada objektu, ki ima prstni odtis v časovnem žigu znotraj etikete »TimeStamp«.

Etiketa predstavlja reducirano Merklovo drevo. Vsebuje zaporedje elementov »Sequence«, ki predstavljajo nivoje Merklovega drevesa. Prvi nivo (nivo 0) ni vključen, saj se vrednost korenskega vozlišča lahko izračuna.

Elementi etikete »Sequence« so opisani v spodnji tabeli.

Ime XML elementa	Tip XML elementa	Obveznost
Order	Atribut	DA
DigestValue	Etiketa	DA

Tabela 13: XML elementi etikete »Sequence«

Atribut »Order«

Vrednost atributa predstavlja nivo reduciranega Merklovega drevesa v obratnem vrstnem redu, kot je določena globina drevesa (glej poglavje 3.6.4.3 Merklovo drevo).
Prva veljavna vrednost atributa je »1«, ki predstavlja prstne odtise končnih vozlišč. Vrednost »2« bi tako predstavljala naslednji nivo, itn.

Etiketa »DigestValue«

Vrednost etikete vsebuje Base64 kodiran prstni odtis.

Primer: XML zapis prikazuje reducirano Merklovo drevo z dvema nivojema.
Prvi nivo (order 1) vsebuje Base64 kodirane prstne odtise končnih vozlišč,
drugi nivo pa vsebuje prstni odtis vozlišča, ki omogoča izračun prstnega odtisa korenskega vozlišča.

```
<HashTree>
<Sequence order="1">
<DigestValue>D+/oVEs6CjRHi3UNL1vk4WcsEkA=</DigestValue>
<DigestValue>EBEhfDzZh9+rfb1Kaqe65o7TTok=</DigestValue>
</Sequence>
<Sequence order="2">
<DigestValue>fBuhe8txb0OmNt27uvYupJTrgBQ=</DigestValue>
</Sequence>
</HashTree>
```

Etiketa »TimeStamp«

Etiketa vsebuje časovni žig in seznam kriptografskih elementov (celotno vejo digitalnih potrdil in informacije o preklicih digitalnih potrdil), ki so potrebni za preverjanje časovnega žiga.
S temi podatki lahko nedvoumno dokažemo veljavnost časovnega žiga v času, ko je bil narejen in posledično avtentičnost vsebine, ki jo pokriva.

Ime XML elementa	Tip XML elementa	Obveznost
TimeStampToken	Etiketa	DA
CryptographicInformationList	Etiketa	NE

Tabela 14: XML elementi etikete »TimeStamp«

Etiketa »TimeStampToken«

Etiketa vsebuje časovni žig pridobljen od izdajatelja varnih časovnih žigov.

Ima obvezen atribut »Type«, ki določa tip časovnega žiga.

Tip	Opis
»XMLENTRUST«	Časovni žig je v formatu XML, kot ga predpisuje W3C konzorcij (www.w3.org/TR/xmldsig-core/) in je kar vsebina TimeStampToken.
»RFC3161«	Časovni žig je narejen po standardu RFC3161. Vsebina TimeStampToken je Base64 kodirana zaradi kompatibilnosti z XML-jem.

Tabela 15: Podprti tipi časovnih žigov

Etiketa »CryptographicInformationList«

Če etiketa je, vsebuje kriptografske elemente, ki omogočajo preverjanje verodostojnosti časovnega žiga. Če etikete ni, potem te elemente vsebuje že sam časovni žig.

Etiketa ima obvezna atributa »Order« in »Type«, ki določa tip kriptografskega elementa.

Tip	Opis
»CERT«	Kriptografski element je Base64 kodirano X509 digitalno potrdilo v binarni obliki.
»CRL«	Kriptografski element je Base64 kodirana lista preklicanih digitalnih potrdil v binarni obliki.
»OCSP«	Kriptografski element je Base64 kodiran odgovor strežnika po protokolu za sprotno preverjanje statusa elektronskega potrdila v binarni obliki.

Tabela 16: Podprti tipi kriptografskih elementov

Naslednji primer prikazuje prvi arhivski časovni žig, ki vsebuje reducirano Merklovo drevo, časovni žig v XML formatu in pripadajoč seznam kriptografskih elementov.

Prvi v seznamu je korensko digitalno potrdilo, ki skupaj z digitalnim potrdilom iz časovnega žiga tvori vejo digitalnih potrdil. Drugi element je lista pretečenih digitalnih potrdil, s katerim lahko preverimo, da digitalno potrdilo, ki je generiralo časovni žig ni pretečeno.

```
<ArchiveTimeStamp order="1">
<HashTree>
<Sequence order="1">
<DigestValue>D+/oVEs6CjRHi3UNL1vk4WcsEkA=</DigestValue>
<DigestValue>EBEhfDzZh9+rfb1Kaqe65o7TTok=</DigestValue>
</Sequence>
<Sequence order="2">
<DigestValue>fBuhe8txb0OmNt27uvYupJTrgBQ=</DigestValue>
</Sequence>
</HashTree>
<TimeStamp>
<TimeStampToken type="XMLENTRUST">
<dsig:Signature xmlns:dsig="http://www.w3.org/2000/09/xmldsig#">...
</dsig:Signature>
</TimeStampToken>
<CryptographicInformationList>
<CryptographicInformation order="1" type="CERT">MIIEHDCCAwS...
</CryptographicInformation>
<CryptographicInformation order="2" type="CRL">MIISKTCCEREC...
</CryptographicInformation>
</CryptographicInformationList>
</TimeStamp>
</ArchiveTimeStamp>
```

3.6.4.5 Enostavno podaljševanje

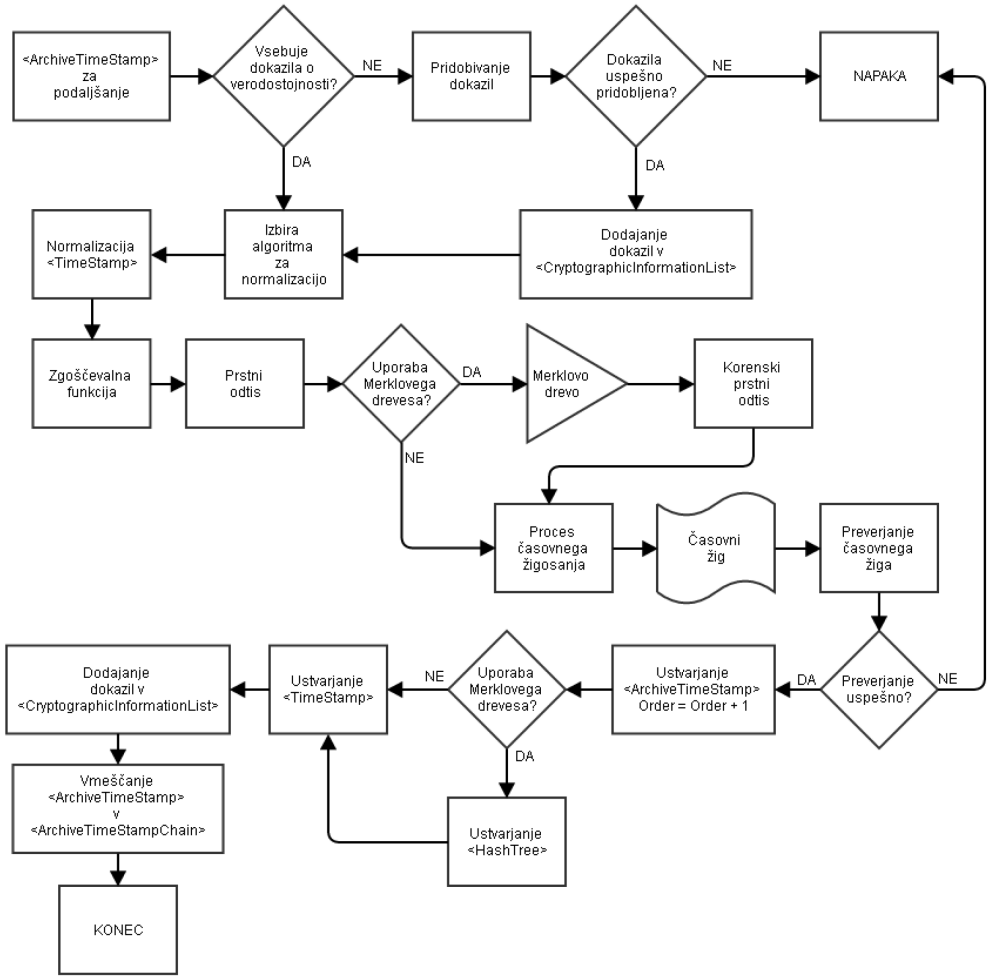
Pri enostavnem podaljševanju se vedno podaljša zadnji arhivski časovni žig v zadnji verigi arhivskih časovnih žigov.

Postopek je naslednji:

1. Preveri se, če zadnji arhivski časovni žig vsebuje vse potrebne kriptografske elemente, ki dokazujejo verodostojnost žiga (certifikate, informacije o preklicanih elektronskih potrdilih).V primeru, da jih ne vsebujejo, jih strežnik samodejno pridobi in doda v pripadajoč seznam kriptografskih elementov.
   Če se dokazil ne more pridobiti, nadaljevanje postopka ni mogoče.
2. Na podlagi etikete »CanonicalizationMethod« iz verige časovnih žigov izberemo algoritem za pretvorbo v normalizirano obliko.
3. Iz zadnjega arhivskega časovnega žiga vzamemo etiketo »TimeStamp« in jo normaliziramo s prej izbranim algoritmom.
4. Z izbrano zgoščevalno funkcijo (določa jo etiketa »DigestMethod« v trenutni verigi ali pa je vsebovana v časovnem žigu, ki ga podaljšujemo) izračunamo prstni odtis normaliziranih podatkov. Tega časovno žigosamo (ali pa dodamo v Merklovo drevo v primeru podaljševanja več časovnih žigov ali kombinacije z ustvarjanjem dokazil) (glej poglavje 3.6.4.2 Postopek podaljševanja dokazil).

5. Preverimo časovni žig (glej poglavje 3.6.6.2 Preverjanja časovnega žiga).
6. Če je preverjanje časovnega žiga neuspešno, je potrebno celoten postopek podaljševanja ponoviti, sicer ne moremo zagotoviti verodostojnosti novega časovnega žiga.
7. Če je preverjanje uspešno, potem ustvarimo novo etiketo »ArchiveTimeStamp« z vrednostjo atributa »Order«, ki je za 1 večje od arhivskega časovnega žiga, ki smo ga podaljšali. Če smo arhivski časovno žig podaljševali s pomočjo Merklovega drevesa, v novi časovni žig vključimo reducirano Merklovo drevo, ki mu pripada. Na koncu ustvarimo etiketo »TimeStamp«, kamor dodamo preverjen časovni žig s seznamom kriptografskih elementov, ki smo jih uporabili v postopku preverjanja.
   Nov arhivski časovni žig dodamo v zadnjo verigo časovnih žigov.

Spodnja slika prikazuje delovanje algoritma za enostavno podaljševanje arhivskih časovnih žigov.



Slika 21: Enostavno podaljševanje arhivskega časovnega žiga

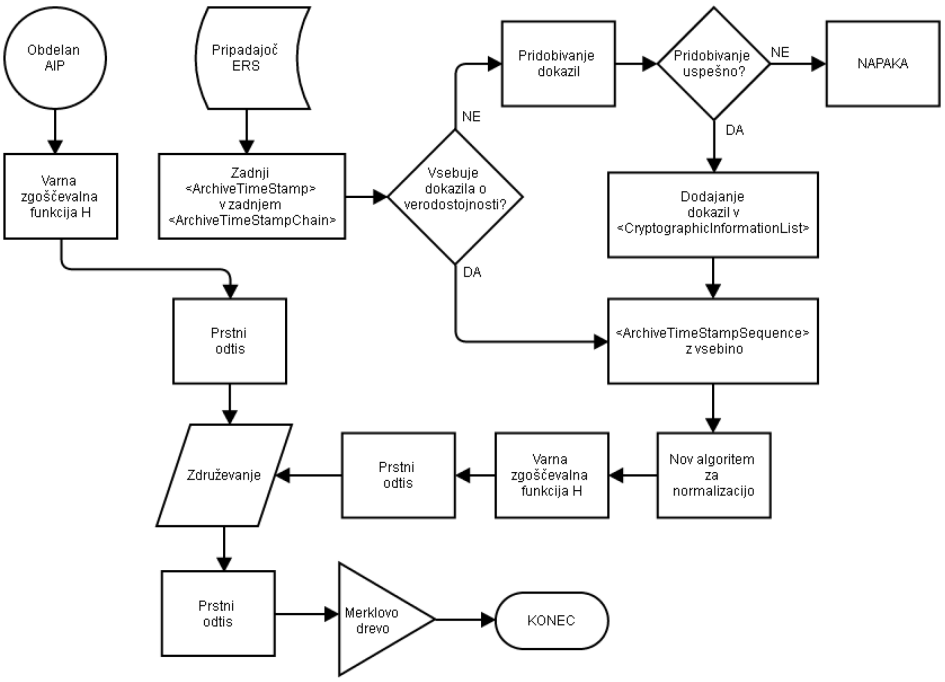
3.6.4.6 Kompleksno podaljševanje

Kompleksno podaljševanje se izvede ob menjavi šibkega zgoščevalnega algoritma za računanje prstnih odtisov z močnejšim. V tem primeru je potrebno kompleksno podaljšati vsa dokazila, ki so bila narejena z šibkejšim algoritmom za vse AIP, ustvarjene v postopku dolgoročne hrambe.

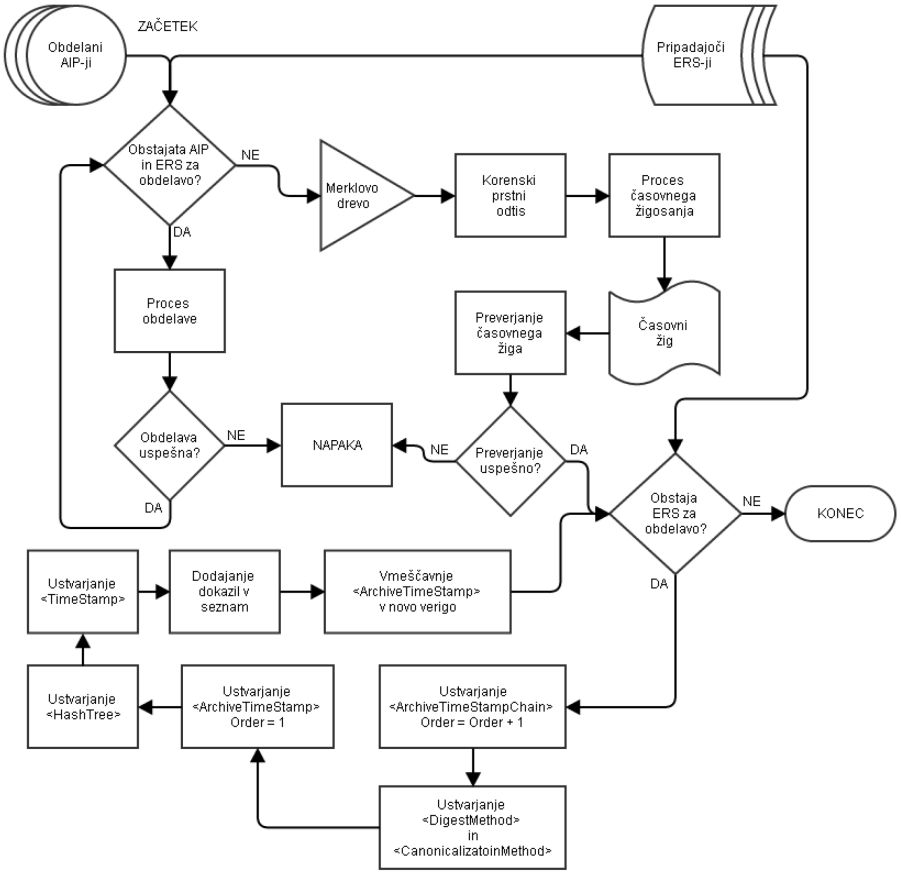
Postopek podaljševanja je naslednji:

1. Izberemo nov močan zgoščevalni algoritem.
2. Izberemo nov algoritem za pretvorbo v normalizirano obliko.
3. Z novim zgoščevalnim algoritmom izračunamo prstni odtis obdelanega AIP.
4. Za pripadajoči ERS preverimo, ali zadnji arhivski časovni žig v trenutni verigi časovnih žigov vsebuje vse potrebne kriptografske elemente za dokazovanje verodostojnosti. Če jih ne vsebuje, potem se jih pridobi in doda v pripadajoč seznam.
5. Če se dokazil ne more pridobiti, nadaljevanje postopka ni mogoče.
6. Iz pripadajočega ERS vzamemo etiketo »ArchiveTimeStampSequence« z njeno vsebino in jo pretvorimo v normalizirano obliko z novim algoritmom za pretvorbo.
7. Za normalizirano vsebino izračunamo prstni odtis z novim zgoščevalnim algoritmom.
8. Prstna odtisa, pridobljena iz obdelanega AIP in normalizirane vsebine združimo (njeni binarni vrednosti razvrstimo v naraščajočem vrstnem redu in ju zlepimo). Z novim zgoščevalnim algoritmom izračunamo prstni odtis in ga dodamo v Merklovo drevo.
9. Postopek iz točk 3. – 7. ponavljamo za vse AIP (in pripadajoče ERS), ki morajo biti kompleksno podaljšani.
10. Preverimo časovni žig. Če je preverjanje neuspešno potem postopka ne moremo nadaljevati.
11. V etiketi »ArchiveTimeStampSequence« izdelamo novo etiketo »ArchiveTimeStampChain« z atributom »Order«. Ta ima vrednost za 1 večjo kot zadnja etiketa »ArchiveTimeStampChain«.
12. Znotraj nove etikete »ArchiveTimeStampChain« zapišemo nov uporabljen zgoščevalni algoritem ter algoritem za pretvorbo v normalizirano obliko (etiketi »DigestMethod« in »CanonicalizationMethod«). Izdelamo novo etiketo »ArchiveTimeStamp« z vrednostjo atributa »Order« 1. Tako začnemo graditi novo verigo arhivskih časovnih žigov.
13. V »ArchiveTimeStamp« vključimo reducirano Merklovo drevo za združen prstni odtis AIP in ERS. Ustvarimo etiketo »TimeStamp«, kamor dodamo preverjen časovni žig s seznamom kriptografskih elementov, ki smo jih uporabili v postopku preverjanja.
14. Postopek iz točk 10. – 12. ponavljamo za vse ERS, ki so bili vključeni v postopek kompleksnega podaljševanja.

Kompleksno podaljševanje z Merklovim drevesom prikazujeta spodnji sliki.



Slika 22: Postopek obdelave AIP in pripadajočega ERS



Slika 23: Postopek kompleksnega podaljševanja z Merklovim drevesom

3.6.4.7 Verifikacija

Verifikacija ERS je postopek v katerem preverimo avtentičnost vseh verig arhivskih časovnih žigov. S tem dokažemo avtentičnost AIP, ki ga ščiti ERS.

V osnovi lahko verifikacijo delimo na:

- verifikacija začetne verige časovnih žigov
- verifikacija ostalih verig časovnih žigov.

3.6.4.7.1 Verifikacija začetne verige

Verifikacija začetne verige poteka na naslednji način:

- Iz verige pridobimo algoritem zgoščevalne funkcije in algoritem za normalizacijo (etiketi »DigestMethod in »CanonicalizationMethod«). Če etiketa »DigestMethod« ne obstaja, pridobimo podatke iz prvega arhivskega časovnega žiga verige - TimeStampToken).
- S pridobljenim zgoščevalnim algoritmom izračunamo prstni odtis pripadajočega obdelanega AIP.
- Če prvi arhivski časovni žig vsebuje reducirano Merklovo drevo, potem se mora izračunani prstni odtis AIP nahajati v prvi etiketi »Sequence« znotraj etikete »HashTree«. Iz reduciranega drevesa izračunamo korenski prstni odtis, ki se mora ujemati z prstnim odtisom v časovnem žigu.
- Če reduciranega Merklovega drevesa ni, potem se mora izračunani prstni odtis AIP ujemati s prstnim odtisom v časovnem žigu.
- Preverimo digitalno potrdilo časovnega žiga v času, ko je ta nastal (s pripadajočimi kriptografskimi objekti).
- Za vse naslednje arhivske časovne žige v verigi naredimo enako verifikacijo.
  Namesto prstnega odtisa AIP vzamemo predhodni arhivski časovni žig,
  ga normaliziramo, na normaliziranih podatkih izračunamo prstni odtis ter vrednosti prstnih odtisov preverimo v reduciranem Merklovem drevesu ali v časovnem žigu (glej prejšnje točke).

3.6.4.7.2 Verifikacija ostalih verig

Verifikacija verig, ki niso začetne poteka na naslednji način:

- Iz verige pridobimo algoritem zgoščevalne funkcije in algoritem za normalizacijo (etiketi »DigestMethod in »CanonicalizationMethod«). Če »DigestMethod« ne obstaja pridobimo podatke iz prvega arhivskega časovnega žiga verige – TimeStampToken.
- S pridobljenim zgoščevalnim algoritmom izračunamo prstni odtis pripadajočega obdelanega AIP.
- Iz »ArchiveTimeStampSequence« odstranimo trenutno verigo in vse naslednje verige. Tako ostanejo v »ArchiveTimeStampSequence« samo predhodne verige, ki so osnova za normaliziranje. Kot je opisano v kompleksnem podaljševanju, normaliziramo celoten »ArchiveTimeStampSequence« in z zgoščevalno funkcijo izračunamo prstni odtis normaliziranih podatkov.
- Prstna odtisa AIP in »ArchiveTimeStampSequence« združimo (razvrstimo naraščajoče ter zlepimo) in izračunamo skupni prstni odtis.
- Če prvi arhivski časovni žig vsebuje reducirano Merklovo drevo, potem se mora skupni prstni odtis nahajati v prvi etiketi »Sequence« znotraj etikete »HashTree«. Iz reduciranega drevesa izračunamo korenski prstni odtis, ki se mora ujemati z prstnim odtisom v časovnem žigu.
- Če reduciranega Merklovega drevesa ni, potem se mora skupni prstni odtis ujemati s prstnim odtisom v časovnem žigu.
- Verifikacija ostalih arhivskih časovnih žigov poteka po enakem postopku, kot je opisan v verifikaciji začetne verige časovnih žigov.

Po uspešni verifikaciji vseh verig po zgoraj opisanih postopkih lahko rečemo, da se vsebina AIP ni spremenila skozi čas (od njenega nastanka do trenutka verifikacije). Če se podatki v AIP ujemajo s podatki v entiteti potem lahko zagotovimo avtentičnost arhivirane entitete.
Če je verifikacija v katerikoli točki neuspešna, potem ne moremo zagotoviti avtentičnosti arhivirane entitete.

3.6.5 AIP

Arhivski informacijski paket (angl. Archival Information Package - AIP) je povzetek metapodatkov in vsebine entitete, ki je predmet zaščite postopkov zagotavljanja avtentičnosti, zbrane v XML datoteki.

AIP potrebujemo za grupiranje metapodatkov in vsebine entitete v arhivski informacijski paket, ki predstavlja vsebino za dolgoročno hrambo podatkov (glej poglavje 3.6.1 Predpogoji).

AIP se izdela v naslednjih primerih:

- če lastnosti entitete ustrezajo vsaj enemu pravilu;
- ko ima zaprta entiteta vsaj en metapodatek ali vsebino, označeno za vključitev v arhivski informacijski paket.

Za več informacij glej poglavje 3.6.1 Predpogoji in poglavje 3.3.4 Predloge.
AIP se shranjuje v podatkovno bazo strežnika IMiS®/ARChive Server. Dostopen je preko odjemalca IMiS®/Client, ki lahko AIP po potrebi pridobi (skupaj z ERS), tudi pri izvozu.
AIP zapis je sestavljen iz naslednjih sekcij:

- glava
- metapodatki
- digitalna potrdila
- informacije o preklicih digitalnih potrdil.

3.6.5.1 Glava

V sekcijo »Glava« so uvrščeni vsi potrebni podatki za pravilno obdelavo in interpretacijo arhivskega informacijskega paketa. Glava je obvezen del AIP.
Etiketa »Header« vsebuje elemente navedene v spodnji tabeli.

Ime XML elementa	Tip XML elementa	Obveznost
Version	Atribut	DA
CanonicalizationMethod	Etiketa	DA

Tabela 17: XML elementi etikete »Header«

Atribut »Version«

Atribut »Version« je ne-predznačeno 32-bitno število, ki predstavlja verzijo arhivskega informacijskega paketa. Verzija predpisuje način, kako naj se AIP obdeluje in interpretira pri preverjanju avtentičnosti entitete.

Vrednosti atributa »Version« in njihov pomen so opisane v spodnji tabeli.

Vrednost atributa	Interpretacija AIP
»1«	Arhivski informacijski paket verzije »1« se obravnava kot celota. Na podlagi vrednosti etikete »CanonicalizationMethod« se izbere predpisano kanonikalizacijsko metodo, ki celoten XML pretvori v normalizirano obliko. Taka oblika je osnova za obdelavo (računanje prstnega odtisa, ...) in preverjanje avtentičnosti vsebine entitete in njenih metapodatkov.

Tabela 18: Interpretacija AIP v odvisnosti od vrednosti atributa »Version«

Etiketa »CanonicalizationMethod«

Etiketa vsebuje atribut »Algorithm«. Njegova vrednost je enotni označevalnik vira (angl. Uniform Resource Identifier – URI), ki predpisuje algoritem za pretvorbo AIP v normalizirano obliko.

W3C konzorcij predpisuje naslednje URI vrednosti:

- http://www.w3.org/TR/2001/REC-xml-c14n-20010315 (algoritem verzije 1.0).
- http://www.w3.org/TR/2002/REC-xml-exc-c14n-20020718 (ekskluzivna verzija algoritma 1.0).
- http://www.w3.org/TR/2008/REC-xml-c14n11-20080502 (algoritem verzije 1.1).

Primer: XML zapis prikazuje AIP verzije 1, ki uporablja algoritem verzije 1.0 za pretvorbo v normalizirano obliko.

```
<aip:Header version="1">
<ds:CanonicalizationMethod algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
</aip:Header>
```

3.6.5.2 Metapodatki

V sekcijo »Metapodatki« so uvrščeni metapodatki entitete, ki so v predlogah nastavljeni za vključitev v arhivski informacijski paket (glej poglavje 3.2.3 Povezava s predlogami) ne glede na to, ali vsebujejo kakšno vrednost ali ne.

Sekcija je sestavljena iz zaporedja »Attribute« etiket, ki vsebujejo elemente navedene v spodnji tabeli.

Ime XML elementa	Tip XML elementa	Obveznost
Id	Atribut	DA
Type	Atribut	DA
Value	Etiketa	NE

Tabela 19: XML elementi etikete »Attribute«

Atribut »Id«

Vrednost atributa »Id« je naziv metapodatka.

Atribut »Type«

Vrednost atributa »Type« predstavlja tip metapodatka (glej poglavje 3.2.1 Vrste atributov).

Etiketa »Value«

Vsebina etikete »Value« predstavlja vrednost atributa oziroma vrednosti, v primeru, da gre za atribut z več vrednostmi. Vsebina je lahko enostavna (prisotna je samo vrednost atributa ali prstni odtis vrednosti) ali kompleksna (poleg vsebine ali prstnega odtisa so prisotni še podatki o digitalnih podpisih vsebine).

V primeru ko je za vrednost prisoten prstni odtis vsebinem potem etiketa vsebuje še etiketo »Digest« z elementi, ki so opisani v naslednji tabeli.

Ime XML elementa	Tip XML elementa	Obveznost
DigestMethod	Etiketa	DA
DigestValue	Etiketa	DA

Tabela 20: XML elementi etikete »Digest«

Etiketa »DigestMethod«

Etiketa vsebuje atribut »Algorithm«, ki je obvezen in katerega vrednost je URI. Ta predpisuje algoritem za izračun prstnega odtisa vsebine. Podprti algoritmi so našteti v spodnji tabeli, ki predstavlja podprte algoritme za izračun prstnih odtisov in pripadajoče URI, določene pri W3C konzorciju (http://www.w3.org/TR/2013/NOTE-xmlsec-algorithms-20130124) ter v specifikaciji RFC 4051 http://www.ietf.org/rfc/rfc4051.txt).

Algoritem	URI
MD5	http://www.w3.org/2001/04/xmldsig-more#md5
SHA1	http://www.w3.org/2000/09/xmldsig#sha1
SHA224	http://www.w3.org/2001/04/xmldsig-more#sha224
SHA256	http://www.w3.org/2001/04/xmlenc#sha256
SHA384	http://www.w3.org/2001/04/xmldsig-more#sha384
SHA512	http://www.w3.org/2001/04/xmlenc#sha512

Tabela 21: Algoritmi in pripadajoči URI

Etiketa »DigestValue«

Etiketa predstavlja Base64 kodirano vrednost digitalnega prstnega odtisa, narejenega z algoritmom, ki je zapisan v atributu »Algorithm« v etiketi »DigestMethod«.

Primer: enostavna vsebina metapodatkov entitete, kjer je naveden avtor in njegov naslov.

- *metapodatek »Author« je tipa »String50«, »Client Address1« in »Client Address2« pa sta tipa »String100«, MultiValue;*
- *metapodatek »Client Address2« nima vrednosti, a je vseeno vključen v arhivski informacijski paket.*

```
<aip:Attribute id="Author" type="22">
<aip:Value>Janez Novak</aip:Value>
</aip:Attribute>
<aip:Attribute id="Client Address1" type="23">
<aip:Value>Brnciceva 41g</aip:Value>
<aip:Value>1231 Ljubljana</aip:Value>
</aip:Attribute>
<aip:Attribute id="Client Address2" type="23"/>
```

Primer: enostavna vsebina metapodatkov entitete, kjer se za vrednost uporablja digitalni prstni odtis vsebine.

```
<aip:Attribute Id="sys:Content" type="42">
    <aip:Value>
    <aip:Digest>
        <ds:DigestMethod algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
        <ds:DigestValue>xcwyBBmUzlLqoHNSt+09lGdmCiw=</ds:DigestValue>
        </aip:Digest>
    </aip:Value>
</aip:Attribute>
```

Kompleksna vsebina vsebuje še podatke o digitalnih podpisih vsebine. Digitalni podpis je vsebovan v etiketi »Signature« in je kodiran v Base64. Etiketa vsebuje elemente, ki so opisani v spodnji tabeli.

Ime XML elementa	Tip XML elementa	Obveznost
Version	Atribut	DA
Type	Atribut	DA

Tabela 22: XML elementi etikete »Signature«

Atribut »Version«

Vrednost etikete predstavlja verzijo digitalnega podpisa.

Atribut »Type«

Vrednost etikete predstavlja tip digitalnega podpisa. Trenutno podprt format je »XMLDSIG«.

Naslednji primer prikazuje kompleksno vsebino, kjer se poleg digitalnega prstnega odtisa nahajata še dva digitalna podpisa vsebine.

```
<aip:Attribute Id="sys:Content" type="42">
    <aip:Value>
        <aip:Signature Version="1" Type="XMLDSIG">ajM5dHoglGlqZ...aip:Signature>
        <aip:Signature Version="1" Type="XMLDSIG">Ym52ODMgI...</aip:Signature>
        <aip:Digest>
            <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
            <ds:DigestValue>xQmFgsuudE4gOkT3Hg/QB65caiQ=</ds:DigestValue>
        </aip:Digest>
    </aip:Value>
</aip:Attribute>
```

3.6.5.3 Digitalna potrdila

V to sekcijo so uvrščene celotne veje digitalnih potrdil, ki pripadajo elektronskim podpisom in so Base64 kodirana. Sekcija je sestavljena iz etiket »Certificate«, ki vsebuje obvezen atribut »Type«, ki predstavlja tip digitalnega potrdila. Trenutno podprt format je »X509DER«. Naslednji zapis prikazuje primer dveh digitalnih potrdil.

```
<aip:Certificate Type="X509DER">bmlzOHYg...</aip:Certificate>
<aip:Certificate Type="X509DER">Ym4zdmJk...</aip:Certificate>
```

3.6.5.4 Informacije o preklicih digitalnih potrdil

Poleg preverjanja časovne veljavnosti digitalnih potrdil so ključnega pomena tudi informacije o preklicih.

Digitalno potrdilo lahko izdajatelj zaradi različnih vzrokov prekliče (izguba zaupanja v tajnost privatnega ključa, izguba zaupanja v izdajatelja digitalnega potrdila, ...).

Preklicano digitalno potrdilo ni več veljavno (zaupanja vredno), prav tako izgubijo veljavnost vsi elektronski podpisi in časovni žigi, ki so bili z njim izdelani.
V arhivskem informacijskem paketu sta podprta CRL in OCSP tipa informacij o preklicih, ki sta šifrirana v Base64 obliki.

Sekcija je sestavljena iz etiket »RevocationData«, ki vsebujejo elemente opisane v spodnji tabeli.

Ime XML elementa	Tip XML elementa	Obveznost
Id	Atribut	DA
Type	Atribut	DA

Tabela 23: XML elementi etikete »RevocationData«

Atribut »Id«

Vrednost atributa »Id« je unikatni identifikator.

Atribut »Type«

Vrednost atributa »Type« predstavlja tip informacije o preklicih digitalnih potrdil.
Spodnja tabela prikazuje podprte vrednosti.

Vrednost	Opis
CRLDER	Informacija je seznam preklicanih digitalnih potrdil v binarni obliki.
OSCP	Informacija je rezultat protokola za sprotno preverjanje statusa elektronskega potrdila v binarni obliki

Tabela 24: Podprti tipi informacij o preklicu digitalnih potrdil

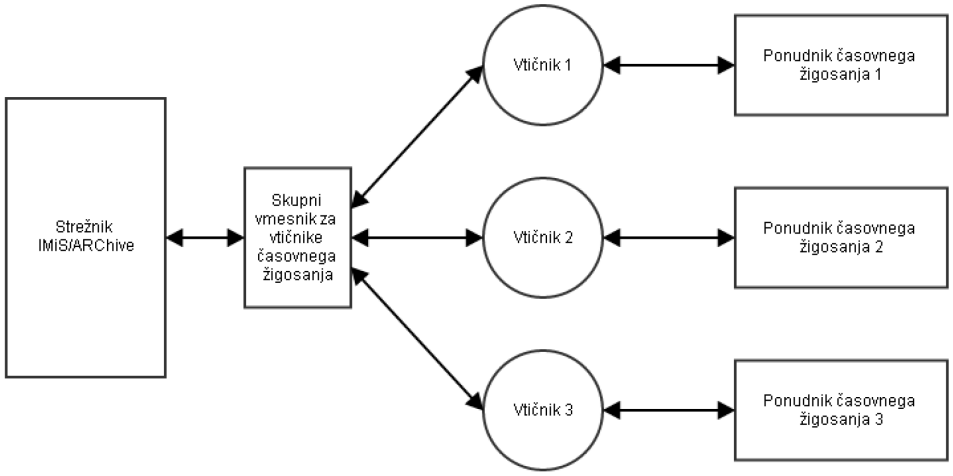
Primer: XML zapis za oba podprta tipa informacij.

```
<aip:RevocationData id="1" type="CRLDER">MIISKTCCERECAQEwDQ...
</aip:RevocationData>
<aip:RevocationData id="2" type="OCSP">MIIDBjCB7wI...
</aip:RevocationData>
```

3.6.6 Časovno žigosanje

Časovni žigi so dokazila, ki dokazujejo obstoj vsebine v času navedenim v časovnem žigu.
Storitev časovnega žigosanja opravljajo zaupanja vredni ponudniki
(npr. SI-TSA: http://www.si-tsa.si/). Ti morajo izpolnjevati stroge varnostne ukrepe,
sicer bi lahko podvomili v verodostojnost časovnih žigov, ki so jih ustvarili.

Strežnik IMiS®/ARChive Server uporablja koncept vtičnikov (angl. Plug-in) za pridobivanje časovnih žigov. Za vsakega ponudnika varnih časovnih žigov, ki svoje žige ponujajo preko tehnološko heterogenih načinov/vmesnikov, ponudimo svoj vtičnik.
Ta zna komunicirati s ponudnikom časovnih žigov in jih na strežniku poznan način vključevati v postopke zagotavljanja avtentičnosti hranjenega gradiva.



Slika 24: Koncept vtičnikov

3.6.6.1 Pridobivanje časovnega žiga

Postopek pridobivanja časovnega žiga je naslednji:

1. Strežnik izračuna prstni odtis podatkov, kateri so predmet časovnega žigosanja.
2. Prstni odtis se preko skupnega vmesnika pošlje vtičniku za časovno žigosanje.
3. Če ponudnik časovnega žiga podpira uporabo poljubne kode (angl. Nonce ali Arbitrary number), vtičnik s pomočjo generatorja naključnih števil ustvari naključno vrednost, ki jo skupaj s prstnim odtisom pošlje ponudniku časovnega žigosanja.
   Poljubna koda preprečuje napade s ponavljanjem (angl. Replay attack - http://en.wikipedia.org/wiki/Replay_attack).
4. Ponudnik časovnega žigosanja k prejetim podatkom doda časovno komponento in vse skupaj podpiše s privatnim ključem.
5. Če je vtičnik uporabil poljubno kodo, jo preveri, če se ista vrednost nahaja tudi v časovnem žigu. Če ga ne najde, časovni žig zavrže saj se upošteva, da je neveljaven.

3.6.6.2 Preverjanje časovnega žiga

Preverjanje časovnega žiga izvaja strežnik. Postopek preverjanja je naslednji:

- preveri se veljavnost digitalnega potrdila, ki je ustvarilo časovni žig;
- preverijo se informacije o preklicih vseh potrdil v verigi vključno z digitalnim potrdilom, ki je ustvarilo časovno žig;
- preveri se parameter »mTimestamp« v podaljških digitalnega potrdila, ki je ustvarilo časovni žig (glej poglavje 3.6.3.7 Razširjena uporaba ključa);
- preveri se veljavnost elektronskega podpisa časovnega žiga.

Če je katerokoli preverjanje neuspešno, se časovni žig obravnava kot neveljaven in ga strežnik zavrže. V tem primeru je potrebno postopek pridobivanja časovnega žiga ponoviti.

3.6.6.3 Primer

Primer v nadaljevanju prikazuje časovni žig v XML formatu. Za preverjanje časovnega žiga v XML formatu strežnik uporablja odprtokodno knjižnico XMLSec (http://www.aleksey.com/xmlsec).

```
<dsig:Signature xmlns:dsig="http://www.w3.org/2000/09/xmldsig#" id="TimeStampToken">
<dsig:SignedInfo>
<dsig:CanonicalizationMethod algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
<dsig:SignatureMethod algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
<dsig:Reference URI="#TimeStampInfo-13ED106F54C2C33ED420000000000007B81">
<dsig:DigestMethod algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
<dsig:DigestValue>LaeChaxwlaM8e9WZlRD0QjxzFrw=
</dsig:DigestValue>
</dsig:Reference>
<dsig:Reference URI="#TimeStampAuthority">
<dsig:DigestMethod algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
<dsig:DigestValue>j8bwhFukHoD6jcjmzgEZtXDF/ko=
</dsig:DigestValue>
</dsig:Reference>
</dsig:SignedInfo>
<dsig:SignatureValue>
sxbZkdzdWPcXHP06k1WhZzglyTtYLXUqaOzPaT/7VkwJ3haur5yiL
qDCO1zRogaPojVC06Ee545/VrdP1JmnzcwFXAW3UU+q6VrDDHLlyD
z4uW9hXxEy31YNkQmJ7BOicHNY9m2TOlk8tjCSec6s5UJxhJP49tY
u8wE7gMSgWpLlnMeAZCE/DVOPlqesVTUYzaLSbEqELpL5qFkvCmNC
TBjnaNuyKe/YhQWbvZOclvHePqyADNwX+IPOsAOS8NezpZHYriBO8
B+cAwglep/gZb1h8zDlqejHS8ibnFmvblk3ZOlbG/Y1SK36yk+Fu5
ya12KHlFOACzx/im3GE8vlWQ==
</dsig:SignatureValue>
<dsig:KeyInfo id="TimeStampAuthority">
<dsig:X509Data>
<dsig:X509Certificate>
```

MIIFYDCCBEigAwIBAgIEQLMGwDANBgkqhkiG9w0BAQUFADA+MQswC
QYDVQQGEwJzaTEbMBkGA1UEChMSc3RhdGUtaW5zdGl0dXRpb25zMR
IwEAYDVQQLEwlzaXRlc3QtY2EwHhcNMTIwNTIxMDc0ODM4WhcNMTc
wNTIxMjExMDU2WjBvMQswCQYDVQQGEwJzaTEbMBkGA1UEChMSc3Rh
dGUtaW5zdGl0dXRpb25zMRIwEAYDVQQLEwlTSVRFU1QtQ0ExLzAUB
gNVBAMTDXRzYS10ZXN0LTIwMTIwFwYDVQQFExAxMTExMTExMTExMT
ExMTExMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAvsX
FScylvL5dbkj/df82Ego08QH41xuz5TcKoBHCME+9fKcNiGVivJll
sZOihB4jta1JABKCEA4yJDEphNV9IK5MIYaqYW/3trijvCSZefGWr
W0kHt3gclDLpR2SbGDEACxpledlA2MZR1bnSDekgCOGbPiiz/jk4Q
PDGaGZ00/Fl5WfazvKqwnvHySsoSiUk7uUH8XgAv3Uv/ghYPjSfas
xXSvy/SKJAusKOyVFXiiT+SmaX+p60KroDkxhuKM9WzoehHcCPLzy
2JUPerr02+AX4zBytPsb+AVS9Xjl3KQIFpovSlph510H9hTused1g
b7gizt/leY3T3Nf4BgIxwIDAQABo4ICMzCCAi8wDgYDVR0PAQH/BA
QDAgeAMBYGA1UdJQEB/wQMMAoGCCsGAQUFBwMIMEAGA1UdIAQ5MDc
wNQYKKwYBBAGvWQEFAzAnMCUGCCsGAQUFBwIBFhlodHRwOi8vd3d3
LmNhLmdvdi5zaS9jcHMvMBoGA1UdEQQTMBGBD3RzYS10ZXN0QGdvd
i5zaTAbBgNVHQkEFDASMBAGCSqGSIb2fQdEHTEDAgEKMIH2BgNVHR
8Ege4wgeswVaBToFGkTzBNMQswCQYDVQQGEwJzaTEbMBkGA1UEChM
Sc3RhdGUtaW5zdGl0dXRpb25zMRIwEAYDVQQLEwlzaXRlc3QtY2Ex
DTALBgNVBAMTBENSTDMwgZGggY6ggYuGWGxkYXA6Ly94NTAwLmdvd
i5zaS9vdT1zaXRlc3QtY2Esbz1zdGF0ZS1pbnN0aXR1dGlvbnMsYz
1zaT9jZXJ0aWZpY2F0ZVJldm9jYXRpb25MaXN0P2Jhc2WGL2h0dHA
6Ly93d3cuc2lnZW4tY2Euc2kvY3JsL3NpdGVzdC9zaXRlc3QtY2Eu
Y3JsMCsGA1UdEAQkMCKADzIwMTIwNTIxMDc0ODM4WoEPMjAxNTA1M
jEyMjE0NDlaMB8GA1UdIwQYMBaAFFRJB0aHzx2JncqucqeooKBpty
HnMB0GA1UdDgQWBBQS8wflsAtvqW4RKcSTY5n4Mbg/dTAJBgNVHRM
EAjAAMBkGCSqGSIb2fQdBAAQMMAobBFY3LjEDAgSwMA0GCSqGSIb3
DQEBBQUAA4IBAQBh2Vgmo+MrbwqVcSvMln7aA9HLjw+HlM5KbXXvM
/XEmFWhVrWFhbqX1f+OoizSfDv4sErCPb3zzeSG2mnEREzk9gqzbt
NlqfvzdZ7Xd7bGUEzwHEcVE8DSW2bJGeSkhyX2AWvt4eHyqwgXply
MzFOVSWteivgwgvMtUZjSbeClpELUblGWzwyxW15SHufOdJdtgcnn
r2hCLtkYJ9ky4T0m5gvvyOxmQi+o3rLvPa5yLeHYa//KzPo+H8CPf
UGqRFTPVSykalm6evTg6CDQvG9jfii1PbaeKQRm0xLQPpVNYFA66e
2DwVv+9UdKzcoBfujvZRtEuPlnphHBlzxEkFU3
</dsig:X509Certificate>
</dsig:X509Data>
</dsig:KeyInfo>
<dsig:Object id="TimeStampInfo-13ED106F54C2C33ED420000000000007B81">
<ts:TimeStampInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
xmlns:ts="http://www.entrust.com/schemas/timestamp-protocol-20020207">
<ts:Policy id="http://www.si-tsa.si/dokumenti/SI-TSA-politika-za-casovni-zig-1.pdf"/>
<ts:Digest>
<ds:DigestMethod algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
<ds:DigestValue>93WQD+wSgMA5KCzcmjYe55NBKEc=
</ds:DigestValue>
</ts:Digest>
<ts:SerialNumber>1084876374609840033695664166241473103450 89
</ts:SerialNumber>
<ts:CreationTime>2014-03-31T15:00:00.089Z
</ts:CreationTime>

<ts:Nonce>10600496071266535864
</ts:Nonce>
</ts:TimeStampInfo>
</dsig:Object>
</dsig:Signature>

Etiketa »dsig:SignedInfo«

Etiketa vsebuje informacije, kaj je v XML predmet elektronskega podpisa in metode za normalizacijo XML pri preverjanju časovnega žiga:

- dsig:CanonicalizationMethod: metoda za normalizacijo XML.
- dsig:SignatureMethod: uporabljen algoritem pri ustvarjanju elektronskega podpisa.
- dsig:Reference URI="#TimeStampInfo-13ED106F54C2C33ED420000000000007B81": referenca na etiketo dsig:Object, ki je predmet vsebine elektronskega podpisa. Vsebuje tudi algoritem ter prstni odtis, ki pripadata tej etiketi.
- dsig:Reference URI="#TimeStampAuthority": referenca na etiketo»dsig:KeyInfo«, ki je prav tako predmet vsebine elektronskega podpisa. Vsebuje algoritem ter prstni odtis, ki pripadata etiketi.

Etiketa »dsig:SignatureValue«

Etiketa vsebuje vrednost elektronskega podpisa.

Etiketa »dsig:KeyInfo«

Etiketa vsebuje informacije o digitalnem potrdilu, ki je ustvarilo časovni žig:

- dsig:X509Data – etiketa vsebuje etiketo dsig:X509Certificate, ki vsebuje digitalno potrdilo, ki je ustvarilo elektronski podpis.

Etiketa »dsig:Object«

Etiketa vsebuje naslednje podatke:

- ts:TimeStampInfo: informacije o protokolu;
- ts:Policy: informacije o politiki ponudnika časovnega žigosanja;
- ts:Digest: prstni odtis (ter URI algoritma, ki ga je izdelal), ki ga je ponudnik prejel v časovno žigosanje;
- ts:SerialNumber: serijska številka;
- ts:CreationTime: datum in čas nastanka časovnega žiga;
- ts:Nonce: poljubna koda, ki je bila uporabljena v postopku ustvarjanja zahtevka za časovno žigosanje.

3.6.7 Pravila

Pravila določajo pogoje, kdaj naj se izvajata postopka ustvarjanja in podaljševanja dokazil, ter katere vsebine so predmet teh postopkov. Ustvarjanje in podaljševanje dokazil poteka paketno. Uporablja se Merklovo drevo, razen v primeru, če se v paketu nahaja samo en AIP ali arhivski časovni žig. Paketi so določeni kot minimalno in maksimalno število arhivskih informacijskih paketov, ki se časovno žigosajo z enim časovnim žigom.
Primer konfiguracije paketa prikazuje naslednji XML zapis:

```
<Settings>
    <MinBatchSize>1</MinBatchSize>
    <MaxBatchSize>300</MaxBatchSize>
</Settings>
```

Etiketa »MinBatchSize«

Vsebuje minimalno število arhivskih informacijskih paketov, ki se časovno žigosajo z enim časovnim žigom. Vrednost ne sme biti manjša od 1 in večja od vrednosti »MaxBatchSize«.

Etiketa »MaxBatchSize«

Vsebuje maksimalno število arhivskih informacijskih paketov, ki se časovno žigosajo z enim časovnim žigom. Vrednost ne sme biti manjša od »MinBatchSize« in večja od 1.000.000.

3.6.7.1 Pravila za ustvarjanje dokazil

Pravila za ustvarjanje dokazil se uporabljajo pri preverjanju, ali entiteta ustreza pogojem za zagotavljanje avtentičnosti. Če lastnosti entitete ustrezajo vsaj enemu pravilu, potem se bo za tako entiteto ustvarilo dokazilo, ki ga pravilo določa. V primeru, da lastnostim entitete ustreza več pravil, potem se upošteva prvo tako pravilo. Prednost imajo pravila, ki veljajo tudi za vsebovane entitete.

Primer pravila prikazuje naslednji XML zapis:

```
<Rule>
    <Id>307a8c60-390c-470d-9692-a43311bd51ef</Id>
    <Enabled>true</Enabled>
    <Type>Explicit</Type>
    <IncludeChildren>true</IncludeChildren>
    <Scope type="ClassificationCode">C=57</Scope>
    <Expression>[sys:Status]="2"</Expression>
    <TemplateFilter></ TemplateFilter>
    <TimeStampProviderId>a48d9c39-456a-470c-9c8b-e54bb91fc1dc</TimeStampProviderId>
</Rule>
```

Etiketa »Id«

Etiketa vsebuje unikatni identifikator pravila.

Etiketa »Enabled«

Vrednost etikete je lahko »True« ali »False« in določa, ali je pravilo omogočeno ali ne.

Etiketa »Type«

Vrednost etikete je lahko »Explicit« ali »Implicit«. Eksplicitna pravila se preverjajo pri operacijah nad entitetami (shranjevanje, sprememba statusa...), v procesu časovnega žigosanja pa se preverjajo vsa pravila.

Etiketa »IncludeChildren«

Vrednost etikete je lahko »True« ali »False«, določa pa veljavnost pravila na pod entitetah.

Etiketa »Scope«

Vrednost etikete vsebuje identifikator entitete, pod katero se pravilo upošteva pri preverjanju lastnosti entitete. Če entiteta ni vsebovana, potem tako pravilo za njo ne velja.

Etiketa »Expression«

Vrednost vsebuje iskalni niz, ki se preverja z lastnostmi entitete. Če iskalni niz ne ustreza vrednostim v entiteti, potem pravilo za entiteto ne velja.

Etiketa »TemplateFilter«

Vrednost etikete vsebuje imena predlog, s katerimi morajo biti entitete narejene, da lahko ustrezajo pravilu. Vrednost se uporablja kot dodatni filter za vrednost »Expression«.
Če je vrednost »TemplateFilter« prazen niz, potem se vrednost ne upošteva pri preverjanju pravila z vrednostjo entitete.

Etiketa »TimeStampProviderId«

Vrednost etikete predstavlja unikatni identifikator ponudnika časovnih žigov, s katerim se bodo entitete, ki ustrezajo pravilu, časovno žigosale.
Primer: Pravilo iz našega primera tako velja za entitete, ki so pod razredom s klasifikacijsko oznako »57« (velja tudi za sam razred) in imajo vrednost »sys:Status« enako 2 (zaprta entiteta) ali pa so pod entitete razreda ali zadeve, ki je zaprta (recimo dokumenti v zaprti zadevi).

3.6.7.2 Pravila za podaljševanje dokazil

Pravila za podaljševanje dokazil se uporabljajo pri enostavnem in kompleksnem podaljševanju. Primer prikazuje naslednji XML zapis:

```
<Rule>
    <Id>6062074f-6d05-4f24-8592-866623ade8bf</Id>
    <Enabled>true</Enabled>
    <Digest>SHA1</Digest>
    <CertificateExpirationTreshold>90d</CertificateExpirationTreshold>
    <TimeStampProviderId>bac87d4d-fd1b-4065-bb36-5d03eeb25b6e</TimeStampProviderId>
</Rule>
```

Etiketa »Id«

Etiketa vsebuje unikatni identifikator pravila.

Etiketa »Enabled«

Vrednost etikete je lahko »True« ali »False« in določa, ali je pravilo omogočeno ali ne.

Etiketa »Digest«

Vrednost etikete predstavlja tip zgoščevalne funkcije, ki se bo uporabila pri časovnem žigosanju.

Etiketa »CertificateExpirationTreshold«

Vrednost predstavlja časovni okvir, v katerem bodo zajeti vsi časovni žigi z istim tipom zgoščevalne funkcije in katerih digitalna potrdila pretečejo v danem časovnem okvirju.

Etiketa »TimeStampProviderId«

Vrednost predstavlja unikatni identifikator ponudnika časovnih žigov, s katerim se bodo dokazila pred iztekom podaljšala.

Pravilo iz našega primera tako velja za vsa dokazila, ki so bila narejena z zgoščevalno funkcijo SHA1, in katerih digitalna potrdila potečejo v roku 90 dni.

3.7 Odbiranje in izločanje

Odbiranje in izločanje je nadzorovan in načrtovan postopek za izvedbo prenosa, uničenja ali trajne hrambe elektronskega gradiva. Vsak razred, zadeva ali dokument (uvrščen neposredno pod razredom) mora imeti določen vsaj en rok hrambe (glej poglavje 3.3.9.2 Upravljanje s povezavami politik hrambe). Ta določa koliko časa je potrebno posamezno entiteto hraniti v arhivskem sistemu. Poleg časovnega okvira rok hrambe vsebuje še privzeto akcijo, ki se bo izvedla v postopku odbiranja in izločanja. Akcijo lahko spremenijo uporabniki s pravicami (člani komisije), ki izvajajo odbiranje in izločanje.

Postopek delimo v naslednje faze:

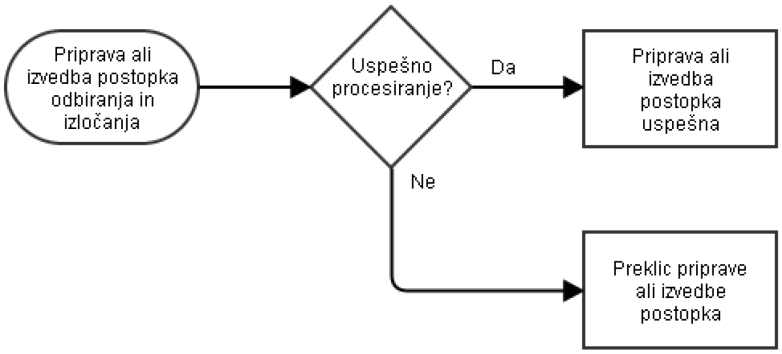
- priprava
- odločanja
- izvedba.

V postopku priprave in izvedbe odbiranja in izločanja lahko pride do napake zaradi različnih vzrokov. V primeru napake se postopek samodejno prekliče.

Izvedejo se naslednje spremembe:

- Status se spremeni v »Zaprto« (angl. Closed).
- Stanje se spremeni v »Neuspešno« (angl. Failed). Vrednost stanja se nahaja v atributu »sys:ret:rev:State«.
- V atribut »sys:ret:rev:Message« se zapiše vzrok preklica.

Postopka pri katerem je prišlo do napake se ne da ponovno pripraviti ali izvesti. Prav tako ni dovoljeno njegovo urejanje. Pri postopku priprave ali izvedbe je potrebno poudariti, da se vse akcije nad entitetami izvajajo v imenu uporabnika, ki je pripravo ali izvedbo sprožil.
Tako na postopek priprave in izvedbe vplivajo uporabniške pravice in stopnja tajnosti na entitetah. V nadaljevanju je prikazan diagram poteka postopka odbiranja in izločanja v primeru priprave ali izvedbe.



Slika 25: Diagram poteka izvajanja postopka odbiranja in izločanja

3.7.1 Postopek priprave za odbiranje in izločanje

Prva faza postopka odbiranja in izločanja je priprava. Uporabnik s pravico ustvari entiteto, ki bo kasneje prešla v postopek odbiranja in izločanja.
Poleg vseh obveznih atributov mora vnesti še vrednost enega izmed naslednjih atributov:

- »sys:ret:rev:Query«: vrednost predstavlja iskalni niz za entitete (glej poglavje 3.5.2 Pravila iskalnega niza), ki bodo vključene v postopek odbiranja in izločanja.
  Uporaba iskalnega niza pri postopku priprave za odbiranje in izločanje je namenjena ad-hoc upravljanju z entitetami mimo politik hrambe.
- »sys:ret:rev:Schedules«: vrednosti predstavljajo unikatne identifikatorje politik hrambe, ki se bodo upoštevale v postopku priprave za odbiranje in izločanje.

V primeru, da imata oba atributa določene vrednosti, se postopek priprave odbiranja in izločanja prekliče, saj gre za neveljavno stanje.
Pomembno vlogo v postopku odbiranja in izločanja ima atribut »sys:ret:rev:Scope«.

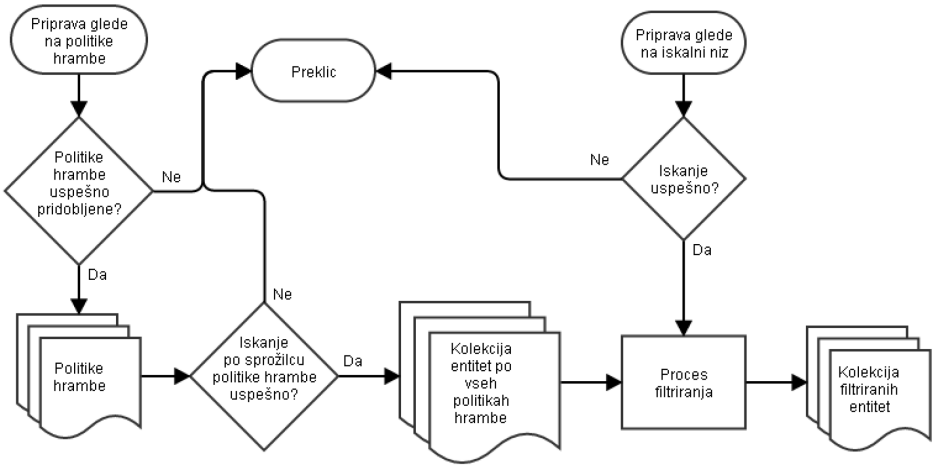
Vrednost atributa je klasifikacijska oznaka entitete, pod katero se bo izvedlo iskanje vseh entitet, ki ustrezajo pogojem. Če vrednost atributa ni nastavljena se bo iskanje izvedlo po celotnem arhivu.
Pripravo postopka odbiranja in izločanja delimo na:

- priprava glede na iskalni niz
- priprava glede na roke hrambe.

Pri postopku priprave glede na iskalni niz se izvede iskanje entitet, ki ustrezajo pogojem v iskalnem nizu. Rezultat je zbirka entitet, nad katero se izvede filtriranje. Filtriranje iz zbirke odstrani vse entitete, ki ne sodijo v postopek odbiranja in izločanja (glej poglavje 3.7.4 Postopek filtriranja). Iz filtrirane zbirke se nato izdelajo XML dokumenti z informacijami o entitetah in dovoljenih akcijah nad njimi, s katerimi lahko uporabniki s pravicami (člani komisije) uspravljajo v postopku odločanja.
Pri postopku priprave glede na roke hrambe velja podobno. Iz vsake politike hrambe, ki je del postopka odbiranja in izločanja, se iz sprožilca vzame iskalni niz (vrednost atributa »sys:ret:pol:Trigger«) in izvede iskanje entitet. Ko je iskanje po vseh rokih hrambe končano, se zbirke vseh iskanj združijo v unijo, ki predstavlja zbirko entitet po vseh rokih hrambe.
Ta zbirka gre skozi postopek filtriranja kjer se dodatno preveri, ali ima entiteta vsaj en efektiven rok hrambe, ki je del postopka odbiranja in izločanja.

Če ta pogoj ni izpolnjen, entiteta ne pripada trenutnemu postopku odbiranja in izločanja.
Na koncu se iz filtrirane zbirke izdelajo XML dokumenti z informacijami o entitetah in dovoljenimi akcijami, katere lahko uporabnik spreminja v postopku odločanja.

Če pri postopku priprave pride do kakršne koli napake (npr: neveljaven iskalni niz, izbrisana politika hrambe, ipd.) se postopek samodejno prekliče. Za ponovno pripravo je potrebno celoten postopek priprave ponoviti. V nadaljevanju je prikazan postopek priprave odbiranja in izločanja.



Slika 26: Priprava postopka odbiranja in izločanja

Po uspešno izvedenem postopku priprave se izdelajo XML dokumenti, ki predstavljajo vsebino postopka.
Ločimo dva tipa XML dokumentov:

- dokumenti samo za branje (nahajajo se v atributu »sys:ret:rev:Lists«);
- dokumenti, ki jih člani komisije v postopku odločanja spreminjajo (nahajajo se v atributu »sys:Content«).

Dokument samo za branje vsebuje poleg informacij o entiteti tudi nabor akcij, ki so dovoljene za izvajanje nad entiteto. Nabor akcij je odvisen od številnih dejavnikov, ki so podrobneje opisani v poglavju (glej poglavje 3.7.4 Postopek filtriranja). Dokumenti, katere člani komisije lahko spreminjano v postopku odločanja vsebujejo poleg informacij o entiteti tudi privzeto akcijo in razlog.

3.7.2 Postopek odločanja pri odbiranju in izločanju

V postopku odločanja člani komisije sprejemajo odločitve, kaj se bo zgodilo z entitetami, ki so bile predmet postopka priprave. Člani komisije izberejo akcijo, ki je dovoljena za posamezno entiteto, dodajo komentar, spremenijo razlog za izbrano akcijo, ipd.

V primeru prenosa entitet v tretji arhivski sistem, lahko dodajo tudi referenco na prenesene entitete ali potrdijo uspešen prenos entitete. V kolikor uspešnega prenosa ne potrdijo, je uničenje entitet na strežniku zavrnjeno.

Po končanem postopku odločanja se postopek izvede ali prekliče.
V primeru izvedbe postopka, se vrednost atributa »sys:ret:rev:Action« postavi na »Zaključeno« (angl. Complete), kar je znak strežniku, da je postopek izveden.
V primeru preklica, se vrednost atributa postavi na »Zavržen« (angl. Discard), kar je znak strežniku za preklic postopka odbiranja in izločanja.

3.7.3 Izvedba postopka odbiranja in izločanja

Izvedbo postopka odbiranja in izločanja delimo na:

- preklic postopka odbiranja in izločanja
- izvedba postopka odbiranja in izločanja.

Pri preklicu postopka odbiranja in izločanja:

- strežnik vrednost atributa »sys:ret:rev:State« nastavi na »Zavržen« (angl. Discarded);
- status se spremeni v »Zaprto« (angl. Closed);
- v atribut »sys:ret:rev:Message« se zapiše sporočilo, da je bil postopek odbiranja in izločanja preklican na zahtevo uporabnika.

Pri postopku izvedbe odbiranja in izločanja se najprej naložijo vse informacije o entitetah iz XML dokumentov, ki so samo za branje. Nato se izvede preverjanje po naslednjih postavkah:

- oblika XML dokumentov mora ustrezati predpisani XSD shemi;
- vse entitete iz XML dokumentov, ki so samo za branje morajo biti prisotne v spremenljivih XML dokumentih in obratno;
- izbrane akcije na entitetah morajo ustrezati dovoljenim akcijam za te entitete.

Če je preverjanje katere koli postavke neuspešno, se postopek odbiranja in izločanja izvede z napako. Celoten postopek je potrebno ponoviti.

Izjema je preverjanje izbrane akcije na entitetah. V tem primeru se izvajanje prekine z napako. Stanje postopka odbiranja in izločanja se postavi na »InReview«. To omogoča uporabniku, da popravi izbrane akcije na entitetah in ponovi postopek. Po uspešnem preverjanju se nad zbirko entitet ponovno izvede postopek filtriranja. Postopek filtriranja je podrobneje opisan v poglavju 3.7.4 Postopek filtriranja.

V nadaljevanju sledi še nekaj praktičnih primerov, zakaj lahko pride do prekinitve izvajanja akcij v postopku izvedbe odbiranja in izločanja.

Primer 1: V postopku odbiranja in izločanja imamo poleg drugih entitet tudi zadevo z dvema vsebovanima zadevama:
F=2015-00011
F=2015-00011^F=00001
F=2015-00011^F=00002

V postopku odločanja je izbrana akcija »Dispose« za vse tri entitete. Člani komisije ugotovijo, da je na zadevo »F=2015-00011« vezano zadržanje postopka. Izvedba akcije je na vseh treh entitetah prekinjena, saj zadržanje postopka rekurzivno vpliva na vse vsebovane entitete. Akcija se izvede na vseh ostalih entitetah, ki so predmet postopka odbiranja in izločanja.

Primer 2: Vzemimo kombinacijo iz prvega primera, le da sedaj na zadevo »F=2015-00011« ni vezano zadržanje postopka odbiranja in izločanja. Zadeva vsebuje novo zadevo »F=2015-00011^F=00003«, ki v postopku priprave postopka še ni obstajala:
F=2015-00011
F=2015-00011^F=00001
F=2015-00011^F=00002
F=2015-00011^F=00003

V tem primeru se akcija »Dispose« izvede na vsebovanih zadevah »F=2015-00011^F=00001« in »F=2015-00011^F=00002«, uničenje zadeve »F=2015-00011« pa ni mogoče, saj le ta vsebuje novo vsebovano zadevo »F=2015-00011^F=00003«, ki pa ni predmet postopka odbiranja in izločanja.
Če pride do napake pri izvajanju akcije na določeni entiteti, se izvedba akcije prekine na vseh nadrejenih entitetah, ki so del postopka odbiranja in izločanja.

Primer 3: Vzemimo kombinacijo iz prvega primera, pri čemer se v vsebovani zadevi »F=2015-00011^F=00002« nahajata dva dokumenta:
F=2015-00011^F=00002
F=2015-00011^F=00002^D=00001
F=2015-00011^F=00002^D=00002

Uporabnik, v imenu katerega se izvaja postopek odbiranja in izločanja nima pravice brisanja dokumenta »F=2015-00011^F=00002^D=00002«. Posledično se tudi vsebovana zadeva »F=2015-00011^F=00002« in njena nadrejena zadeva »F=2015-00011« ne uničita. Uniči pa se vsebovana zadeva »F=2015-00011^F=00001«.

Primer 4: Vzemimo kombinacijo iz prvega primera, kjer je akcija za vsebovanih zadevah »F=2015-00011^F=00001« in »F=2015-00011^F=00002« nastavljena na »Permanent«. Akcija na zadevi »F=2015-00011« je nastavljena na »Dispose«.
Ob uspešni izvedbi akcije »Premanent« je izvedba akcije »Dispose« neuspešna, saj zadeva »F=2015-00011« vsebuje dve vsebovani zadevi, ki se hranita trajno. Posledično se zadeva ni uniči.

Izvedba vseh akcij se zapiše v revizijsko sled (če je le ta omogočena), poleg tega pa se zapišejo še v ločeno poročilo. Poročilo je sestavljeno iz XML in tekstovnega dokumenta.
V XML dokumentu so zabeležene akcije na entitetah, ki so del postopka odbiranja in izločanja.
V tekstovnem dokumentu so zabeležene tudi akcije na entitetah, ki niso del postopka odbiranja in izločanja (dokumenti v zadevah). Takšne entitete so posredno odvisne, saj nadrejene entitete določajo akcije, ki se bodo na njih izvedle.

3.7.4 Postopek filtriranja

V postopku filtriranja se za celotno zbirko entitet preveri, ali ustrezajo vsem pogojem postopka odbiranja in izločanja. Iz postopka priprave se izločijo vse entitete (ter posledično tudi vse njihove nadrejene entitete), ki ustrezajo naslednjim pogojem:

- Uporabnik, v imenu katerega se izvaja postopek priprave nima pravice videti obstoja entitete.
- Na entiteti je prisotno vsaj eno zadržanje postopka odbiranja in izločanja.
- Entiteta ne vsebuje vsaj enega efektivnega roka hrambe, ki bi ustrezal naboru rokov hrambe, s katerimi se izvaja postopek priprave za odbiranje in izločanje.
  V primeru postopka priprave odbiranja in izločanja po iskalnem nizu tega preverjanja ni.
- Entiteta vsebuje podrejene entitete, ki so predmet odbiranja in izločanja, vendar niso prisotne v zbirki entitet za trenutno pripravo odbiranja in izločanja.

Za vse ostale entitete se preveri prisotnost atributa »sys:Significance«.
Naslednje vrednosti vplivajo na dovoljene akcije, ki so na voljo na dotični entiteti:

- Vrednosti »Vital« ali »Permanent« na entiteti: vplivajo na to, da se privzeta akcija postavi na »Permanent«. Uporabnik ne more izbrati druge akcije kot »Review«, saj atribut določa pomembnost entitete. Vrednost vpliva tudi na vse nadrejene entitete, ki se jim posledično spremeni privzeta akcija.

- Vrednost »Retain«: predstavlja opozorilo, da se mora entiteta ohraniti, zato se privzeta akcija nastavi na »Review«. Uporabnik ima možnost spremembe te akcije. Vrednost nima vpliva na nadrejene entitete.

Na privzeto akcijo entitete vplivajo tudi naslednji pogoji:

- Če ima entiteta več efektivnih rokov hrambe pomeni, da je entiteta v konfliktu. Privzeta akcija se postavi na »Review«. Uporabnik ima možnost izbire vseh akcij, ki so na voljo.
- Če entiteta vsebuje druge entitete, ki so že bile v drugih postopkih odbiranja in izločanja ter so postale trajno gradivo, se tudi dotični entiteti privzeta akcija postavi na »Permanent«. To velja tudi za vse nadrejene entitete. Uporabnik nima možnosti izbire druge akcije razen »Review«.

Če stanje entitete ne ustreza nobenemu zgoraj opisanemu pogoju, se privzeta akcija nastavi na vrednost, ki jo vsebuje politika hrambe, ki je za to entiteto veljavna. Uporabnik ima možnost izbire vseh akcij, ki so na voljo.

Postopek filtriranja se uporablja tudi pri postopku izvedbe odbiranja in izločanja.
V tem primeru se označijo entitete (ter posledično vse nadrejene entitete), nad katerimi se izbrane akcije zaradi različnih vzrokov ne smejo izvesti:

- Na entiteto je vezano vsaj eno zadržanje postopka odbiranja in izločanja.
- Med izvajanjem akcije na vsebovanih entitetah je prišlo do napake (npr.: uporabnik, v imenu katerega se izvaja postopek odbiranja in izločanja nima zadostnih pravic za izvedbo operacije, ipd).
- Stanje na strežniku ob izvedbi odbiranja in izločanja je drugačno od stanja ob pripravi postopka.

  Primer: nove entitete, ki so predmet odbiranja in izločanja, niso pa prisotne v trenutnem postopku, ipd.

- Izbrane akcije so v konfliktu.

  Primer: vsebovana zadeva ima izbrano akcijo »Review«, nadrejena zadeva pa akcijo »Dispose«. V tem primeru se izbrana akcija na zadevi ne more izvesti, ker vsebuje zadevo, ki se bo odbirala v drugem postopku odbiranja in izločanja.

3.7.5 XML format dokumentov

V postopku odbiranja in izločanja nastopajo naslednje vrste XML dokumentov:

- dokumenti samo za branje
- dokument za urejanje
- poročila.

3.7.5.1 Sintaksa dokumentov samo za branje

Dokumenti samo za branje vsebujejo stanje na strežniku v postopku priprave za odbiranje in izločanje. Vsebujejo tudi informacije o entiteti (klasifikacijsko oznako, interni Id, naslov), veljavne akcije, razlog in efektivne roke hrambe, ki veljajo za to entiteto.

3.7.5.1.1 Etiketa »Review«

Etiketa predstavlja korenski element in vsebuje elemente iz spodnje tabele.

Ime XML elementa	Tip XML elementa	Obveznost
Header	Etiketa	Da
Entity	Etiketa	Ne

Tabela 25: XML elementi etikete "Review"

3.7.5.1.2 Etiketa »Header«

Etiketa vsebuje verzijo XML dokumenta, datum začetka postopka priprave odbiranja in izločanja, klasifikacijsko oznako entitete (predstavlja obseg iskanja entitet za postopek odbiranja in izločanja), akcije, ki so na voljo, vnaprej določene razloge ter identifikatorje vseh rokov hrambe. Elementi etikete so opisani v spodnji tabeli.

Ime XML elementa	Tip XML elementa	Obveznost
Version	Atribut	Da
StartDate	Atribut	Da
Scope	Atribut	Ne
Action	Etiketa	Da
Reason	Etiketa	Da
Schedule	Etiketa	Da

Tabela 26 XML elementi etikete "Header"

Atribut »Version«

Atribut predstavlja verzijo XML dokumenta.

Atribut »StartDate«

Atribut predstavlja datum in čas začetka priprave postopka odbiranja in izločanja.

Atribut »Scope«

Atribut predstavlja klasifikacijsko oznako entitete, pod katero se je izvedlo iskanje entitet v postopku priprave. Če atribut ni prisoten pomeni, da je bilo iskanje izvedeno po celotnem arhivu.

Etikete »Action«

Etikete predstavljajo nabor akcij, ki so na voljo v postopku izvedbe odbiranja in izločanja.
Etiketa ima obvezen atribut »Id«.

Etikete »Reason«

Etikete predstavljajo privzete razloge za akcije, ki se bodo izvršile nad entitetami. Etiketa ima obvezen atribut »Id«, ki predstavlja unikaten identifikator katerega naslavljajo druge etikete.

Etikete »Schedule«

Etikete vsebujejo »Id« politik hrambe za trenuten postopek odbiranja in izločanja. Te so efektivne za posamezne entitete ter politike hrambe, nad katerimi se izvaja postopek priprave. Etiketa ima obvezen atribut »Id«, ki predstavlja unikaten identifikator katerega naslavljajo druge etikete.

3.7.5.1.3 Etiketa »Entity«

Etikete vsebujejo informacije o entiteti, ki je vključena v postopek odbiranja in izločanja.

Elementi etikete so opisani v spodnji tabeli.

Ime XML elementa	Tip XML elementa	Obveznost
Id	Atribut	Da
IdType	Atribut	Da
Type	Atribut	Da
Id85	Atribut	Da
Title	Atribut	Da
Action	Atribut	Da
Reason	Atribut	Ne
Schedule	Etiketa	Da

Tabela 27: XML elementi etikete "Entity"

Atribut »Id«

Atribut predstavlja identifikator entitete.

Atribut »IdType«

Atribut predstavlja tip identifikatorja entitete. Veljavne vrednosti so:

- »C«: identifikator je klasifikacijska oznaka
- »E«: identifikator je unikatni zunanji identifikator
- »I«: identifikator je interni identifikator entitete.

Atribut »Type«

Atribut predstavlja tip entitete. Veljavne vrednosti so:

- »C«: entiteta predstavlja razred
- »F«: entiteta predstavlja zadevo
- »D«: entiteta predstavlja dokument.

Atribut »Id85«

Atribut predstavlja interni »Id« entitete.

Atribut »Title«

Atribut predstavlja naslov entitete.

Atribut »Action«

Atribut je 32-bitna nepredznačena vrednost, ki vsebuje informacije o veljavnih akcijah na entiteti ter privzeto akcijo za entiteto. Prvi štirje biti vrednosti (od desne proti levi oz. z LSB strani) predstavljajo veljavne akcije in so opisani v spodnji tabeli. Če je vrednost bita »1«, je akcija dovoljena, v nasprotnem primeru akcija ni dovoljena.

Bit 3 – »Review«	Bit 2 – »Transfer«	Bit 1 – »Permanent«	Bit 0 – »Dispose«

Tabela 28: Pozicija bitov ki predstavljajo dovoljene akcije

Naslednjih 24-bitov se ne uporablja. Vrednost v končnih 4-bitov predstavlja vrednost atributa »Id« v »Action« etiketi znotraj etikete »Header« in obenem privzeto akcijo za entiteto.

Atribut »Reason«

Atribut predstavlja referenco na atribut »Id« v »Reason« etiketah v etiketi »Header«, vrednost »Reason« pa predstavlja privzet razlog za odbiranje in izločanje entitete.

Etiketa »Schedule«

Etiketa vsebuje obvezen atribut »Id«, ki vsebuje referenco na atribut »Id« v »Schedule« etiketah v etiketi »Header«: Vrednost predstavlja interni identifikator za rok hrambe, ki je efektiven za entiteto.

Primer:

```
<Review>
    <Header StartDate="2015-08-17T18:14:19.170+02:00" Version="1">
        <Action Id="1">Dispose</Action>
        <Action Id="2">Permanent</Action>
        <Action Id="3">Transfer</Action>
        <Action Id="4">Review</Action>
        <Reason Id="R0">Razlog 1</Reason>
        <Reason Id="R1">Razlog 2</Reason>
        <Schedule
        Id="S0">0d5021dd60c06fcf09fe42d6ac61a4434727e5a69d9253d5ed17ef84879e1996
        </Schedule>
        <Schedule
        Id="S1">098197cff3ea91e9feadfd2a629ec4ad0cf3d06e6025c616853a58363819c8f6
        </Schedule>
    </Header>
    <Entity Id="C=99" IdType="C" Type="C"
    Id85="jkh920856nvbghd84ikfnbj2185hwqbo58djhn378jck085jd" Title="Legacy object containers"
    Action= "536870927" Reason="R1">
        <Schedule Id="S1"/>
    </Entity>
    <Entity Id="C=04^F=2015-00012^F=00001" IdType="C" Type="F"
    Id85="b43ghjf983jxcgfsa6r9fhnnbnd84hfcsfk93e7tb184jshfg" Title="Review test"
    Action="536870927" Reason="R1">
        <Schedule Id="S1"/>
    </Entity>
</Review>
```

Vrednost atributa »Action« za entiteto »C=04^F=2015-00012^F=00001« je: 536870927.

Binarna reprezantacija je naslednja: 00100000000000000000000000001111.

Na tej entiteti so vse akcije omogočene (vrednost prvih 4 bitov je 1111). Zadnji 4 biti nam dajo vrednost 0010 (2), kar je vrednost »Id« atributa za akcijo »Permanent«.

3.7.5.2 Sintaksa dokumentov za urejanje

Sintaksa XML dokumentov, ki se urejajo v postopku odločanja je zelo podobna dokumentom samo za branje. Vsebuje nekaj dodatnih atributov, ki se uporabljajo v postopku izvajanja odbiranja in izločanja.

3.7.5.2.1 Etiketa »Review«

Etiketa predstavlja korenski element dokumenta in vsebuje elemente iz spodnje tabele.

Ime XML elementa	Tip XML elementa	Obveznost
Header	Etiketa	Da
Entity	Etiketa	Ne

Tabela 29: Elementi etikete "Review"

3.7.5.2.2 Etiketa »Header«

Etiketa vsebuje verzijo XML dokumenta, akcije, ki so na voljo in pripadajoči razlogi.

Elementi etikete so opisani v spodnji tabeli.

Ime XML elementa	Tip XML elementa	Obveznost
Version	Atribut	Da
Action	Etiketa	Da
Reason	Etiketa	Da

Tabela 30: Elementi etikete "Header"

Atribut »Version«

Atribut predstavlja verzijo XML dokumenta.

Etiketa »Action«

Etikete predstavljajo nabor akcij, ki so na voljo. Vsebujejo obvezen »Id« atribut, ki predstavlja referenco na akcijo, ki je izbrana za posamezno entiteto v postopku izvajanja odbiranja in izločanja.

Etiketa »Reason«

Etikete predstavljajo nabor vzrokov za izvedbo akcij nad entitetami v postopku izvajanja odbiranja in izločanja. Vsebujejo obvezen atribut »Id«, ki predstavlja referenco na vzrok, na katerega se posamezna entiteta sklicuje.

3.7.5.2.3 Etiketa »Entity«

Poleg atributov, ki se nanašajo na entiteto, etiketa vsebuje še izbrano akcijo in vzrok.

V primeru akcije »Transfer« je potrebno z atributom potrditi še uspešen prenos entitete.

Opcijsko se lahko vnese referenca na preneseno entiteto ter komentar.

Elementi etikete so opisani v spodnji tabeli.

Ime XML elementa	Tip XML elementa	Obveznost
Id	Atribut	Da
IdType	Atribut	Da
Action	Atribut	Da
Reason	Atribut	Da
TrfSucceed	Atribut	Ne
TrfRefId	Atribut	Ne
Comment	Atribut	Ne

Tabela 31: Elementi etikete "Entity"

Atributa »Id« in »IdType« predstavljata identifikator entitete in tip identifikatorja ter sta enaka kot v dokumentu samo za branje (glej poglavje 3.7.5.1 Sintaksa dokumentov samo za branje).

Atribut »Action«

Vrednost atributa je referenca na izbrano akcijo v etiketi »Header«.

Atribut »Reason«

Vrednost atributa je referenca na izbran razlog v etiketi »Header«.

Atribut »TrfSuceed«

V kolikor je atribut prisoten, ta vrednost atributa pove, ali je bil prenos entitete uspešen ali ne (vrednosti »True« ali »False«).

Atribut »TrfRefId«

Atribut vsebuje vrednost, ki predstavlja referenco na preneseno entiteto.

Atribut »Comment«

Atribut vsebuje komentar uporabnika, ki izvaja postopek odločanja.

Primer:

```
<Review>
    <Header Version="1">
        <Action Id="A0">Dispose</Action>
        <Action Id="A1">Permanent</Action>
        <Action Id="A2">Transfer</Action>
        <Action Id="A3">Review</Action>
        <Reason Id="R0">Razlog 1</Reason>
        <Reason Id="R1">Razlog 2</Reason>
    </Header>
    <Entity Id="C=99" IdType="C" Action="A2" Reason="R1" TrfSucceed="true" TrfRefId="IMiSARC-1-
    192749948393985769347589569493874 85"/>
    <Entity Id="C=99^C=01" IdType="C" Action="A0" Reason="R1"/>
    <Entity Id="C=99^C=02" IdType="C" Action="A0" Reason="R1"/>
</Review>
```

V postopku odločanja se je zgodilo naslednje:

- *entiteti »C=99^C=01« in »C=99^C=02« sta označeni za uničenje (vrednost atributa »Action« je »A0, kar je referenca na akcijo »Dispose«);*
- *entiteta »C=99« je bila uspešno prenesena (atribut »TrfSucceed« ima vrednost »true«). Vnesla se je tudi referenca na preneseno entiteto.*

3.7.5.3 Sintaksa poročil

Poročila vsebujejo informacije o entiteti, izbrani akciji in informacijo, ali je bila izbrana akcija uspešno izvedena.

3.7.5.3.1 Etiketa »Report«

Etiketa predstavlja korenski element in vsebuje elemente iz spodnje tabele.

Ime XML elementa	Tip XML elementa	Obveznost
Header	Etiketa	Da
Entity	Etiketa	Ne

Tabela 32: Elementi etikete "Report"

3.7.5.3.2 Etiketa »Header«

Etiketa vsebuje verzijo XML dokumenta in akcije, ki si bile izbrane za posamezno entiteto. Elementi etikete so opisani v spodnji tabeli.

Ime XML elementa	Tip XML elementa	Obveznost
Version	Atribut	Da
Action	Etiketa	Da

Tabela 33: Elementi etikete "Header"

Etiketa »Action«

Etikete predstavljajo nabor akcij, ki so na voljo. Vsebujejo obvezen »Id« atribut, ki predstavlja referenco na akcijo, ki je izbrana za posamezno entiteto v postopku izvajanja odbiranja in izločanja.

3.7.5.3.3 Etiketa »Entity«

Poleg atributov, ki se nanašajo na entiteto, etiketa vsebuje tudi izbrano akcijo in informacijo, ali je bila akcija uspešno izvedena. Elementi etikete so opisani v spodnji tabeli.

Ime XML elementa	Tip XML elementa	Obveznost
Id	Atribut	Da
IdType	Atribut	Da
Action	Atribut	Da
Error	Atribut	Ne
Message	Etiketa	Ne

Tabela 34: Elementi etikete "Entity"

Atributa »Id« in »IdType« predstavljata identifikator entitete in tip identifikatorja in sta enaka kot v dokumentu samo za branje (glej poglavje 3.7.5.1 Sintaksa dokumentov samo za branje).

Atribut »Action«

Vrednost atributa je referenca na izbrano akcijo v etiketi »Header«.

Atribut »Error«

Če je atribut prisoten ima vrednost »True«. To pomeni, da izbrana akcija na entiteti ni bila uspešno izvedena. Vzrok za napako je opisan v znački »Message«.

Atribut »Message«

Če je značka prisotna vsebuje opis napake, ki se je zgodil na strežniku pri izvedbi akcije.

Primer:

```
<Report>
    <Header Version="1">
        <Action Id="A0">Dispose</Action>
        <Action Id="A1">Permanent</Action>
        <Action Id="A2">Transfer</Action>
        <Action Id="A3">Review</Action>
    </Header>
    <Entity Id="C=01^C=05" IdType="C" Action="A0"/>
    <Entity Id="C=01^C=06" IdType="C" Action="A0" Error="true">
        <Message>Access denied. Insufficient rights to disposed the entity.</Message>
    </Entity>
</Report>
```

Iz poročila je razvidno, da je bila entiteta »C=01^C=05« uspešno uničena, pri entiteti »C=01^C=06« pa je prišlo do napake, saj uporabnik, v imenu katerega se je izvajal postopek odbiranja in izločanja nima pravice brisanja na tej entiteti.

3.8 Imeniške storitve

Imeniške storitve omogočajo delo z strežniškim imenikom, ki vsebuje registrirane uporabnike in uporabniške skupine. So pomemben del strežniške infrastrukture, saj nudijo podporo za naslednje postopke in operacije:

- preverjanje istovetnosti (angl. Authentication);
- pridobivanje informacij o registrirani entiteti v strežniškem imeniku preko atributov;
- upravljanje z seznamom dostopnih pravic (angl. Access Control List - ACL).

Preverjanje istovetnosti

Preverjanje istovetnosti je postopek, ki se izvede pri vzpostavitvi uporabniške seje.
Če je preverjanje uspešno, imeniška storitev potrdi identiteto uporabnika,
ki se povezuje na strežnik IMiS®/ARChive Server. Identiteta uporabnika je potrjena,
ko se uporabniške poverilnice (angl. User credentials) ujemajo s podatki v strežniškem imeniku.
Poverilnice so preprosto rečeno skupina informacij, ki nedvoumno zagotovijo, da so avtentikacijski podatki, uporabljeni za izdelavo podatkov poverilnice enaki tistim, ki so bili uporabljeni za izdelavo podatkov v strežniški podatkovni bazi poverilnic.

Tako nek ključ ali geslo z matematičnimi postopki izdela podatke poverilnice, ki strežniku zagotovijo, da so izdelane s ključem, ki ga je uporabnik uporabil pri zadnji spremembi poverilnic.

Pridobivanje informacij preko atributov

Imeniške storitve omogočajo pridobivanje informacij o registriranih entitetah v strežniškem imeniku preko atributov. Pridobivajo se na podlagi unikatnega 32-bitnega identifikatorja, ki je dodeljen vsaki registrirani entiteti v strežniškem imeniku.
Na podlagi identifikatorja lahko pridobimo informacije kot so uporabniški račun, ime in priimek uporabnika, opis in drugi podatki imeniške entitete. (glej poglavje 3.2.1 Vrste atributov)

Upravljanje s seznamom dostopnih pravic

Seznam dostopnih pravic prav tako uporablja 32-bitni unikatni identifikator, kamor so vezane pravice in vloge. Imeniške storitve so povezovalni člen med entiteto v imeniku in pripadajočim unikatnim identifikatorjem.

Istočasno tudi razrešujejo uporabniške skupine, ki jim posamezni uporabniki pripadajo.
Te podatke uporablja ACL pri računanju efektivnih pravic (glej poglavje 3.3.5.2.6 Efektivne pravice v ACL).

3.8.1 Tipi

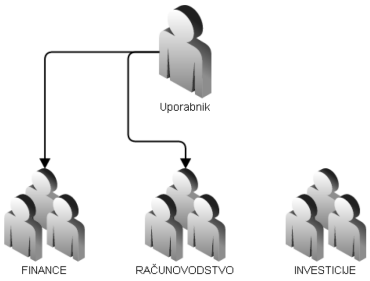
Strežniški imenik podpira dva tipa entitet:

- uporabnike
- uporabniške skupine.

3.8.1.1 Uporabniki

Uporabniki so osebe, ki dostopajo do informacij na strežniku. Uporabnike lahko poljubno umeščamo v uporabniške skupine brez omejitev.

Primer: Uporabnik pripada dvema skupinama: »FINANCE« in »RAČUNOVODSTVO«. Administrator lahko za posameznega uporabnika določi, ali je uporabnik aktiven ali neaktiven. Neaktivnim uporabnikom je vzpostavljanje seje s strežnikom onemogočeno.

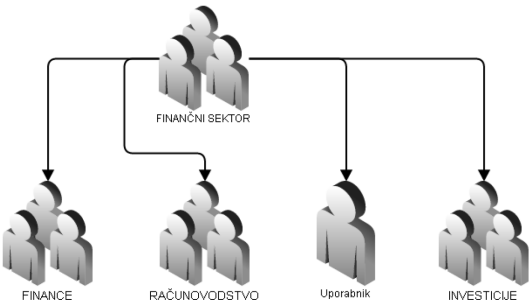


Slika 27: Pripadnost uporabnika v uporabniških skupinah

3.8.1.2 Uporabniške skupine

Uporabniške skupine so imeniške entitete, ki so jim dodeljene skupinske pravice na strežniku. S tem na lažji način kontroliramo dostop do posameznih dokumentov, razredov in zadev za večje število uporabnikov. Uporabniške skupine lahko poljubno gnezdimo, tudi rekurzivno. V primeru, da se pri razreševanju uporabniških skupin zazna rekurzija, potem se taka skupina preskoči, saj je njena vsebina že razrešena.

Primer: V uporabniški skupini »FINANČNI SEKTOR« so gnezdene skupine »FINANCE«, »RAČUNOVODSTVO« in »INVESTICIJE«, hkrati pa vsebuje tudi enega uporabnika.



Slika 28: Primer gnezdenja uporabniških skupin

3.8.2 Sistemsko določene entitete

Strežnik IMiS®/ARChive Server ima v svojem imeniku vnaprej določene naslednje sistemske ali navadne entitete:

- sys:Administrator
- sys:Administrators
- sys:CurrentUser
- sys:Server
- sys:Everyone
- Anonymous.

Entiteta »sys:Administrator«

Imeniška entiteta predstavlja uporabnika, lokalnega administratorja s polnimi dostopnimi pravicami. Privzeto se nahaja v uporabniški skupini »sys:Administrators«.
Na strežniku je prepovedano nastavljanje vsakršnih pravic, povezanih s to entiteto.

Entiteta »sys:Administrators«

Imeniška entiteta predstavlja uporabniško skupino s polnimi dostopnimi pravicami na strežniku. Prav tako kot za uporabnika »sys:Administrator«, je tudi za skupino prepovedano nastavljanje vsakršnih pravic, povezanih s to entiteto.
Če se uporabnik nahaja v tej uporabniški skupini, pravice te skupine prevladajo nad vsemi pravicami ostalih skupin v katerih se isti uporabnik še nahaja.

Entiteta »sys:CurrentUser«

Imeniška entiteta predstavlja abstrakcijo uporabnika v strežniškem sistemu. Identifikator »sys:CurrentUser« se v postopku ustvarjanja nove entitete (razreda, zadeve ali dokumenta) zamenja z identifikatorjem uporabnika, ki ustvarja entiteto.
S tem uporabnik prevzame dovoljenja in prepovedi, ki postanejo eksplicitne za uporabnika na tej entiteti.

Entiteta »sys:Server«

Imeniška entiteta predstavlja strežniškega uporabnika, v imenu katerega se izvajajo strežniške storitve, kot je na primer servis za dolgoročno arhiviranje vsebin. Strežniškega uporabnika od drugih uporabnikov loči to, da se ga ne da uporabiti za logiranje na strežnik ter izvajanje operacij v njegovem imenu.

Entiteta »sys:Everyone«

Imeniška entiteta predstavlja uporabniško skupino, katere člani so vsi imeniški uporabniki, članstva v skupini pa se ne sme spreminjati. Za razliko od drugih sistemskih skupin se za to skupino lahko spreminjajo dostopne pravice, kar posledično pomeni, da se dostopne pravice spreminjajo tudi za vse imeniške uporabnike.

Entiteta »Anonymous«

Imeniška entiteta predstavlja uporabnika, ki se na strežnik povezuje s starejšimi odjemalci, ki ne podpirajo avtentikacije uporabnika. Takega uporabnika strežnik obravnava kot anonimnega in ima take dostopne pravice, kot so nastavljene za Anonymous entiteto.

Za primer vzemimo ustvarjanje entitete na podlagi predloge, ki ima v listi dostopnih pravic (ACL) za uporabnika »sys:CurrentUser« nastavljeno pravico urejanja (pravici branja in pisanja).
V postopku ustvarjanja entitete s to predlogo se iz njenega ACL prenesejo te pravice in se dodajo na entiteto (kot eksplicitne) za trenutnega uporabnika, ki to entiteto ustvarja.
Tako bo imel tak uporabnik zmeraj pravico urejati entitete, ki jih je sam ustvaril.

3.8.3 Komponente entitete

Komponente entitet v imeniških storitvah določajo lastnosti in tip entitet (uporabnike ali uporabniške skupine). Komponente z njihovimi opisi so predstavljene v nadaljevanju.

»Id«

32-bitni unikatni identifikator, ki se samodejno dodeli vsaki entiteti v postopku njenega ustvarjanja in je nespremenljiv. Identifikator se uporablja tako za pridobivanje dodatnih informacij preko atributov, kot tudi pri računanju efektivnih dostopnih pravic v ACL.

»Type«

8-bitna vrednost, ki predstavlja tip entitete v imeniških storitvah.
Veljavni vrednosti sta naslednji:

- TYPE_GROUP: Vrednost predstavlja uporabniško skupino
- TYPE_USER: Vrednost predstavlja uporabnika.

»Account«

Znakovni niz je unikatna vrednost, ki predstavlja uporabniški račun ali ime uporabniške skupine. Vrednost se lahko med življenjskim ciklom entitete spreminja. Največja velikost znakovnega niza je omejena s 256 zlogi UTF-8 znakov.

»FirstName«

Znakovni niz predstavlja ime uporabnika. Vrednost ni unikatna in se lahko med življenjskim ciklom entitete spreminja. Največja velikost znakovnega niza je omejena s 256 bajti UTF-8 znakov.

»LastName«

Znakovni niz predstavlja priimek uporabnika. Vrednost ni unikatna in se lahko med življenjskim ciklom entitete spreminja.
Največja velikost znakovnega niza je omejena s 256 zlogi UTF-8 znakov.

»Description«

Znakovni niz predstavlja opis entitete. Vrednost ni unikatna in se lahko med življenjskim ciklom entitete spreminja.
Največja velikost znakovnega niza je omejena s 256 zlogi UTF-8 znakov.

»Email«

Znakovni niz predstavlja elektronsko pošto uporabnika ali uporabniške skupine.
Vrednost ni unikatna in se lahko med življenjskim ciklom entitete spreminja.
Največja velikost znakovnega niza je omejena s 512 zlogi UTF-8 znakov.

»Flags«

32 bitna ne-predznačena vrednost opisuje lastnosti entitete:

- Enabled: določa ali je uporabniški račun omogočen ali onemogočen;
- Deleted: določa ali je uporabnik ali skupina izbrisana;
- Locked: določa ali je uporabniški račun zaklenjen zaradi preseženega števila neuspešnih prijav;
- AdvancedAuthenticationEnabled: določa, ali uporabniški račun omogoča avtentikacijo preko napredne avtentikacijske metode;
- PreSharedKeyAuthenticationEnabled: določa, ali uporabniški račun omogoča avtentikacijo preko napredne avtentikacijske metode s šifriranjem preko deljenega ključa;
- SRP6aAuthenticationEnabled: določa, ali uporabniški račun omogoča avtentikacijo preko SRP-6a avtentikacijske metode.
- ExternalAuthenticationEnabled: določa, ali uporabniški račun omogoča zunanjo avtentikacijo z LDAP in/ali Kerberos.

»AuthenticationType«

Številčna vrednost predstavlja tip avtentikacije, ki jo uporabnik lahko izbere za avtentikacijo na strežniku. Trenutno podprte vrednosti so:

- TYPE_SRP6A predstavlja avtentikacijo na strežniku po SRP protokolu (angl. Secure Remote Password protocol);
- TYPE_EXTERN predstavlja zunanje avtentikacijske mehanizme kot sta LDAP in Kerberos.

»SRP6Acredentials«

Entiteti, ki predstavlja uporabnika se lahko dodeli poverilnico, ki je osnova za preverjanje avtentikacije uporabnika na strežniku. Parametra poverilnice sta znakovni niz v UTF-8 obliki, ki predstavlja uporabniško geslo in številčna vrednost, ki predstavlja SRP skupino.
Slednja določa uporabo praštevil v SRP protokolu (glej poglavje 3.8.5.1 SRP).

»SecurityClass«

Številčna vrednost predstavlja stopnjo tajnosti, ki je nastavljena uporabniku ali uporabniški skupini. Ta določa dostop do entitet z nižjo ali enako stopnjo tajnosti (glej poglavje 3.3.5 Dostopi).

3.8.4 Sinonimi

Strežnik omogoča uporabo sinonimov uporabniških računov. Funkcionalnost je uporabna, ko želimo povezati arhivski sistem z obstoječimi sistemi za avtentikacijo tretjih ponudnikov. Entiteti imenika lahko pripišemo poljubno število sinonimov, ki jih lahko uporabniki uporabljajo pri prijavi. Sinonim mora biti med sinonimi unikaten, prav tako ne sme ustrezati nobeni vrednosti atributa imeniške entitete »Account«.

3.8.5 Avtentikacija

Avtentikacija uporabnika je del postopka, ki se izvede pri vzpostavljanju uporabniške seje z arhivskim strežnikom. Strežnik podpira dve vrsti avtentikacije:

- Avtentikacija s uporabo PAKE (angl. Password-Authenticated Key Agreement) protokola (SRP). Avtentikacijo s PAKE protokolom lahko uporabljajo vsi uporabniki (lokalni in sinhronizirani), če imajo poverilnice shranjene v podatkovni bazi strežnika.
  V postopku avtentikacije IMiS®/ARChive Server preveri, ali se poverilnice, ki jih je poslal uporabnik ujemajo s poverilnicami, shranjenimi v podatkovni bazi strežnika.;

- Avtentikacija z uporabo zunanjih servisov (Active Directory ...).
  V postopku avtentikacije zunanji servis preveri poverilnice ter sporoči strežniku, ali se poverilnice ujemajo ali ne.

Če se ujemajo, je uporabnik dokazal istovetnost in lahko opravlja operacije na strežniku skladno z njegovimi pravicami. Če se poverilnice ne ujemajo, strežnik zavrne uporabniško sejo, saj uporabnik ni dokazal svoje verodostojnosti.

3.8.5.1 Avtentikacija z uporabo PAKE protokola

Za postopek avtentikacije strežnik uporablja protokol SRP, ki je razširitev protokola PAKE (angl. Password-Authenticated Key Agreement).
PAKE protokol je interaktivna metoda, ki omogoča dvema ali več stranem, da vzpostavijo varni komunikacijski kanal, brez uporabe infrastrukture javnih ključev (PKI), samo s poznavanjem gesla (http://en.wikipedia.org/wiki/Password-authenticated_key_agreement).

PAKE protokol mora zadostovati naslednjim varnostnim zahtevam:

- odpornost na pasiven napad s slovarjem (angl. Off-line dictionary attack resistence);
- odpornost na aktiven napad s slovarjem (angl. On-line dictionary attack resistence);
- prihodnjo tajnost (angl. Forward secrecy);
- zaščita pred vplivanjem na druge seje (angl. Known-session security).

Odpornost na pasiven napad s slovarjem

Napadalec je lahko pasiven (lahko samo opazuje protokol) ali aktiven (aktivno spreminja podatke, prenesene po protokolu). V obeh primerih se po protokolu ne sme pošiljati nobenih podatkov (prstnih odtisov, gesla, ...), kar bi napadalcu omogočalo razkritje gesla s pomočjo mavričnih tabel (angl. Rainbow tables) ali z uporaba napada z grobo silo (angl. Brute force attack).

Odpornost na aktiven napad s slovarjem

V primeru, da je napadalec aktiven (aktivno spreminja podatke, prenesene po protokolu), je praktično nemogoče napadalcu preprečiti, da bi naključno ugibal geslo. Kljub temu mora PAKE protokol zagotoviti, da je storjena najmanjša možna škoda v primeru, da napadalcu napad uspe (v primeru da je napad uspešen lahko napadalec ugane natanko eno geslo). Take vrste napadov se da preprosto zaznati na strežniku, ter jih preprečiti (npr. z zavrnitvijo vzpostavljanja seje z določene IP številke po določenem številu neuspelih poizkusov avtentikacije).

Prihodnja tajnost

Ker ni zagotovila, da bo geslo ostalo dolgoročno tajno, mora PAKE protokol zagotoviti zaščito preteklih sejnih ključev, tudi če se geslo razkrije. Poleg tega mora protokol tudi zagotoviti, da v primeru, ko napadalec pasivno opazuje izmenjavo ključev (angl. Key exchange), ne more uganiti sejnega ključa kljub temu, da pozna geslo.

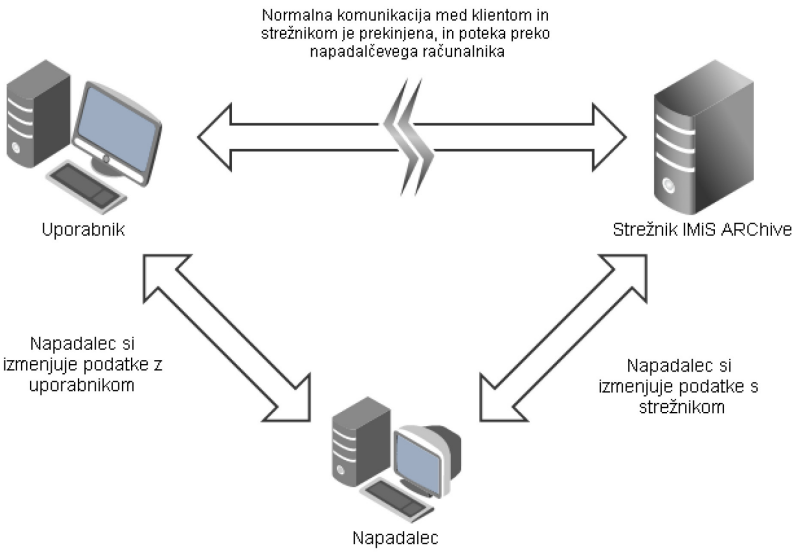
Zaščita pred vplivanjem na druge seje

Če napadalcu uspe napad in vzpostavi sejo s strežnikom v imenu nekoga drugega (angl. Impersonation attack) se predvideva, da napadalec pozna celoten postopek vzpostavljanja seje (vključno z vsemi vmesnimi koraki, ki jih izvajata strežnik in odjemalec in niso javno znani, oziroma se ne pošiljajo po protokolu).

PAKE protokol mora omogočiti, da kompromitacija seje ne ogrozi varnosti drugih, že vzpostavljenih sej. Več informacij o PAKE protokolu je na voljo na spletnem naslovu https://eprint.iacr.org/2010/190.pdf.

3.8.5.1.1 SRP

SRP protokol je razširjena različica PAKE protokola, ki preprečuje vrinjenemu napadalcu (angl. Man-in-the-middle) pridobivanje informacij. S temi informacijami bi lahko z grobo silo uganil geslo, ne da bi aktivno spreminjal podatke, ki si jih izmenjujeta strežnik in odjemalec.



Slika 29: Primer napada na IMiS®/ARChive Server z vrinjenim napadalcem

Lastnosti SRP protokola so naslednje:

- Omogoča avtentikacijo na strežnik.
- Odporen je na napad s slovarjem.
- Ne potrebuje infrastrukture javnih ključev ter posledično ni potrebe po zaupanja vrednih tretjih osebah (angl Trusted third party), kot so izdajatelji digitalnih potrdil.
- Odjemalec posreduje strežniku t.i. ničelni dokaz o poznavanju gesla (angl. Zero-knowledge password proof). To je interaktivna metoda, pri kateri odjemalec dokaže strežniku, da pozna vrednost gesla, ne da bi strežniku poslal kakršenkoli dokaz (prstni odtis gesla, ...) o vrednosti le-tega.

SRP (verzija 6) se uporablja pri /v:

- zagotavljanju varnosti transportnega sloja (angl. Transport layer security – SRP verzijo opisuje povezava http://en.wikipedia.org/wiki/TLS-SRP);
- EAP protokolu (angl. Extensible authentication protocol –RFC 3748);
- SAML protokolu.

SRP je poleg tega še del standardov IETF (RFC 2944, RFC 2945, RFC 5054), IEEE (P1363.2) in ISO (IEC 11770-4).

3.8.5.1.2 Protokol

Vse matematične operacije v SRP protokolu so omejene na operacije, ki so določene znotraj matematičnega obroča (angl. Mathematical ring - http://en.wikipedia.org/wiki/Ring_%28mathematics%29).

Parametri protokola so naslednji:

- *N* – varno praštevilo, ki mora ustrezati formuli: $N = 2 \times q + 1$, kjer je *q* praštevilo. Praštevilo N določa velikost matematičnega obroča in je javno znana vrednost;
- *g* – generator multiplikativne skupine (kombinacije g in N so opisane v RFC 5054 - http://www.tools.ietf.org/html/rfc5054);
- *H()* – zgoščevalna funkcija;
- *k* – množiteljski parameter, ki ustvari asimetrijo v SRP protokolu in s tem pripomore k odpornosti na aktiven napad;

- *s* – naključna vrednost imenovana sol (angl. Salt, ki se uporablja pri ustvarjanju prstnega odtisa gesla; http://en.wikipedia.org/wiki/Salt_%28cryptography%29);
- *I* – uporabniško ime;
- *p* – geslo v čistopisu;
- *x* – privatni ključ;
- *v* – strežniški verifikator;
- *u* – mešalni parameter (angl. Scrambling parameter);
- *a, b* – naključni števili, ki predstavljata kratko veljavna ključa;
- *A, B* – javna parametra;
- *S* – sejni ključ;
- *K* – varen sejni ključ, ki je rezultat uspešne avtentikacije;
- M_1 – dokaz odjemalca o poznavanju gesla;
- M_2 - dokaz strežnika o poznavanju gesla.

Predpogoj za začetek avtentikacije s SRP protokolom je, da ima strežnik shranjen verifikator »*v*«, s katerim bo preveril vsebino uporabniškega gesla. Za ustvarjanje verifikatorja je potreben privatni ključ »*x*«, ki se izračuna na naslednji način:

$$x = H(s, p)$$

Pri računanju privatnega ključa uporabimo naključno vrednost »*s*« in geslo v čistopisu. Vrednost in geslo združimo in izračunamo prstni odtis, ki predstavlja vrednost privatnega ključa.
Nato izračunamo strežniški verifikator po formuli:

$$v = g^x$$

Strežnik IMiS®/ARChive Server shrani verifikator skupaj z naključno vrednostjo »*s*«, privatni ključ pa uniči, saj ga ne potrebuje več. Izračunan verifikator (skupaj z naključno vrednostjo) tako pripada natančno določenemu uporabniku, zato je potrebna povezava med verifikatorjem in uporabnikom, kateremu ta pripada. To funkcijo na strežniku opravlja imenik, oziroma imeniške storitve.

Sedaj se lahko izvede postopek avtentikacije. Najprej odjemalec in strežnik izračunata množiteljski parameter iz praštevila in generatorja multiplikativne skupine:

$$k = H(N, g)$$

Odjemalec začne avtentikacijo tako, da z varnim generatorjem naključnih števil ustvari naključno število »*a*« in na podlagi tega izračuna vrednost parametra »*A*«.
Nato uporabniško ime »*k*« skupaj s parametrom »*A*« pošlje strežniku.
Parameter »*A*« se izračuna po naslednji formuli:

$$A = g^a$$

Strežnik prejme uporabniško ime in parameter »*A*«, Nato s pomočjo imeniških storitev pridobi pripadajoč verifikator »v« in naključno vrednost »*s*«. Z varnim generatorjem naključnih števil izračuna vrednost »*b*« in na podlagi tega izračuna parameter »*B*«.
Nato se parameter »*B*« skupaj z naključno vrednostjo »*s*« pošlje strežniku.
Parameter »*B*« se izračuna po naslednji formuli:

$$B = k \times v + g^b$$

Po prvi izmenjavi parametrov obe strani preverita veljavnost parametrov »*A*« in »*B*«.
Parametra sta veljavna, če ustrezata naslednjim kriterijem:

$A \neq 0 \ (mod\ N)$ - preverjanje na strežniku

$B \neq 0 \ (mod\ N)$ - preverjanje na odjemalcu

Če katerikoli izmed parametrov ne ustreza kriterijem, potem se postopek avtentikacije prekine.

Če parametra ustrezata kriterijem, se postopek avtentikacije nadaljuje s tem, da odjemalec in strežnik izračunata mešalni parameter »*u*« po naslednji formuli:

$$u = H(A, B)$$

Mešalni parameter mora ustrezati kriteriju $u \neq 0$, sicer se postopek avtentikacije prekine.
V nadaljevanju avtentikacije odjemalec in strežnik izračunata vsak svoj dokaz o poznavanju gesla in varen sejni ključ »*K*«.

<u>Računanje varnega sejnega ključa na odjemalcu</u>
Odjemalec iz vpisanega gesla »*p*« in od strežnika prejete naključne vrednosti »*s*«, izračuna privatni ključ »*x*« po že znani formuli $x = H(s, p)$. Na podlagi privatnega ključa se izračuna sejni ključ »*S*« in njegov prstni odtis »*K*«, ki predstavlja varen sejni ključ. Računanje varnega sejnega ključa poteka po naslednjih formulah:

$$S = (B - k \times g^x)^{(a + u \times x)}$$

$$K = H(S)$$

Računanje varnega sejnega ključa na strežniku

Strežnik izračuna varen sejni ključ po naslednjih formulah:

$$S = (A \times v^u)^b$$

$$K = H(S)$$

Odjemalec prvi pošlje strežniku svoj dokaz o poznavanju gesla in o varnem sejnem ključu. Dokaz je izračunan po naslednji formuli:

$$M_1 = H(H(N)\ xor\ H(g), H(I), s, A, B, K)$$

Strežnik po enaki formuli izračuna svoj primer dokaz M_1 tako, da v računanje vključi svojo vrednost varnega sejnega ključa »*K*«. Vrednosti dokazov se morata ujemati, sicer je prišlo do napake pri avtentikaciji in strežnik zavrne sejo.

Če se vrednosti ujemata, potem strežnik izračuna svoj dokaz po naslednji formuli in ga pošlje odjemalcu:

$$M_2 = H(A, M_1, K)$$

Odjemalec po enaki formuli izračuna svoj primer dokaza M_2, kjer vključi svoj varen sejni ključ »*K*«. Če se vrednosti dokazov ne ujemata, potem je prišlo do napake pri avtentikaciji in odjemalec zavrne sejo. Če se dokaza ujemata, je avtentikacija uspela. Poleg uspešne avtentikacije pa imata odjemalec in strežnik še varen sejni ključ, ki ga lahko uporabita za šifriranje podatkov in tako vzpostavita varen transportni kanal.

3.8.5.2 Avtentikacija z zunanjim servisom

Avtentikacijo z zunanjim servisom lahko uporabljajo vsi uporabniki, ki so bili sinhronizirani v lokalni imenik iz oddaljenega imenika.

Podprte avtentikacijske metode z zunanjim servisom so naslednje:

- LDAP (angl. Lightweight Directory Access Protocol);
- Kerberos.

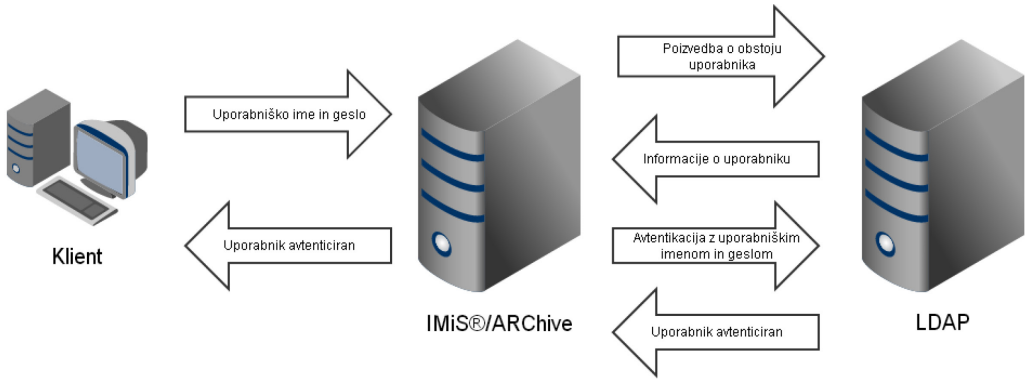
3.8.5.2.1 LDAP

LDAP je aplikacijski protokol za dostop in vzdrževanje porazdeljenih imeniških storitev preko internetnega protokola. Protokol omogoča omrežni dostop do podatkov uporabnikov, skupin, servisov ter drugih entitet na centralnem strežniku. Podroben opis protokola je opisan v RFC 4511 (https://www.ietf.org/rfc/rfc4511.txt).

Postopek avtentikacije preko LDAP protokola poteka na strežniku po naslednjih korakih:

1. odjemalec pošlje strežniku IMiS®/ARChive Server uporabniško ime in geslo za avtentikacijo;
2. strežnik IMiS®/ARChive Server se poveže na LDAP strežnik ter naredi poizvedbo za uporabnika z uporabniškim imenom, ki ga je odjemalec poslal strežniku;
3. če uporabnik na LDAP strežniku obstaja, ga strežnik IMiS®/ARChive Server poizkuša avtenticirati z uporabniškim imenom in geslom, ki ga je odjemalec poslal strežniku;
4. uporabnik je avtenticiran, če je poslal pravilne poverilnice za LDAP strežnik.

Če se katerikoli izmed korakov ne izvede oz. se izvede z napako, potem uporabnik ni avtenticiran in seja s strežnikom IMiS®/ARChive Server ni vzpostavljena. Naslednja slika prikazuje uspešen postopek avtentikacije z uporabo LDAP strežnika.



Slika 30: LDAP avtentikacija

LDAP protokol (verzija 3) podpira naslednje tipe avtentikacije:

- anonimni dostop (angl. Anonymous); pri povezavi na strežnik se avtentikacija ne izvede;
- preprosta avtentikacija (angl. Simple); uporabniško ime in geslo se pošljeta v čistopisu;
- SASL avtentikacija (angl. Simple Authentication and Security Layer); protokol uporablja metodo »poziv-odziv« (angl. Challenge-response) za avtentikacijo ter vzpostavitev varne povezave za prenos podatkov.

Ker se uporabniško ime in geslo pri LDAP avtentikaciji pošiljata po mreži je priporočljivo, da se pošiljata po zavarovanem kanalu (angl. Secure Sockets Layer). S tem se izognemo napadu z vrinjenim napadalcem. Zavarovan kanal mora potekati med odjemalcem in strežnikom IMiS®/ARChive Server. V kolikor se za komunikacijo med strežnikom IMiS®/ARChive Server in LDAP strežnikom uporablja anonimni dostop ali preprosta avtentikacija, je priporočeno imeti zavarovan kanal tudi med njima.

3.8.5.2.2 Kerberos

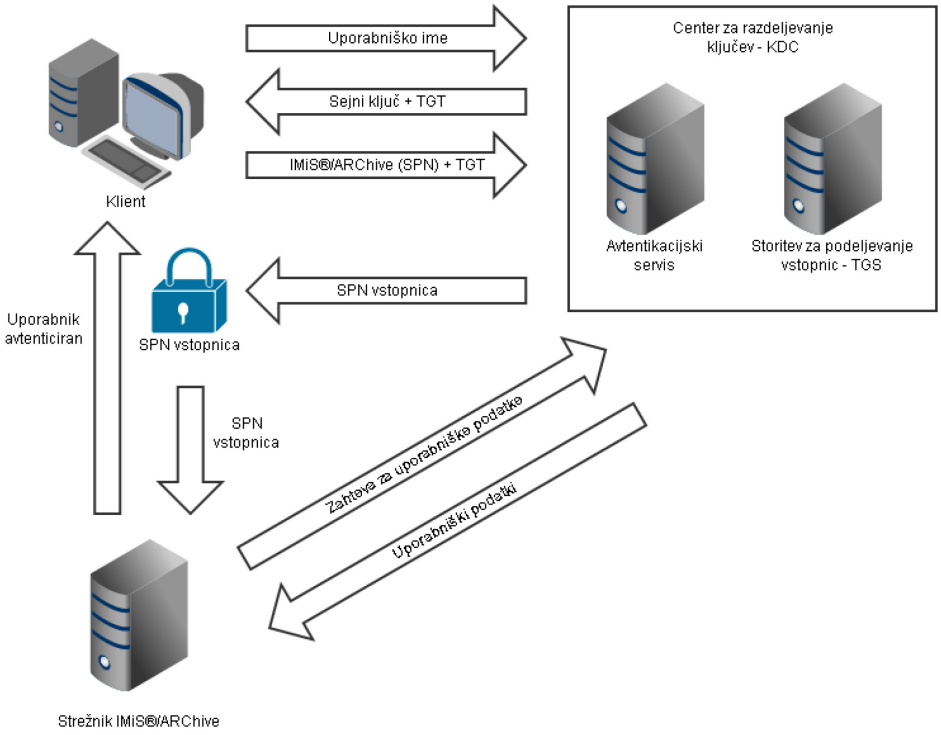
Kerberos je omrežni avtentikacijski protokol, ki deluje na principu vstopnic (angl. tickets). Slednje omogočajo varno avtentikacijo po nezavarovanem mrežnem kanalu ter medsebojno avtentikacijo (angl. Mutual Authentication). To pomeni, da lahko odjemalec in strežnik, ki sodelujeta v avtentikacijskem procesu preverita identiteto drug drugega.
Za delovanje protokola je potrebna t.i. zaupanja vredna tretja stran (angl. Trusted Third Party). Ponavadi je to strežnik, ki izdaja s simetričnim ključem šifrirane vstopnice. Protokol je odporen proti prisluškovanju (angl. Eavesdropping) in napadom s ponovitvijo (angl. Replay Attack). Podrobno je obrazložen v specifikaciji RFC 4120 (https://www.ietf.org/rfc/rfc4120.txt).

Zaupanja vredna tretja stran je sestavljena iz avtentikacijske storitve (angl. Authentication Service) in storitve za podeljevanje vstopnic (angl. Ticket Granting Service ali TGS).
Obe storitvi skupaj sestavljata center za razdeljevanje ključev (angl. Key Distribution Center ali KDC). Avtentikacija klienta na IMiS®/ARChive strežnik po Kerberos protokolu poteka na naslednji način:

1. Uporabnik vpiše uporabniško ime in geslo v odjemalca, ki nato pošlje uporabniško ime v čistopisu centru za razdeljevanje ključev (KDC). Slednji preveri ali uporabnik obstaja in odjemalcu pošlje šifriran odgovor. Če je uporabnik pravilno vpisal geslo odjemalec odgovor dešifrira. Dešifrirani podatki vsebujejo TGS sejni ključ ter šifrirano vstopnico za podeljevanje vstopnic (angl. Ticket Granting Ticket ali TGT). Sejni ključe se nadalje uporablja za komunikacijo s TGS., TGT vstopnica pa se uporablja za zahteve po servisnih vstopnicah. Če je odjemalec uspešno dešifriral odgovor, je pridobil sejni ključ ter TGT. To je dovolj, da uspešno konča avtentikacijo.
2. Po uspešni avtentikaciji odjemalec zahteva vstopnico za strežnik IMiS®/ARChive Server. Strežnik na KDC pošlje zahtevek, ki vsebuje TGT vstopnico ter servisni indentifikator (angl. Service Principal Name ali SPN). Slednji predstavlja strežnik IMiS®/ARChive Server. KDC preveri obstoj SPN in pošlje šifrirano SPN vstopnico odjemalcu.

3. Odjemalec pošlje strežniku IMiS®/ARChive Server prejto SPN vstopnico.
4. Strežnik IMiS®/ARChive Server preveri veljavnost vstopnice tako, da jo dešifrira s ključi, ki so bili ustvarjeni ob registraciji SPN na KDC. Če dešifriranje vstopnice uspe, potem vstopnica pripada trenutnemu SPN. Naknadno strežnik preveri, ali se uporabnik ujema z uporabnikom, za katerega je bila SPN vstopnica narejena (podatke o uporabniku pridobi iz KDC). Če se uporabnik ujema pomeni, da je uporabnik uspešno avtenticiran na KDC. Prav tako je uspešno pridobil vstopnico za SPN kar pomeni, da se lahko avtenticira tudi na IMiS®/ARChive strežniku.

Kerberos protokol omogoča tudi t.i enkratni vpis (angl. Single Sign On ali SSO).
Uporabnik se v domeno prijavi le enkrat. Ob prijavi pridobi TGT vstopnico, s katero zahteva vstopnice za različne SPN, ne da bi se od uporabnika zahtevalo ponovno vpisovanja gesla.
V primeru SSO vpisa se prvi korak pri avtentikaciji izpusti.
Naslednja slika prikazuje primer uspešne avtentikacije na strežnik IMiS®/ARChive Server z uporabo Kerberos protokola.



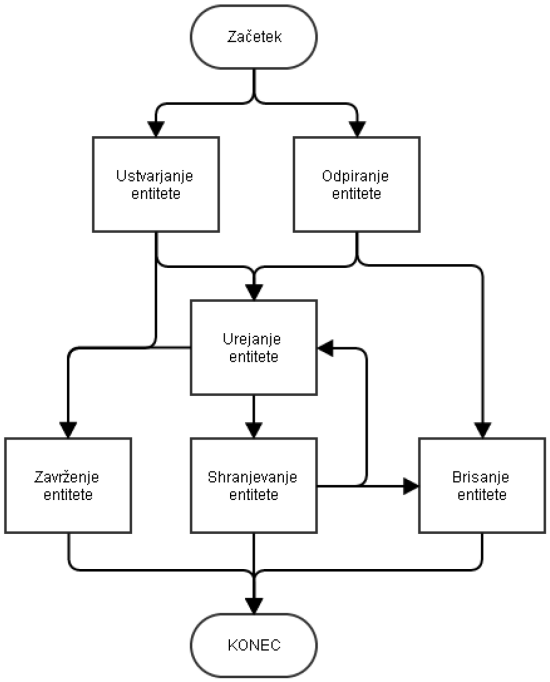
Slika 31: Kerberos avtentikacija

Pri Kerberos avtentikaciji je potrebno omeniti, da so vstopnice časovno omejene.
Zato je potrebno poskrbeti za časovno usklajenost KDC in SPN servisa. V nasprotnem primeru lahko pride do težav pri preverjanju SPN vstopnic ter posledično do avtentikacije.

3.8.6 Življenjski cikel entitet imenika

Življenjski cikel entitete v imeniških storitvah delimo na naslednje vrste postopkov:

- odpiranje entitete imenika
- ustvarjanje entitete imenika
- urejanje entitete imenika
- shranjevanje entitete imenika
- zavrženje entitete imenika (angl. Discard)
- brisanje entitete imenika.



Slika 32: Povezava med postopki v življenjskem ciklu entitete v imeniku

3.8.6.1 Odpiranje entitete imenika

Odpiranje entitete imenika je postopek pridobivanja primerka (angl. Instance) entitete iz imenika. Strežnik IMiS®/ARChive Server v postopku uvodne nastavitve (angl. Initialization) naloži vse entitete, ki se v imeniku nahajajo. Istočasno pri odpiranju entitete imenika preveri, če je entiteta predhodno naložena. V primeru da je naložena, vrne njen primerek, v nasprotnem primeru pa javi napako.

3.8.6.2 Ustvarjanje entitete imenika

Postopek ustvarjanja entitete imenika je prvi v življenjskem ciklu entitete. Na začetku se določi tip entitete in uporabniški račun ali ime uporabniške skupine, če gre za skupino.
V postopku urejanja lahko nastavimo še dodatne komponente glede na to, za kateri tip entitete gre. V primeru, da se novo ustvarjena entiteta imenika zavrže preden je le-ta shranjena, se vsebina imenika ne spremeni.

3.8.6.3 Urejanje entitete imenika

Urejanje entitete imenika je mogoče na novo ustvarjenih entitetah ter entitetah, ki so že v imeniku in niso v procesu sinhronizacije (glej poglavje 3.8.7 Sinhronizacija).
Pri urejanju entitet imenika se lahko nastavljajo in spreminjajo komponente,
ki so opisane v poglavju 3.8.3 Komponente entitete.

3.8.6.4 Shranjevanje entitete imenika

Shranjevanje entitete imenika je postopek kjer se vse vrednosti, ki smo jih ustvarili ali spremenili med urejanjem, shranijo v imenik.

3.8.6.5 Zavrženje entitete imenika

Do zavrženja entitete imenika pride, ko ne želimo shraniti novo ustvarjene entitete ali entitete v urejanju. V obeh primerih se vsebina imenika ne spremeni, saj v primeru urejanja entitete v imeniku ostane entiteta z zadnjimi shranjenimi vrednostmi, novo ustvarjena entiteta imenika pa pride v imenik le v primeru shranjevanja.

3.8.6.6 Brisanje entitete imenika

Uporabnik lahko briše entitete imenika le, če niso v procesu sinhronizacije (glej poglavje 3.8.7 Sinhronizacija). Brisanje entitete imenika poteka tako, da se entiteto v imeniku označi kot izbrisano, atributu »Account« se doda unikatni identifikator, ki omogoča ponovno registriranje entitete z isto »Account« vrednostjo, kot jo je imela entiteta pred izbrisom, fizično pa se entiteta ne izbriše. Tak uporabnik ali uporabniška skupina za strežnik ne obstaja več. Vseeno pa je potrebno ohraniti njihove 32-bitne identifikatorje za podporo atributov in zagotavljanja unikatnosti 32-bitnih identifikatorjev, uporabljenih pri računanju efektivnih pravic v ACL.

3.8.7 Sinhronizacija

Sinhronizacija internega strežniškega imenika je predpogoj za uporabo zunanjih avtentikacijskih metod, kot sta LDAP in Kerberos. Strežnik za sinhronizacijo uporablja koncept vtičnikov (glej poglavje 3.8.8 Vtičniki). Za vsakega ponudnika oddaljenih imeniških storitev (npr. Microsoft Active Directory, OpenLDAP, ...) uporabnik s pravico (administrator) registrira vtičnik in ga konfigurira za določenega ponudnika.

Za delovanje sinhronizacije so obvezni naslednji parametri:

- URL naslov do strežnika imeniške storitve.
- Uporabniško ime in geslo, v imenu katerega se bodo izvajale akcije na strežniku.
  Če je omogočen anonimni dostop, uporabniško ime in geslo nista potrebna).
- Osnovno razločevalno ime (angl. Base Distinguished Name ali BaseDN).
  S tem imenom se okvrino določi predel imenika, ki se bo sinhroniziral.
- Razločevalno ime uporabnikov (angl. User Distinguished Name ali UserDN) in razločevalno ime skupin (angl. Group Distinguished Name ali GroupDN) nista obvezna podatka.
  Uporabljata se v primeru, ko želimo dodatno segmentirati sinhronizacijo uporabnikov in skupin glede na osnovno razločevalno ime.
- Filtri za iskanje uporabnikov in skupin v imeniku.
- V primeru, da se za dostop do oddaljenega imenika uporablja zaščiten kanal, je potrebno določiti še pot do shrambe certifikatov za vzpostavitev zaščitenega kanala.

Primeri uporabe razločevalnih imen:

- *BaseDN = »DC=example,DC=com«, UserDN in GroupDN nista nastavljena.*
  Na osnovi nastavitev se bodo sinhronizirali vsi uporabniki in skupine, ki se nahajajo v BaseDN.
- *BaseDN = »DC=example,DC=com«, UserDN = »OU=Users«, GroupDN ni nastavljen.*
  Na osnovi nastavitev se bodo iskali vsi uporabniki pod »OU=Users,DC=example,DC=com«.
  Grupe se bodo iskale pod BaseDN, torej pod »DC=example,DC=com«.
- *BaseDN = »DC=example,DC=com«, UserDN = »OU=Users«, GroupDN = »OU=Groups«.*
  Na osnovi teh nastavitev se bodo iskali vsi uporabniki pod »OU=Users,DC=example,DC=com«, skupine pa pod »OU= Groups,DC=example,DC=com«.

Primeri preprostih iskalnih filtrov za Microsoft Active Directory:

- *»(&(objectCategory=person)(objectClass=user))«. Rezultat filtra so vsi objekti kategorije osebe in razreda uporabnik (vsi uporabniki).*
- *»(objectCategory=group)«. Rezultat filtra so vsi objekti kategorije skupine (vse skupine).*

Ob sinhronizaciji iz oddaljenega imenika v lokalni strežniški imenik se vse entitete označijo kot sinhronizirane. Obenem jih uporabnik s pravico (administrator) poveže še z ponudnikom imeniške storitve, s katero je bila entiteta sinhronizirana (glej poglavje 3.8.3 Komponente entitete). Sinhroniziranih entitet ni mogoče urejati in brisati, saj so del sinhronizacijskega procesa.

Lahko se jim spremeni samo stopnja tajnosti ter dodatne avtentikacijske metode (recimo SRP). Če želimo sinhronizirano entiteto spreminjati ali brisati, jo mora uporabnik najprej izključiti iz procesa sinhronizacije. Taka entiteta se v procesu sinhronizacije ne bo sinhronizirala do ponovne vključitve v sinhronizacijo.

3.8.8 Vtičniki

Vtičniki za imeniške storitve omogočajo sinhronizacijo ter avtentikacijo uporabnikov z zunanjimi servisi. Vtičnike lahko razdelimo v dve skupini:

- vtičnik za Active Directory
- generičen LDAP vtičnik.

Vtičnik za Active Directory omogoča sinhronizacijo in avtentikacijo (LDAP, Kerberos) uporabnikov z Microsoft Active Directory. Generičen LDAP vtičnik omogoča sinhronizacijo tudi z drugimi servisi (OpenLDAP...) ter samo LDAP avtentikacijo. Vtičnike se konfigurira preko xml zapisa, ki se nahaja znotraj »Arguments« etikete.

3.8.8.1 Konfiguracija vtičnika za Active Directory

V nadaljevanju so predstavljene tabela z etiketami, ki so skupne obema vtičnikoma ter tabela z etiketami za Active Directory vtičnik.

Ime etikete	Opis
Class	Ime razreda, kjer je vtičnik implementiran. Vrednosti: - com.imis.imisarc.server.aaa.impl.ActiveDirectory - com.imis.imisarc.server.aaa.impl.GenericLdapConnector
LdapURL	Pot do ldap/ldaps strežnika.
LdapUsername	Uporabnik, v imenu katerega bo vtičnik izvajal poizvedbe na strežniku.
LdapPassword	Uporabniško geslo.
LdapAuthenticationType	Tip ldap avtentikacije: NONE, SIMPLE, DIGEST-MD5, CRAM-MD5
LdapBaseDN	Osnovno razločevalno ime.

Ime etikete	Opis
LdapUserDN	Razločevalno ime uporabnikov.
LdapGroupDN	Razločevalno ime skupin.
LdapUserObjectFilter	Filter za iskanje uporabnikov (glede na osnovno razločevalno ime ter razločevalno ime uporabnikov).
LdapGroupObjectFilter	Filter za iskanje skupin (glede na osnovno razločevalno ime ter razločevalno ime skupin).
LdapReadTimeout	Časovna omejitev branja odgovora iz strežnika (vrednost v milisekundah).
LdapQueryTimeout	Časovna omejitev izvajanja poizvedbe na strežniku (vrednost v milisekundah).
LdapPageSize	Število rezultatov poizvedbe na eni strani pri uporabi ostranjevanja.
LdapRecursiveQueries	»True« za rekurzivne poizvedbe, sicer »false«.
SSLTSType	Tip shrambe digitalnih potrdil v primeru, da se uporablja ldap preko ssl.
SSLTSFile	Pot do datoteke, ki predstavlja shrambo digitalnih potrdil.
SSLTSPassword	Geslo za odklep shrambe digitalnih potrdil.
SSLProtocols	Podprti ssl protokoli (ločeni z vejico).
LdapAuthenticationFilter	Filter za ldap avtentikacijo uporabnikov.

Tabela 35: Skupne konfiguracijske etikete vtičnikov

Ime etikete	Opis
KrbServicePrincipal	Servisni indikator za Kerberos.
KrbKeytabLocation	Lokacija Kerberos keytab datoteke.
KrbDebugMode	»True« za Kerberos debug mode, drugače »False« (debug mode omogoča podroben zapis avtentikacije v log).

Ime etikete	Opis
AccountAliasList	Etiketa se uporablja za nastavitev Active Directory vrednosti za sinhronizacijo »Account« komponente ter sinonimov (glej poglavje 3.8.4 Sinonimi). Privzeto se »Account« vrednost entitete sinhronizira z vrednostjo »sAMAccountName« polja, sinonimi se ne sinhronizirajo. Vrednosti v etiketi so ločene z vejico. Prva vrednost predstavlja polje za sinhronizacijo z »Account« komponento, ostale vrednosti pa predstavljajo sinonime. *Primer:* *<AccountAliasList>sAMAccountName,userPrincipalName</AccountAliasList>* »Account« takega uporabnika se bo nastavil na vrednost »sAMAccountName« polja, njegov sinonim pa bo vrednost »userPrincipalName« polja.

Tabela 36: Konfiguracijske etikete Active Directory vtičnika

Primer konfiguracije vtičnika za izvajanje sinhronizacije z OpenLDAP strežnikom predstavlja naslednja xml struktura:

```
<Arguments>
  <Class>com.imis.imisarc.server.dir.impl.GenericLdapConnector</Class>
  <LdapURL>ldap://pot.do.streznika</LdapURL>
  <LdapUsername>cn=Manager,dc=my-domain,dc=com</LdapUsername>
  <LdapPassword>geslo</LdapPassword>
  <LdapAuthenticationType>simple</LdapAuthenticationType>
  <LdapBaseDN>dc=my-domain,dc=com</LdapBaseDN>
  <LdapUserDN>ou=Userji</LdapUserDN>
  <LdapGroupDN>ou=Grupe</LdapGroupDN>
  <LdapUserObjectFilter>objectClass=posixAccount</LdapUserObjectFilter>
  <LdapGroupObjectFilter>objectClass=posixGroup</LdapGroupObjectFilter>
  <LdapReadTimeout>12000</LdapReadTimeout>
  <LdapQueryTimeout>6000</LdapQueryTimeout>
  <LdapPageSize>256</LdapPageSize>
  <LdapRecursiveQueries>true</LdapRecursiveQueries>
</Arguments>
```

3.8.8.2 Konfiguracija generičnega LDAP vtičnika

V nadaljevanju je opisana konfiguracija generičnega LDAP vtičnika. Za pravilno delovanje generičnega LDAP vtičnika je potrebno konfigurirati preslikave med polji na LDAP strežniku in ključi, ki predstavljajo komponente imeniške entitete. Administrator lahko določi tudi svoje ključe, ki pa morajo biti unikatni. Če preslikava med ključi in komponentami imeniških entitet ni določena v konfiguraciji, vtičnik za tako preslikavo vrne privzeto vrednost. Preslikave se v konfiguraciji označijo z etiketo »Field«. Povezave med ključi in komponentami entitete so predstavljene v naslednji tabeli.

Ključ	Komponenta entitete	Privzeta vrednost
sys:dir:Account	Account	Prazen niz.
sys:dir:FirstName	FirstName	Prazen niz.
sys:dir:LastName	LastName	Prazen niz.
sys:dir:Description	Description	Prazen niz.
sys:dir:Email	Email	Prazen niz.
sys:dir:UUID	Vrednost je interna in se uporablja kot unikatni indentifikator za sinhronizacijo.	Prazen niz.
sys:dir:DistinguishedName	Vrednost je interna in se uporablja za razreševanje članstva v skupinah.	Prazen niz.
sys:dir:Flags:Enabled	Flags.Enabled	True.
sys:dir:Flags:Locked	Flags.Locked – vrednost se sinhronizira samo za uporabnike	False.
sys:dir:Aliases	Vrednost nastavlja sinonime (vrednost se sinhronizira samo za uporabnike - (glej poglavje 3.8.4 Sinonimi).	Prazen seznam vrednosti.
sys:dir:GroupMembers	Vrednost je interna in se uporablja pri sinhronizaciji in razreševanju članstva v skupinah (vrednost se uporablja samo pri sinhronizaciji skupin).	Prazen seznam vrednosti.

Tabela 37: Preslikave med ključi in komponentami entitete

Vsaki preslikavi moramo določiti tip preslikave (atribut »type«), le-ta pa naprej določa, kako se bodo podatki v preslikavi obdelali pred sinhronizacijo. Podatki se obdelajo glede na zaporedje rezerviranih vrednosti, ki je zapisana v atributu »typeExt« (ločeni so z vejico) in so predstavljene v naslednji tabeli.

Rezervirane vrednosti »typeExt«	Opis
MASK	Decimalna številska vrednost, ki se uporablja pri »maskiranju« (izvajanje AND operacije).
RDX	Osnova za številčno pretvorbo.
OFF	Predstavlja pozicijo bita za testiranje (začetni bit je na poziciji 0).
VAL	Seznam vrednosti, ki predstavljajo vrednost »true« (vrednosti so ločene z #).
NEG	»True« negira vrednost, drugače »false«.
DEL	Vrednost predstavlja ločitveni niz za ločevanje vrednosti v EXPRESSION tipu preslikave.

Tabela 38: Vrednosti "typeExt" atributa

Tabela v nadaljevanju prikazuje podprte tipe preslikav z njihovimi opisi ter vrstnim redom izvajanja obdelav (vrednosti atributa »typeExt«).

Tip preslikave	Vrednosti »typeExt«	Opis
STRING	/	Vrednost je predstavljena kot niz in se ne obdeluje.
BINARY	/	Vrednost se kodira v Base64.
BITFIELD	RDX, OFF (obvezna vrednost), NEG	Vrednost se obdela glede na vrednost »typeExt«, rezultat je bodisi »true« ali »false«.
BOOLEAN	VAL (obvezna vrednost), NEG	Vrednost se obdela glede na vrednost »typeExt«, rezultat je bodisi »true« ali »false«.

Tip preslikave	Vrednosti »typeExt«	Opis
INT	RDX, NEG, MASK	Vrednost se obdela glede na vrednost »typeExt«, rezultat je število v desetiški obliki.
EXPRESSION	DEL	Vrednost je sestavljena iz ključev preslikav (ključ preslikave se mora nahajati med %), ki se ovrednotijo.
LDAP_AUTHENTICATION	/	Vrednost se uporablja za LDAP avtentikacijo in ni predmet preslikave.
DISTINGUISHED_NAME	/	Vrednost predstavlja razločevalno ime in se ne obdeluje.

Tabela 39: Tipi preslikav

Vsaki preslikavi lahko z atributom »scope« določimo, ali velja za sinhronizacijo uporabnikov ali skupin. Če atribut ni določen, potem preslikava velja za sinhroniziranje uporabnikov in skupin. Naslednja tabela prikazuje uporabnike na OpenLDAP strežniku z atributi in njihovimi vrednostmi, ki so bili uporabljeni za primere konfiguracije preslikav.

Atribut »cn«	Atribut »telephoneNumber«	Atribut »postalCode«	Atribut »uidNumber«	Atribut »homeDirectory«
test4	00000010	FF	6666	/home/test4
test5	133546	1234321	63361	/home/test5
test6	54321	11112222	62784	/home/test6

Tabela 40: Primer uporabniških atributov in njihove vrednosti

Primer 1: Preprosta konfiguracija, kjer se »cn« preslika v »sys_dir:Account«, »homeDirectory« v »sys:dir:Description«, »uidNumber« v »sys:dir:UUID«, »telephoneNumber« v »sys:dir:Email« in »postalCode« kot »sys:dir:LastName«.

```
<Arguments>
   <Field key="sys:dir:Account" type="string">cn</Field>
   <Field key="sys:dir:Description" type="string">homeDirectory</Field>
   <Field key="sys:dir:UUID" type="string">uidNumber</Field>
   <Field key="sys:dir:Email" type="string">telephoneNumber</Field>
   <Field key="sys:dir:LastName" type="string">postalCode</Field>
</Arguments>
```

Primer 2: Konfiguracijo iz prvega primera spremenimo tako, da »sys:dir:Description« tipiziramo kot »BINARY«, »uidNumber« pa kot »INT« v šestnajstiški obliki, pretvorbo v desetiški sistem izvedemo tako, da v »typeExt« določimo RDX=16.

```
<Arguments>
   <Field key="sys:dir:Account" type="string">cn</Field>
   <Field key="sys:dir:Description" type="binary">homeDirectory</Field>
   <Field key="sys:dir:UUID" type="int" typeExt="RDX=16">uidNumber</Field>
   <Field key="sys:dir:Email" type="string">telephoneNumber</Field>
   <Field key="sys:dir:LastName" type="string">postalCode</Field>
</Arguments>
```

Rezultati pretvorbe za posameznega uporabnika so prikazani v naslednji tabeli.

Uporabnik	»sys:dir:Description«	»sys:dir:UUID«
test4	L2hvbWUvdGVzdDQ=	26214
test5	L2hvbWUvdGVzdDU=	406369
test6	L2hvbWUvdGVzdDY=	403332

Tabela 41: Rezultati pretvorbe (Primer 2)

Primer 3: Konfiguraciji iz prvega primera dodamo preslikavo za »sys:dir:DistinguishedName«, ki jo tipiziramo kot »DISTINGUISHED_NAME«.

```
<Arguments>
   <Field key="sys:dir:Account" type="string">cn</Field>
   <Field key="sys:dir:Description" type="string">homeDirectory</Field>
   <Field key="sys:dir:UUID" type="string">uidNumber</Field>
   <Field key="sys:dir:Email" type="string">telephoneNumber</Field>
   <Field key="sys:dir:LastName" type="string">postalCode</Field>
   <Field key="sys:dir:DistinguishedName" type="DISTINGUISHED_NAME"/>
</Arguments>
```

Rezultati pretvorbe za posameznega uporabnika so prikazani v naslednji tabeli.

Uporabnik	»sys:dir:DistinguishedName«
test4	uid=test4,ou=Userji,dc=my-domain,dc=com
test5	uid=test5,ou=Userji,dc=my-domain,dc=com
test6	uid=test6,ou=Userji,dc=my-domain,dc=com

Tabela 42: Rezultati pretvorbe (Primer 3)

Primer 4: Konfiguraciji iz prejšnega primera spremenimo preslikavo za »sys:dir:UUID« tako, da uporabimo »EXPRESSION«, ki ga sestavimo iz »sys:dir:DistinguishedName«, »sys:dir:Description« ter »uidNumber« (povezuje jih pomišljaj). Ker »uidNumber« ne moremo direktno naslavljati, določimo novo preslikavo z unikatni ključem (npr. »uid-stevilka«), tak ključ pa nato uporabimo v definiciji izraza.

```
<Arguments>
   <Field key=\"sys:dir:Account\" type=\"string\">cn</Field>
   <Field key=\"sys:dir:Description\" type=\"string\">homeDirectory</Field>
   <Field key=\"sys:dir:UUID\" type=\"expression\">%sys:dir:DistinguishedName%-%sys:dir:Description%-
%uid-stevilka%</Field>
   <Field key=\"sys:dir:Email\" type=\"string\">telephoneNumber</Field>
   <Field key=\"sys:dir:LastName\" type=\"string\">postalCode</Field>
   <Field key=\"sys:dir:DistinguishedName\" type=\"DISTINGUISHED_NAME\"/>
   <Field key=\"uid-stevilka\" type=\"string\">uidNumber</Field>
</Arguments>
```

Rezultati pretvorbe za posameznega uporabnika so prikazani v naslednji tabeli.

Uporabnik	»sys:dir:UUID«
test4	uid=test4,ou=Userji,dc=my-domain,dc=com-/home/test4-6666
test5	uid=test5,ou=Userji,dc=my-domain,dc=com-/home/test5-63361
test6	uid=test6,ou=Userji,dc=my-domain,dc=com-/home/test6-62784

Tabela 43: Rezultati pretvorbe (Primer 4)

Primer 5: Uporabnike, ki imajo »homeDirectory« na ldap strežniku nastavljene na »/home/test4« ali »/home/test5« bi radi onemogočili. Konfiguraciji iz prejšnjega primera dodamo preslikavo za »sys:dir:Flags:Enabled«, kjer navedemo željene vrednosti, končni rezultat pa negiramo.

```
<Arguments>
   <Field key="sys:dir:Account" type="string">cn</Field>
   <Field key="sys:dir:Description" type="string">homeDirectory</Field>
   <Field key="sys:dir:UUID" type="expression">%sys:dir:DistinguishedName%-%sys:dir:Description%-
%uid-stevilka%</Field>
   <Field key="sys:dir:Email" type="string">telephoneNumber</Field>
   <Field key="sys:dir:LastName" type="string">postalCode</Field>
   <Field key="sys:dir:DistinguishedName" type="DISTINGUISHED_NAME"/>
   <Field key="uid-stevilka" type="string">uidNumber</Field>
   <Field key="sys:dir:Flags:Enabled" type="boolean"
typeExt="VAL=/home/test4#/home/test5,NEG=true">homeDirectory</Field>
</Arguments>
```

Primer 6: Uporabnikom iz prejšnjega primera dodamo sinonime tako, da dodamo preslikavo za »sys:dir:Aliases«. Če želimo več sinonimov, potem preslikavo določimo kot »EXPRESSION«, kjer mu v »typeExt« določimo ločitveni znak (DEL). Tak izraz se bo najprej razdelil na podizraze glede na ločitveni znak, nato pa se bo vsak podizraz ovrednotil. Pogoj za uspešno ovrednotenje celotnega izraza je ta, da ima vsak podizraz enako število rezultatov.

```
<Arguments>
   <Field key="sys:dir:Account" type="string">cn</Field>
   <Field key="sys:dir:Description" type="string">homeDirectory</Field>
   <Field key="sys:dir:UUID" type="expression">%sys:dir:DistinguishedName%-%sys:dir:Description%-
%uid-stevilka%</Field>
   <Field key="sys:dir:Email" type="string">telephoneNumber</Field>
   <Field key="sys:dir:LastName" type="string">postalCode</Field>
   <Field key="sys:dir:DistinguishedName" type="DISTINGUISHED_NAME"/>
   <Field key="uid-stevilka" type="string">uidNumber</Field>
   <Field key="sys:dir:Flags:Enabled" type="boolean"
typeExt="VAL=/home/test4#/home/test5,NEG=true">homeDirectory</Field>
   <Field key="sys:dir:Aliases" type="expression"
typeExt="DEL=#">%sys:dir:DistinguishedName%#%sys:dir:Description%#%uid-stevilka%</Field>
</Arguments>
```

Primer 8: Prikazan je primer konfiguracije za sinhronizacijo uporabnikov in skupin z Microsoft Active Directory.

```
<Arguments>
   <Field key="sys:dir:Account" type="string">sAMAccountName</Field>
   <Field key="sys:dir:FirstName" type="string">givenName</Field>
   <Field key="sys:dir:LastName" type="string">sn</Field>
   <Field key="sys:dir:Description" type="string">description</Field>
   <Field key="sys:dir:Email" type="string">mail</Field>
   <Field key="sys:dir:UUID" type="binary">objectGUID</Field>
   <Field key="sys:dir:DistinguishedName" type="string">distinguishedName</Field>
   <Field key="sys:dir:Flags:Enabled" type="bitfield"
typeExt="OFF=1,NEG=true">userAccountControl</Field>
   <Field key="sys:dir:Flags:Locked" type="boolean" typeExt="VAL=0,NEG=true"
scope="user">lockoutTime</Field>
   <Field key="sys:dir:Aliases" type="string">userPrincipalName</Field>
   <Field key="sys:dir:GroupMembers" type="string" scope="group">member</Field>
</Arguments>
```

3.9 Varnostno kopiranje in obnovitev podatkov

Za učinkovito varovanje podatkov je potrebno zagotoviti redno izdelavo varnostnih kopij. Pomembno je, da varnostne kopije ne pridejo v roke nepooblaščenim osebam in da v primeru nesreče ostanejo nepoškodovane. Varnostno kopiranje zmanjšuje tveganje izgube podatkov zaradi tehnične okvare nosilca zapisa, napake v programu, naravne nesreče, nepooblaščenega dostopa, človeškega faktorja, ...
Omogoča obnovitev podatkov in povrnitev prejšnjega stanja.

Strežnik IMiS®/ARChive Server omogoča varnostno kopiranje in obnavljanje:

- dokumentarnega gradiva
- načrta razvrščanja gradiva v celoti ali samo izbranih razredov, zadev in dokumentov
- metapodatkov
- revizijske sledi
- digitalnih potrdil
- liste dostopnih pravic
- strežniškega imenika.

Spodnja tabela prikazuje tabele v bazi z opisom vsebine za varnostno kopiranje in obnavljanje.

Tabela	Opis
ACLENTRY	Vnosi za listo dostopnih pravic.
ACLENTRYVALIDITY	Časovne omejitve dostopnih pravic.
ATTRIBUTE	Atributi, ki so določeni na strežniku IMiS®/ARChive Server.
ATTRIBUTEVALUE	Povezovalna tabela med atributom, vrednostjo in entiteto, kateri vrednost pripada.
ATTRIBUTEVALUEBINARY	Tabela vsebuje binarne vrednosti atributov.
ATTRIBUTEVALUEDATETIME	Tabela vsebuje datumske vrednosti s časovno komponento.
ATTRIBUTEVALUEDOUBLE	Tabela vsebuje realna (racionalna) števila v plavajoči vejici z dvojno natančnostjo.
ATTRIBUTEVALUEFILE	Tabela vsebuje vrednosti o datotekah.
ATTRIBUTEVALUEINT32	Tabela vsebuje predznačena 32 bitna cela števila.
ATTRIBUTEVALUEINT64	Tabela vsebuje predznačena 64 bitna cela števila.
ATTRIBUTEVALUEINT128	Tabela vsebuje predznačena 128 bitna cela števila.
ATTRIBUTEVALUEINT128IDX	Tabela vsebuje indeksirana predznačena 128 bitna cela števila.
ATTRIBUTEVALUESTRING	Tabela vsebuje neomejene nize UTF-8 znakov (omejeni so z možnostjo platforme).
ATTRIBUTEVALUESTRING10	Tabela vsebuje nize UTF-8 znakov, dolgih 10 bytov.

Tabela	Opis
ATTRIBUTEVALUESTRING20	Tabela vsebuje nize UTF-8 znakov, dolgih 20 bytov.
ATTRIBUTEVALUESTRING30	Tabela vsebuje nize UTF-8 znakov, dolgih 30 bytov.
ATTRIBUTEVALUESTRING40	Tabela vsebuje nize UTF-8 znakov, dolgih 40 bytov.
ATTRIBUTEVALUESTRING50	Tabela vsebuje nize UTF-8 znakov, dolgih 50 bytov.
ATTRIBUTEVALUESTRING100	Tabela vsebuje nize UTF-8 znakov, dolgih 100 bytov.
ATTRIBUTEVALUESTRING200	Tabela vsebuje nize UTF-8 znakov, dolgih 200 bytov.
ATTRIBUTEVALUESTRING20IDX	Tabela vsebuje indeksirane nize UTF-8 znakov, dolgih 20 bytov.
ATTRIBUTEVALUESTRING30IDX	Tabela vsebuje indeksirane nize UTF-8 znakov, dolgih 30 bytov.
ATTRIBUTEVALUESTRING40IDX	Tabela vsebuje indeksirane nize UTF-8 znakov, dolgih 40 bytov.
ATTRIBUTEVALUESTRING50IDX	Tabela vsebuje indeksirane nize UTF-8 znakov, dolgih 50 bytov.
ATTRIBUTEVALUESTRING100IDX	Tabela vsebuje indeksirane nize UTF-8 znakov, dolgih 100 bytov.
ATTRIBUTEVALUESTRING200IDX	Tabela vsebuje indeksirane nize UTF-8 znakov, dolgih 200 bytov.
ATTRIBUTEVALUEUINT32	Tabela vsebuje ne-predznačena 32 bitna cela števila.
ATTRIBUTEVALUEUINT64	Tabela vsebuje ne-predznačena 64 bitna cela števila.
ATTRIBUTEVALUEUINT128	Tabela vsebuje ne-predznačena 128 bitna cela števila.
ATTRIBUTEVALUEUINT128IDX	Tabela vsebuje indeksirana ne-predznačena 128 bitna cela števila.
DIGITALCERTIFICATE	Tabela vsebuje digitalna potrdila, ki se nahajajo v strežniški shrambi digitalnih potrdil.
COUNTER	Tabela vsebuje vrednosti števcev za avtomatsko generiranje klasifikacijskih kod.
DIGITALSIGNATURE	Tabela vsebuje digitalne podpise.
DIRECTORYENTRY	Tabela vsebuje entitete (uporabnike ali uporabniške skupine) v strežniškem imeniku.
DIRECTORYENTRYALIAS	Tabela vsebuje sinonime entitet v strežniškem imeniku.
DIRECTORYENTRYGROUP	Tabela vsebuje podatke o članih uporabniških skupin.
ENTITY	Tabela vsebuje podatke o strežniških entitetah (razredih, zadevah, dokumentih).
ENTITYCLASS	Tabela vsebuje podatke o razredih na strežniku.
ENTITYDOCUMENT	Tabela vsebuje podatke o dokumentih na strežniku.
ENTITYFOLDER	Tabela vsebuje podatke o zadevah na strežniku.
PROFILE	Tabela vsebuje podatke o HSM profilih.
PROPERTY	Generična tabela nastavitev.
TEMPLATE	Tabela vsebuje strežniške predloge.
TEMPLATEATTRIBUTE	Tabela vsebuje povezave med strežniškimi predlogami in atributi, ki jim pripadajo.
TEMPLATEBIND	Tabela vsebuje podatke o povezavah med strežniškimi

Tabela	Opis
	predlogami.
ARCHIVALINFORMATIONPACKAGE	Tabela vsebuje arhivske informacijske pakete (AIP).
ARCHIVALINFORMATIONPACKAGEQUEUE	Tabela predstavlja čakalno vrsto za AIP.
CREATEAIPQUEUE	Tabela predstavlja čakalno vrsto za ustvarjanje AIP.
AIPJOBTIMESTAMP	Tabela vsebuje podatke o postopku časovnega žigosanja.
REVOCATIONDATA	Tabela vsebuje informacije o preklicih digitalnih potrdil.
CONTENTPROCESSINGQUEUE	Tabela vsebuje informacije za indeksiranje in pretvorbo vsebin.
CONTENTPROCESSINGERROR	Tabela vsebuje informacije vsebinah, nad katerimi se je neuspešno izvršilo indeksiranje ali pretvorba.
TIMESTAMP	Tabela vsebuje časovne žige.
VOLUME	Tabela vsebuje podatke o HSM volumnih.
LOOKUPTABLE	Tabela vsebuje statuse, pomembnost ter stopnje tajnosti entitet. Poleg tega vsebuje še prioritete dostave elektronske pošte ter status fizičnega gradiva.
HASH128	Tabela vsebuje prstne odtise dolžine 128 bitov.
HASH160	Tabela vsebuje prstne odtise dolžine 160 bitov.
HASH224	Tabela vsebuje prstne odtise dolžine 224 bitov.
HASH256	Tabela vsebuje prstne odtise dolžine 256 bitov.
HASH384	Tabela vsebuje prstne odtise dolžine 384 bitov.
HASH512	Tabela vsebuje prstne odtise dolžine 512 bitov.
REVOCATIONDATATIMESTAMP	Tabela vsebuje povezave med informacijami o preklicih digitalnih potrdil in časovnimi žigi, katerim pripadajo.
REVOCATIONDATADIGITALSIGNATURE	Tabela vsebuje povezave med informacijami o preklicih digitalnih potrdil in elektronskimi podpisi, katerim pripadajo.
ENTITYRETENTIONDISPOSITIONSCHEDULE	Tabela vsebuje informacije o povezavah med roki hrambe ter entitetami.
TIMESTAMPARCHIVALINFORMATIONPACKAGE	Tabela vsebuje arhivske informacijske pakete za časovne žige.
CERTIFICATEBODY	Tabela vsebuje binarne ali Base64 kodirane podatke, ki predstavljajo digitalno potrdilo.
FILEDESCRIPTION	Tabela vsebuje opise digitalnih vsebin.
ATTRIBUTEVALUENULL	Tabela vsebuje entitete z atributi, katerih vrednosti so »null«.
REVIEWQUEUE	Tabela vsebuje preglede odbiranja in izločanja, ki so v postopku izdelave ali v postopku izvajanja.
CONTENTBIND	Tabela vsebuje informacije o odvisnosti vsebin v entiteti.

Tabela 44: Tabele z opisom podatkov za varnostno kopiranje in obnavljanje

Strežnik IMiS®/ARChive Server dodatno omogoča samodejno izdelavo varnostnih kopij vseh, za delovanje in konsistentnost, pomembnih datotek. Te obsegajo podatkovno zbirko, kjer strežnik IMiS®/ARChive Server hrani metapodatke o gradivih, podatke revizijske sledi, konfiguracijo, ... in datoteke arhiviranih gradiv.

Kot izdelovalec programske opreme strankam priporočamo, da področje varnostnega kopiranja in obnovitve (angl. Backup/ Restore) arhivskega strežnika uredijo s profesionalnim orodjem tipa odjemalec/strežnik (npr. IBM Tivoli Storage Manager, HP Data Protector, CA ARCserve, ...).

3.9.1 Varnostno kopiranje

Za izvedbo varnostnega kopiranja moramo na IMiS®/ARChive Server namestiti odjemalca za varnostno shranjevanje datotek. Lokacija odjemalca je odvisna od tega, katero orodje tipa odjemalec/strežnik uporabimo.

3.9.1.1 Nastavitve varnostnega kopiranja

Nastavitve varnostnega kopiranja so odvisne od tega, katero orodje za varnostno kopiranje uporabimo. V večini primerov se opis nastavitev nahaja v njihovi dokumentaciji.
V osnovi pa lahko varnostno kopiranje ločimo na naslednje tipe:

- popolno varnostno kopiranje (angl. Full backup), ki obsega kopiranje vseh podatkov;
- prirastno varnostno kopiranje (angl. Incremental backup), ki obsega kopiranje podatkov, ki so se spremenili od časa zadnjega varnostnega kopiranja;
- diferencialna varnostna kopija (angl. Differential backup), ki obsega kopiranje podatkov, ki so se spremenili od časa zadnjega popolnega varnostnega kopiranja.

V kombinaciji z orodjem `dbtool` se za IMiS®/ARChive Server lahko uporabi katerikoli tip varnostnega kopiranja.

3.9.2 Obnovitev podatkov

Obnovitev arhiviranih dokumentov na strežniku IMiS®/ARChive Server je odvisna od stanja strežnika in želenega rezultata po obnovitvi podatkov strežnika.
V primeru negotovosti glede uporabnosti podatkov, ki so ostali na diskih, ali če obstaja možnost restavriranja stanja pred odpovedjo diskov, vam svetujemo, da se obrnete na našo tehnično podporo preko e-pošte: podpora@imis.si.

Pri obnovitvi podatkov iz varnostne kopije je zagotovljena celovitost podatkov, kar velja tudi za revizijsko sled.Obnovitev podatkov strežnika IMiS®/ARChive Server je odvisna od stanja strežnika in želenega rezultata po obnovitvi podatkov.

Za uspešno obnovitev podatkov mora strežnik opravilo varnostnega kopiranja shraniti najmanj za:

- bazo strežnika (privzeta lokacija je v imeniku/iarc/db);
- volumne strežnika z vso vsebino, rekurzivno (privzeta lokacija je v imeniku/iarc/vol);
- nastavitveno datoteko strežnika iarc.conf (privzeta lokacija je v imeniku /etc).

Popolno kopijo pri varnostnem kopiranju dosežemo, če opravilo varnostnega kopiranja poleg minimalnih zahtev shrani še:

- imenik /iarc/webadmin z vso vsebino, rekurzivno
- imenik /opt/IS/imisarc z vso vsebino, rekurzivno
- datoteko /etc/init.d/iarcd
- imenik /var/log/iarc z vso vsebino.

3.9.2.1 Nastavitve obnovitve podatkov

Pri postopku obnovitve podatkov najprej pridobimo zadnje shranjene tekstovne datoteke s podatki iz baze ter vsebino volumnov strežnika. Pri tem je potrebno paziti na pravilne nastavitve dostopnih pravic za uporabnika, v imenu katerega se strežnik zaganja (privzet uporabnik je iarc, za spreminjanje lastniških pravic pa lahko uporabimo orodja kot sta chown ali chmod).

Postopek obnovitev volumnov je naslednji:

- če vsebina starih volumnov obstaja (privzeta lokacija se nahaja v imeniku /iarc/vol) se izbriše;
- objekte volumnov iz varnostne kopije uporabnik kopira na volumne strežnika;
- za kopirane objekte je potrebno nastaviti dostopne pravice (pravice branja in pisanja).

Postopek obnovitve strežniške baze je naslednji:

- prazno strežniško bazo se uporabnik kopira na strežniško lokacijo (privzeta lokacija je v imeniku /iarc/db);
- na strežniški bazi se nastavi pravice branja in pisanja;
- tekstovne datoteke baze iz varnostne kopije se kopirajo na začasno lokacijo na disku;
- tekstovnim datotekam nastavimo pravice branja;
- strežniško bazo obnovimo z uporabo orodja dbtool . Pri tem lahko obnavljamo posamezne tabele z uporabo parametra –t ali pa uporabimo parameter –a, ki uvozi podatke v vse tabele, za katere tekstovne datoteke obstajajo; glej tudi poglavje 8.3 Konfiguriranje).

3.9.3 Primer

Pri izdelavi varnostnih kopij podatkov strežnika IMiS®/ARChive Server je potrebno upoštevati tudi možnost izgube ali delne okvare podatkov na sami varnostni kopiji in možnost uničenja medija. Dobra praksa vključuje izdelavo kopij varnostnih kopij in hranjenje dvojnikov na oddaljeni lokaciji. Varnostne kopije in njih kopije hranimo na ustrezno varovanem ognjevarnem mestu. Možnost popolne izgube podatkov zmanjšamo tudi z ustreznim načrtom izdelave in hranjenja varnostnih kopij za daljše obdobje.

Primer: Varnostne kopije strežnika IMiS®/ARChive Server izdelujemo dnevno.
Izvirne varnostne kopije hranimo v ognjevarni omari na lokaciji strežnika, kopije varnostnih kopij pa v sefu na oddaljeni lokaciji. Varnostne kopije, narejene ob koncu tedna, ob koncu meseca in ob koncu leta, hranimo v sefu še primerno dolgo, glede na trajanje medija in uporabnost varnostno kopiranih vsebin.

Uporabnost varnostnih kopij je potrebno tudi periodično preverjati. Podatke iz varnostne kopije restavriramo na drugo lokacijo in jih primerjamo z izvornimi podatki s strežnika IMiS®/ARChive Server. Načrt izdelave in hranjenja varnostnih kopij je potrebno periodično preverjati vsaj ob bistvenih spremembah, kot so zamenjave strežnikov, medijev ali infrastrukture, ki neposredno vpliva na dogodke povezane s strežnikom IMiS®/ARChive Server. Načrt praviloma preveri revizor informacijskih sistemov, ki se do njega tudi opredeli.

3.9.4 Težave pri obnovitvi podatkov

V primeru negotovosti glede uporabnosti podatkov, ki so ostali na diskih ali če obstaja možnost restavriranja stanja pred odpovedjo diskov vam svetujemo, da se obrnete na tehnično osebje proizvajalca strežnika IMiS®/ARChive Server.
Primeri v nadaljevanju prikazujejo težave, ki se lahko zgodijo pri postopku obnovitve podatkov.

Opis težave #1:

Pri postopku obnovitve podatkov pri uvozu z orodjem dbtool dobimo naslednjo napako (ime tekstovne datoteke je lahko drugačno in je odvisno od tega, katero tabelo v bazi obnavljamo):

Using 'sl_SI.UTF-8' locale settings.
Document Root: /iarc/db/
Port: 21553
attribute.txt file, open error 13: Permission denied.
Import command completed with 1 error(s).

Rešitev težave #1: Dostopne pravice za tekstovne datoteke niso pravilno nastavljene. Uporabniku, v imenu katerega se bo izvajal strežnik (privzet uporabnik je iarc) nima bralnih pravic na tekstovnih datotekah. Rešitev je nastavitev pravilnih dostopnih pravic tekstovnih datotek in ponovitev postopka obnavljanja podatkov.

Opis težave #2:

V postopku obnovitve podatkov pri uvozu z orodjem dbtool dobimo naslednjo napako:

Using 'sl_SI.UTF-8' locale settings.
Document Root: /iarc/db/
Port: 21553
Error; Unable to move to requested record (dberr#-940) while going backwards! (file = RaimaKeyIterator.cpp, func = moveTo, line = 200)

Rešitev težave #2: Dostopne pravice za strežniško bazo niso pravilno nastavljene. Rešitev je nastavitev dostopnih pravic branja in pisanja na strežniški bazi za uporabnika, v imenu katerega se bo IMiS®/ARChive Server zaganjal. Sledi postopek obnavljanja podatkov.

Opis težave #3:

V postopku obnovitve podatkov pri uvozu z orodjem dbtool dobimo naslednjo napako (ime tekstovne datoteke in vrednost tabele je lahko drugačno in je odvisno od tega, katero tabelo v bazi obnavljamo):

Using 'sl_SI.UTF-8' locale settings.
Document Root: /iarc/db/
Port: 21553
Importing attribute.txt (3194 bytes) to ATTRIBUTE ... Error; Tried to insert duplicate key into table 10008! (file = RaimaDataSet.cpp, func = InsertObject, line = 207)

Rešitev težave #3: Strežniška baza v katero želimo uvoziti že vsebuje podatke, ki so podvojeni s podatki v tekstovnih datotekah. Rešitev za takšen tip napak je več:

- strežniško bazo zamenjamo s prazno ter ponovimo postopek uvoza podatkov;
- posamezno tabelo, pri kateri dobimo v postopku uvoza podatkov zgoraj opisano napako najprej izbrišemo (uporabimo `init` ukaz v `dbtool-u`), ter ponovimo postopek uvoza podatkov za to tabelo (to ne velja za revizijsko sled, saj se le-te ne da izbrisati);
- pri uvozu podatkov z orodjem `dbtool` uporabimo možnost `–o`, ki morebitne obstoječe podatke v bazi prepiše s podatki iz tekstovnih datotek.

Primer: ukaz za uvoz vseh tekstovnih datotek z možnostjo prepisa:
su - iarc -s /bin/bash -c "cd /opt/IS/imisarc && ./dbtool -f /etc/iarc.conf -w /iarc/db/ -o -a imp"

3.10 Skladiščenje vsebine

Vsebina entitet se shranjuje na HSM (angl. Hierarchical Storage Management) profilih in volumnih. HSM je hierarhično zasnovan sistem za shranjevanje podatkov, kjer se vsebina, do katere se pogosto dosopa, shranjuje na hitrjših nosilcih, vsebina z malo dostopi pa se hrani na počasnejših nosilcih (za več informacij glej: https://en.wikipedia.org/wiki/Hierarchical_storage_management). Profili predstavljajo logično zaključeno HSM celoto, kjer so posamezni volumni razporejeni po nivojih glede na hitrost (od najhitrjših do najpočasnejših). Volumni predstavljajo nosilce podatkov, s katerimi IMiS®/ARChive Server operira.

3.10.1 Profili

Profili predstavljajo logično zaključeno HSM celoto. Poleg osnovnih informacij (ime, opis), profil vsebuje še skupno velikost vseh volumnov, ki pripadajo profilu, skupen zaseden prostor, skupno število vsebin na vseh volumnih ter naslednje lastnosti, ki pogojujejo delovanje profila:

- »ReadOnly« – vsebin na volumnih, ki pripadajo profilu se ne more urejati;
- »WORM« (angl. write once read many) – vsebine se na volumnih, ki pripadajo profilu lahko ustvarja, urejati se jih ne more;
- »StopAddingObjects« - vsebine na volumnih se lahko bere in ureja, novih vsebin se ne more dodajati.

Profilu lahko določimo razred in s tem omejimo vsebine, ki se bodo shranjevale na volumne profila. Če ima profil določen razred, potem se bodo na volumne shranjevale samo vsebine entitet, ki so vsebovane entitete nastavljenega razreda. Če razred ni določen, profil velja za celoten arhiv. Zaporedje volumnov na profilu (znotraj istega nivoja) se lahko spreminja.
S tem lahko administrator nadzira zasedenost volumnov na profilu.

Primer konfiguracije profila:

```
<ns1:Profile xsi:type="ns1:StorageProfile">
   <ns1:Name xsi:type="xsd:string">Dokumenti</ns1:Name>
   <ns1:Objects xsi:type="xsd:unsignedInt">1095174</ns1:Objects>
   <ns1:Size xsi:type="xsd:unsignedLong">51539607552</ns1:Size>
   <ns1:Used xsi:type="xsd:unsignedLong">44696233984</ns1:Used>
   <ns1:ReadOnly xsi:type="xsd:boolean">false</ns1:ReadOnly>
   <ns1:WORM xsi:type="xsd:boolean">false</ns1:WORM>
   <ns1:StopAddingObjects xsi:type="xsd:boolean">false</ns1:StopAddingObjects>
   <ns1:Volume level="0">vol20</ns1:Volume>
   <ns1:Volume level="0">vol06</ns1:Volume>
   <ns1:Volume level="0">vol05</ns1:Volume>
   <ns1:Volume level="0">vol00</ns1:Volume>
   <ns1:Volume level="0">vol01</ns1:Volume>
   <ns1:Volume level="0">vol02</ns1:Volume>
   <ns1:Class xsi:type="ns1:EntityId" type="ClassificationCode"/>
</ns1:Profile>
```

Etiketa »Name«: Vsebina etiketa vsebuje ime profila.

Etiketa »Objects«: Vsebina etikete vsebuje število vsebin na vseh volumnih na profilu.

Etiketa »Size«: Vsebina etikete vsebuje skupno velikost vseh volumnov na profilu (v bajtih).

Etiketa »Used«: Vsebina etikete vsebuje skupno velikost zasedenega prostora na vseh volumnih na profilu (v bajtih).

Etiketa »ReadOnly«: Vrednost »True« predstavlja profil, ki omogoča samo branje vsebine. Vrednost »False« omogoča tudi ustvarjanje in urejanje vsebine (odvisno je od drugih lastnosti profila).

Etiketa »WORM«: Vrednost »True« predstavlja profil, ki omogoča samo ustvarjanje in branje vsebin. Vrednost »False« omogoča tudi urejanje vsebine (odvisno je od drugih lastnosti profila).

Etiketa »StopAddingObjects«: Vrednost »True« onemogoči ustvarjanje novih vsebin na volumnih profila. Vrednost »False« omogoča ustvarjanje novih vsebin (odvisno je od drugih lastnosti profila).

Etikete »Volume«: Vrednost etikete predstavlja ime volumna, ki je dodeljen profilu. Atribut »level« predstavlja HSM nivo volumna, zaporedje etiket pa predstavlja zaporedje volumnov na profilu.

Etiketa »Class«: Vrednost etikete predstavlja identifikator razreda, kateremu je profil dodeljen. Atribut »type« predstavlja tip identifikatorja.

3.10.2 Volumni

Volumni predstavljajo nosilce podatkov, s katerimi strežnik operira. Vsak volumen vsebuje definicijo, to je absolutna pot do datotečnega sistema, kamor se bo vsebina fizično shranila. Poleg tega volumni vsebujejo še informacije o njihovi velikosti, profilu, številu vsebin na volumnu, zaseden prostor, »ReadOnly«, »WORM« in »StopAddingObjects« nastavitve (njihov povem je enak kot na profilu), ter informacijo, ali je volumen vpet (angl. mounted) ali ne. Če volumen ni vpet, potem so zahteve za dostop do vsebine na takem volumnu zavrnjene. Volumen se lahko menja med profili, vendar samo če je prazen. Naslednji zapis prikazuje primer konfiguracije volumna.


```
<ns1:Volume xsi:type="ns1:StorageVolume">
   <ns1:Name xsi:type="xsd:string">vol05</ns1:Name>
   <ns1:Definition xsi:type="xsd:string">/iarc/vol/vol05</ns1:Definition>
   <ns1:Profile xsi:type="ns1:StorageVolumeProfile">
      <ns1:Name xsi:type="xsd:string">Dokumenti</ns1:Name>
      <ns1:Level xsi:type="xsd:unsignedByte">0</ns1:Level>
      <ns1:Sequence xsi:type="xsd:unsignedInt">2</ns1:Sequence>
   </ns1:Profile>
   <ns1:Objects xsi:type="xsd:unsignedInt">37212</ns1:Objects>
   <ns1:Size xsi:type="xsd:unsignedLong">8589934592</ns1:Size>
   <ns1:Used xsi:type="xsd:unsignedLong">8411480064</ns1:Used>
   <ns1:ReadOnly xsi:type="xsd:boolean">false</ns1:ReadOnly>
   <ns1:WORM xsi:type="xsd:boolean">false</ns1:WORM>
   <ns1:StopAddingObjects xsi:type="xsd:boolean">false</ns1:StopAddingObjects>
   <ns1:Mounted xsi:type="xsd:boolean">true</ns1:Mounted>
</ns1:Volume>
```

Etiketa »Name«: Vrednost etikete predstavlja ime volumna.

Etiketa »Definition«: Vrednost etikete predstavlja absolutno pot na datotečne sistemu, kamor se bo vsebina fizično shranjevala.

Etiketa »Profile«: Etiketa vsebuje ime profila, HSM nivo ter zaporedje v profilu, kjer se volumen nahaja.

Etiketa »Objects«: Vrednost predstavlja število vsebin na volumnu.

Etiketa »Size«: Vrednost predstavlja velikost volumna (v bajtih).

Etiketa »Used«: Vrednost predstavlja zaseden prostor na volumnu (v bajtih).

Etiketa »ReadOnly«: Vrednost »True« predstavlja volumen »samo za branje«. Vrednost »False« pomeni, da administrator lahko ustvarja in ureja vsebine na volumnu (odvisno od drugih lastnosti volumna).

Etiketa »WORM«: Vrednost »True« predstavlja volumen, ki omogoča samo ustvarjanje in branje vsebin. Vrednost »false« omogoča tudi urejanje vsebine (odvisno od drugih lastnosti volumna).

Etiketa »StopAddingObjects«: Vrednost »True« onemogoča ustvarjanje novih vsebin na volumnu. Vrednost »False« omogoča ustvarjanje novih vsebin (odvisno od drugih lastnosti volumna).

Etiketa »Mounted«: Vrednost »True« pomeni, da je volumen vpet in da omogoča serviranje shranjene vsebine. Vrednost »False« pomeni, da dostop do vsebine ni možen. Prav tako ni možno ustvarjanje nove vsebine.

4 SISTEMSKE ZAHTEVE

4.1 Strojna oprema

Strežniki, ki jih lahko danes kupimo na tržišču, večinoma zadoščajo zahtevam strežnika IMiS®/ARChive Server saj potrebuje malo virov in zato brez težav deluje tudi v virtualnih okoljih. Pozornost je potrebno posvetiti ustrezni arhitekturi strežnika in jo uskladiti z eno izmed podprtih arhitektur produkta, večinoma gre za Intel x86 platforme v 32 in 64 bitnih izvedbah.

4.1.1 Načrtovanje procesorske moči strežnika

Pri izbiri procesorske moči je potrebno posvetiti pozornost predvidenim obremenitvam strežnika (število odjemalcev, število vzporednih uporabniških sej, povprečne velikosti arhiviranih vsebin, uporaba revizijske sledi, ...).

Glede na trenutne funkcionalne lastnosti produkta, je na trgu zadovoljiva količina različnih procesorjev srednjih ali višjih zmogljivosti, ki omogoča kvalitetno obratovalno okolje.
Navadno se pri izbiri procesorske moči lahko orientiramo glede na priporočene zahteve samega operacijskega sistema.
Za namišljen sistem 500 uporabnikov s povprečno 200 vpogledi na dan pri povprečni velikosti arhivirane vsebine 100KB, bi tudi v primeru sočasne uporabe vseh uporabnikov s transakcijami v istem časovnem obdobju zadostoval 1x procesor tehnologij Xeon QuadCore srednje frekvenčne zmogljivosti ali 1x procesor družine Intel Core i5/i7 srednje frekvenčne zmogljivosti.

4.1.2 Načrtovanje pomnilniških kapacitet strežnika

Pri načrtovanju velikosti pomnilnika strežnika IMiS®/ARChive Server je potrebno upoštevati:

- zahteve operacijskega sistema;
- osnovne zahteve strežnika, ki za samo delovanje potrebuje približno 512 MB;
- število hkratnih uporabnikov; vsaka povezava bo potrebovala približno 256 KB;
- minimalna priporočena velikost pomnilnika je enaka seštevku 512MB in minimalne velikosti pomnilnika, ki je zahtevan s strani proizvajalca operacijskega sistema.

Priporočena velikost pomnilnika je enaka seštevku potrebe po pomnilniku s strani servisov samega operacijskega sistema in 1024MB (1GB) za delovanje samega strežnika IMiS®/ARChive Server.

4.1.3 Načrtovanje diskovnih kapacitet strežnika

Pri načrtovanju diskovnih kapacitet strežnika je potrebno upoštevati naslednje:

- zahteve operacijskega sistema;
- predviden povprečen dnevni prirastek objektov;
- predviden prirast objektov na račun pretvorbe starega papirnega arhiva v elektronsko;
- velikost povprečnega objekta;
- predvideni čas uporabe strežnika (npr. 5 let).

Objekti, ki jih hrani strežnik IMiS®/ARChive Server na svojih volumnih so lahko različnih tipov in lahko izvirajo iz različnih računalniških okolij.
Objekti, skenirani z IMiS®/Scan pri ločljivosti 300pik/inčo, v črno-beli tehniki (barvna globina 1 bit) pri uporabi privzete metode za zgoščevanje (CCITT G4 T6) zavzamejo povprečno 45KB na skenirano stran.

Z uporabo drugih zgoščevalnih metod, barvne globine in ločljivosti se velikost praviloma poveča (glej tabelo v nadaljevanju).

Barvna globina	Črno/belo (1 bit)	Svinsko (8 bit)	Barvno (24 bit)
Brez	605 KB	5 MB	15 MB
CCITT G3	85 KB	x	X
CCITT G4 T6	45 KB	x	X
JBIG	36 KB	x	x
JBIG 2bit	x	84 KB	x
JBIG 3bit	x	165 KB	x
JBIG 4bit	x	420 KB	x
Packed bits	109 KB	5 MB	15 MB
LZW	75 KB	3,2 MB	x
Packed bits 8 bit	x	x	x
Packed bits 24 bit	x	x	x
ZIP	56 KB	3 MB	9 MB
Wang JPEG	x	315 KB	363 KB
Sekvenčni JPEG	x	315 KB	360 KB
Progresivni JPEG	x	310 KB	334 KB

Tabela 45: Povprečne velikosti skeniranega dokumenta glede na metode stiskanja

Pri izbiri ustrezne metode za zgoščevanje je potrebno upoštevati, da prenos večjih objektov preko računalniške mreže, zahteva večjo pasovno širino in lahko vpliva tudi na odzivnost računalniške mreže.
Odsvetujemo skeniranje v sivinah ali barvah, saj danes večina skenerjev,
ki so namenjeni zajemu dokumentov uporablja napredne metode in filtre za grafično obdelavo, kar zagotavlja optimalno kvaliteto skeniranih dokumentov.

Priporočamo uporabo diskovja z ustreznim varovanjem podatkov in razširljivostjo.
Priporočamo uporabo sodobnih diskovnih krmilnikov, ki omogočajo predpomnjenje branja in pisanja. Predpomnilnik naj ima avtonomno podporo napajanja ali pa izveden s tehnologijo »Flash Memory« (EEPROM tehnologija), ki lahko hrani podatke brez pomoči električne energije, kot to zahteva statičen RAM, s katerim so opremljeni starejši RAID krmilniki.
Diski naj bodo združeni v redundantno diskovno polje. Zaradi izkoristka priporočamo diskovno polje tipa RAID5 z dodatnim rezervnim diskom.

Odsvetujemo diskovje, ki bi bilo strežniku IMiS®/ARChive Server dostopno preko lokalne mreže, npr: NAS (network attached storage) ali diskovje, ki se nahaja na drugem strežniku in je strežniku IMiS®/ARChive Server dano v souporabo preko protokolov kot so CIFS, NSF, ...

4.1.4 Komunikacijske poti

IMiS® odjemalci komunicirajo s strežnikom IMiS®/ARChive Server preko omrežnih vrat številka 16807, v kolikor v nastavitveni datoteki /etc/iarc.conf ni nastavljeno drugače.
Potrebno je poskrbeti, da je komunikacija preko teh vrat omogočena, pri čemer naj pravila na požarnih zidovih ali drugi aktivni omrežni opremi dovoljujejo vzpostavitev povezave odjemalec-strežnik s strani IMiS® odjemalca, medtem ko vzpostavitev povezave strežnik-odjemalec s strani strežnika ni predvidena/potrebna.

4.1.5 Priklop na mrežno opremo

Priporočamo podvojene povezave in priklop na hrbtenico lokalne mreže s čim manj posredniki. Najbolje je, če je strežnik priklopljen na glavno mrežno stikalo.
Omrežni protokol med odjemalcem IMiS® in strežnikom IMiS®/ARChive Server je optimiziran za pakete v velikosti 32KB v primeru podatkovnih paketov (branje/pisanje arhivirane vsebine) in manjše v primeru ukaznih paketov.
Posamezen omrežni paket lahko izjemoma preseže 32KB v kolikor zahtevek ali odgovor na zahtevek to zahteva (večji obseg podatkov enega zahtevka).

Vsa komunikacija med strežnikom in odjemalcem je zgoščena po GZIP in je iz tega vidika propustnost največja. Testi med razvojem in pri opazovanju velikih produkcijskih sistemov kažejo na ozko grlo prav v širini komunikacijskih poti in strojne opreme.

4.1.6 Administratorske pravice

Pravice, ki jih administrator strežnika IMiS®/ARChive Server pri svojem delu potrebuje, so ekvivalentne korenskemu uporabniku root. Pravice navadno dodeli administrator strežnika in morajo zadoščati za namestitev, nadgradnjo in administriranje strežnika.
Za svoje delovanje ne potrebuje priviligiranih pravic.
Namestitvena skripta strežnika IMiS®/ARChive Server pri namestitvi ustvari uporabniški račun iarc in skupino iarc s katerim so zagnani vsi procesi strežnika.
V primeru napada preko morebitne napake v aplikacijski kodi strežnika je s tem onemogočen kakršen koli prevzem pravic korenskega uporabnika.

4.1.7 Nadzor delovanja strojne opreme

Večina proizvajalcev strojne strežniške opreme ob nakupu strežnika priloži tudi sistem za nadzor strojne opreme (npr. IBM Tivoli, HP Insight Systems Manager, Dell OpenManage). Uporaba takega sistema je zelo dobrodošla, ko pride do težav pri delovanju opreme ali v primeru, ko nadzorniki sistema potrebujejo podatke o delovanju strežnikov. Dobrodošlo je tudi, če nadzorni sistem omogoča sporočanje napak pri delovanju sistemov preko mobilnih telefonov ali elektronske pošte.

4.1.8 Minimalne zahteve

- strežnik z Intel Pentium x86 ali x86_64 procesorjem 800Mhz ali drugim kompatibilnim procesorjem z x86 arhitekturo (glej minimalne zahteve namestitvene platforme – operacijskega sistema);
- 1GB RAM (glej minimalne zahteve namestitvene platforme – operacijskega sistema in prištej 512 MB);
- ustrezna diskovna kapaciteta za pričakovan obseg arhiviranih vsebin, minimalno 1GB za delovanje strežnika;
- dostop do omrežja po TCP/IP protokolu (IPv4 ali IPv6);
- katera koli strojna oprema, ki nudi podporo izvajanju Linux operacijskega sistema z navedenimi podprtimi distribucijami v omrežnem načinu.

4.1.9 Priporočene zahteve

- strežnik z več-jedrnim Intel Xeon E5/E7 ali Xeon 5xxx/6xxx/7xxx (x86_64) procesorjem 2GHz (ali zmogljivejšim);
- 4GB SDRAM (DDR3/DDR4) visokih frekvenc ali več;
- hitra matična plošča z Front Side Bus-om višjih frekvenc (1GHz ali hitrejše);
- volumni na RAID5 logičnih diskih/particijah (vnaprej predvideni 3-5 letni prirast prostora na diskih);
- SCSI/SAS kontrolerji z write-back cache zmožnostmi (do 40% boljša učinkovitost), priporočljivo je 128MB cache ali več s podporo baterijskemu napajanju ali flash spominu v primeru prekinitve napajanja *;
- hitri SCSI/SAS diski (10k/15k RPM) z ustreznim predpomnjenjem *;
- redundantno napajanje z vzpostavljenim sistemom hlajenja;
- redundantni omrežni priklop 1Gbps ali več z IPv4 ali IPv6 protokolom.

** diskovni podsistem lahko zamenjamo z ustreznimi omrežnimi SAN volumni, ki so performančno primerljivi priporočenim lokalnim diskovnim kapacitetam.*
** Produkt normalno deluje tudi v svetovno priznanih virtualizacijskih okoljih kot so VMware ESX/ESXi, Microsoft Hyper-V, Oracle VM ipd. v kolikor so mu zagotovljeni ustrezni virtualni resursi, ki omogočajo podobno performančno okolje, kot ga dosegamo z zgornjo priporočeno strojno opremo.*

4.2 Programska oprema

4.2.1 Operacijski sistemi

Strežnik IMiS®/ARChive Server deluje na operacijskem sistemu x86/x86_64, na distribucijah Red Hat in SuSE z naslednjimi derivati:

- RHEL 4.x
- RHEL 5.x
- RHEL 6.x
- CentOS 4.x
- CentOS 5.x
- CentOS 6.x
- SLES 11.x
- SLES 12.x
- OpenSuSE 11.x
- OpenSuSE 12.x.

4.2.1.1 Priporočene zahteve

Linux OS (RedHat EL/Fedora, SuSE SLES/OpenSuSE (vsi zasnovani na kateremkoli 2.6.x jedru). Pri namestitvi in delovanju strežnika IMiS®/ARChive Server mora operacijski sistem zagotoviti spodaj navedena orodja in knjižnice. Orodja in knjižnice operacijskega sistema Linux so lahko sestavni del različnih namestitvenih paketov operacijskega sistema.

4.2.2 Seznam obveznih sistemskih orodij

bash (več na: http://www.linuxmanpages.com/man1/bash.1.php)
chmod (več na: http://www.linuxmanpages.com/man1/chmod.1.php)
chown (več na: http://www.linuxmanpages.com/man1/chown.1.php)
cp (več na: http://www.linuxmanpages.com/man1/cp.1.php)
echo (več na: http://www.linuxmanpages.com/man1/echo.1.php)
grep (več na: http://www.linuxmanpages.com/man1/grep.1.php)
mv (več na: http://www.linuxmanpages.com/man1/mv.1.php)
ps (več na: http://www.linuxmanpages.com/man1/ps.1.php)
pwd (več na: http://www.linuxmanpages.com/man1/pwd.1.php)
rm (več na: http://www.linuxmanpages.com/man1/rm.1.php)
rmdir (več na: http://www.linuxmanpages.com/man1/rmdir.1.php)
rpm (več na: http://www.linuxmanpages.com/man8/rpm.8.php)
sed (več na: http://www.linuxmanpages.com/man1/sed.1.php)
sh (več na: http://www.linuxmanpages.com/man1/sh.1.php)
su (več na: http://www.linuxmanpages.com/man1/su.1.php)
touch (več na: http://www.linuxmanpages.com/man1/touch.1.php)
ip (več na: http://www.linuxmanpages.com/man7/ip.7.php)
ldconfig (več na: http://www.linuxmanpages.com/man1/ps.1.php)
awk (več na: http://www.linuxmanpages.com/man1/awk.1.php)
find (več na: http://www.linuxmanpages.com/man1/find.1.php)
id (več na: http://www.linuxmanpages.com/man1/id.1.php)
ipcrm (več na: http://www.linuxmanpages.com/man8/ipcrm.8.php)
ipcs (več na: http://www.linuxmanpages.com/man8/ipcs.8.php)
killall (več na: http://www.linuxmanpages.com/man1/killall.1.php)
setsid (več na: http://www.linuxmanpages.com/man2/setsid.2.php)
groupadd (več na: http://www.linuxmanpages.com/man8/groupadd.8.php)
useradd (več na: http://www.linuxmanpages.com/man8/useradd.8.php)

4.2.3 Seznam obveznih sistemskih knjižnic

libc.so.6
libm.so.6
libpthread.so.0
libstdc++.so.6
libdl.so.2
libgcc_s.so.1
librt.so.1
libbz2.so.1
libz.so.1
rpmlib

4.2.4 Minimalne zahteve

Operacijski sistem: Linux OS (RedHat EL/Fedora, SuSE SLES/OpenSuSE (vsi zasnovani na kateremkoli 2.6.x jedru).

5 NAMESTITEV

V nadaljevanju je opisan postopek namestitve s pomočjo konzolnih orodij.
Postopek namestitve strežnika IMiS®/ARChive Server lahko opravi korenski uporabnik (root) ali uporabnik z ekvivalentnimi pravicami (sudo). Poteka po korakih in je enoten za vse ciljne skupine oseb, ki strežnik nameščajo.

5.1 Postopek namestitve

Namestitev je mogoče opraviti le v okolju, ki izpolnjuje vsaj minimalne zahteve za namestitev ene od podprtih Linux distribucij.
Minimalne zahteve nadgradimo v skladu s predvidenimi potrebami (glej poglavje 4.1 Strojna oprema in poglavje 4.2 Programska oprema).
Postopek namestitve strežnika IMiS®/ARChive Server je preprost in v nadaljevanju opisan po korakih.

Korak 1

V konzolo operacijskega sistema se prijavimo kot root uporabnik ali ukaze izvajamo z ekvivalentom root uporabnika preko orodja sudo. Diskovni pogoni, ki so predvideni za uporabo strežnika IMiS®/ARChive Server, naj bodo predhodno pripravljeni in dostopni v datotečnem sistemu. Lokacija /iarc je privzeta za večino datotek strežnika (podatkovna baza, datoteke dokumentov, ...). Namestitev opravimo z orodjem rpm, ki je sestavni del podprtih Linux distribucij.

Korak 2

Izvedemo ukaz rpm za namestitev namestitvenega paketa:

[user1@iarc ~1# sudo rpm -ivh imisarc.9.1.1406-600.0001.el4.i386.rpm

Ime paketa za namestitev je lahko tudi drugačno. Odvisno je od Linux distribucije in verzije strežnika IMiS®/ARChive Server.

Korak 3

Ob uspešni namestitvi se izpiše naslednje (izpis se lahko spreminja glede na uporabljeno Linux distribucijo):

Preparing ## [100%]
1:imisarc ## [100%]
Performing POSTINSTALLATION Actions
POSTINSTALLATION Actions Done

Korak 4

Postopek namestitve ustvari naslednje imenike in datoteke:

/iarc/db in	
/iarc/db/iarc	Imenika, ki vsebujeta datoteke podatkovne baze strežnika.
/iarc/fti	Imenik, ki vsebuje podatkovno bazo sistema za iskanje po polnem besedilu dokumentov.
/iarc/vol	Imenik, ki vsebuje volumne, kamor strežnik IMiS®/ARChive Server shranjuje datoteke dokumentov.
/iarc/wcache	Imenik, ki ga strežnik uporablja za predpomnenje datotek dokumentov pri nastajanju ali urejanju.
/iarc/rcache	Imenik za predpomnenje datotek dokumentov, za katere uporabniki podajo zahtevo za pregledovanje.
/iarc/work	Imenik, v katerem strežnik shranjuje datoteke začasne narave pri posameznih procesih.
/opt/IS/imisarc	Vsebuje izvajalne datoteke in knjižnice, ki jih strežnik potrebuje za svoje delovanje.
/etc/iarc.conf	Konfiguracijska datoteka strežnika.
/etc/init.d/iarcd	Zagonska skripta strežnika.

5.2 Ponamestitveni postopki

Ponamestitvene postopke strežnika IMiS®/ARChive Server lahko opravi korenski uporabnik ali uporabnik z ekvivalentnimi pravicami (administrator, šolano osebje, ...).

5.2.1 Nastavitev števila hkrati odprtih datotek

Vsak proces na Linux operacijskem sistemu potrebuje za delovanje določene pravice.
To dosežemo tako, da proces teče s privilegiji določenega uporabniškega računa, kateremu dodelimo ustrezne pravice. Poleg namestitve datotek postopek namestitve ustvari tudi uporabniški račun z imenom iarc.
Po namestitvi je potrebno ročno nastaviti še največje število hkrati odprtih datotek za uporabniški račun strežnika IMiS®/ARChive Server. Priporočena vrednost je največ 4096 hkrati odprtih datotek.

Nastavitev uredimo v datoteki /etc/security/limits.conf tako, da vpišemo naslednji dve vrstici:

iarc soft nofile 4096
iarc hard nofile 4096

Pravice lahko nastavimo tudi za vse uporabnike, ki pripadajo skupini iarc , čeprav to eksplicitno ni potrebno:

@iarc soft nofile 4096
@iarc hard nofile 4096

Enak učinek je možno doseči tudi z zaustvaritvijo datoteke /etc/security/limits.d/iarc.conf, v katero vnesemo zgornji dve vrstici, za uporabnika ali skupino.
Odločitev za en ali drug pristop naj temelji na:

- notranjih pravilih izvajanja sistemske administracije strežnika, kamor se strežnik IMiS®/ARChive Server namešča;
- osebni preferenci glavnega sistemskega administratorja – skrbnika.

pam_limits modul PAM arhitekture (Pluggable Authentication Modules for Linux, http://en.wikipedia.org/wiki/Linux_PAM) namreč upošteva vse nastavitvene ukaze iz direktorija /etc/security/limits.d/, kot tudi samo nastavitveno datoteko /etc/security/limits.conf.

5.2.2 Nastavitev samodejnega zagona

Ob namestitvi je strežnik IMiS®/ARChive Server nastavljen na samodejni zagon.
Samodejni zagon strežnika ob zagonu operacijskega sistema lahko nastavimo tudi ročno:

- za RHEL in CentOS distribucije z ukazom: **chkconfig iarcd on;**
- za SLES in OpenSuSE distribucije z ukazom: **chkconfig iarcd on** ali **yast** (yast: standardno nastavitveno orodje v SLES in OpenSuSE distribucijah).

Nastavitev samodejnega zagona strežniške storitve lahko preverimo:

- za RHEL in CentOS distribucije z ukazom: **chkconfig iarcd –list**;
  (izpiše na katerih nivojih se servis samodejno zažene).
- za SLES in OpenSuSE distribucije z ukazom: **chkconfig iarcd –list** ali **yast**.

Pomembno je, da se servis zažene na nivoju 3 in 5 kar potrjuje naslednji izpis:

```
[user1@iarc ~]# sudo chkconfig iarcd –list
iarcd          0:off  1:off  2:on   3:on   4:on   5:on   6:off
```

5.3 Preizkušanje namestitve in nastavitev

Preizkus uspešnosti namestitve lahko opravimo v nekaj korakih:

Korak 1

V nastavitveni datoteki /etc/iarc.conf, v sekciji [Log], nastavimo parameter LogLevel na vrednost 7 in zaženemo IMiS®/ARChive Server strežniško storitev.

Izpiše se sporočilo:

```
[user1@iarc ~]# sudo service iarcd start
Starting IMiS/ARChive HSM Storage Server:                [  OK  ]
```

Korak 2

Z ukazom pstree –G preverimo stanje delujočih procesov in njihovih niti.

Izpiše se seznam, katerega del je tudi (izpis se lahko spreminja glede na distribucijo):

```
[user1@iarc ~]# sudo pstree –G
init─┬─ ...
...
   ├─iarcd─┬─iarcd───7*[{iarcd}]
   │       └─iavol───{iavol}
```

Tak izpis se pojavi v primeru, da ima v iarc.conf, v sekciji [Server], parameter ConnChilds vrednost 1, ReqThreads pa 7. Izpis prikazuje glavni proces (iarcd),
ki upravlja s povezovalnim procesom (drugi iarcd), ta ima 7 niti in proces iavol,
ki ima eno nit, ki upravlja z volumni strežnika IMiS®/ARChive Server.

Korak 3

Z ukazom netstat –tan preverimo ali povezovalni in administrativni proces strežnika IMiS®/ARChive Server na TCP vratih, določenih v iarc.conf pričakujeta zahtevke.

V primeru, da so nastavljena privzeta TCP vrata, izpis vsebuje:

```
[user1@iarc ~]# netstat -tan
Proto Recv-Q Send-Q  Local Address  Foreign Address     State
...
tcp       0      0  *:16807         *:*             LISTEN
tcp       0      0  *:16808         *:*             LISTEN
```

6 NADGRADNJA

Nadgradnje strežnika IMiS®/ARChive Server lahko opravi korenski uporabnik ali uporabnik z ekvivalentnimi pravicami (administrator, šolano osebje, ...).

Ob nadgradnji je potrebno izvesti nekatere izmed naslednjih postopkov (lahko tudi vse, če je preskok verzije strežnika večji, npr. ob nadgradnji na višjo glavno verzijo):

- preverjanje konsistentnosti interne baze podatkov strežnika;
- izvoz interne baze podatkov strežnika;
- nadgradnja izvajalnih programov;
- nadgradnja knjižnic (opcijsko);
- razširitev sheme baze podatkov strežnika;
- uvoz baze podatkov strežnika potem, ko je bila shema razširjena;
- ureditev pravic in lastništva na datotekah strežnika.

6.1 Postopek nadgradnje

Korak 1

Pred nadgradnjo je potrebno opraviti izvoz baze strežnika IMiS®/ARChive Server in narediti varnostno kopijo po postopku, kot je opisano v različnih poglavjih te dokumentacije.

Korak 2

Nadgradnja je lahko precej dolgotrajna, kar je odvisno predvsem od števila objektov, ki jih hrani strežnik IMiS®/ARChive Server. Nadgradimo ga z ukazom:

```
[user1@iarc ~]# sudo rpm -Uvh imisarc.9.1.1406-600.0001.el4.i386.rpm
```

Ob pravilnem delovanju proces nadgradnje izpiše vsebino, podobno spodnji (variacije so odvisne od distribucije Linux sistema):

```
Shutting down IMiS/ARChive HSM Storage Server:          [  OK  ]
Verifying IMiS/ARChive HSM Storage Server Database (BDB edition) integrity (this may take a while depending on your object store size)...
Database is consistent!
Exporting IMiS/ARChive HSM Storage Server Database (this may take a while depending on your object store size)...
Database Export succefull! Upgrade can proceed.
Performing POSTINSTALLATION Actions
Importing exported database files (this may take a while depending on your object store size)...
Done.
```

Korak 3

Po nadgradnji je smiselno preveriti uspešnost prenosa obstoječega stanja interne baze podatkov strežnika IMiS®/ARChive Server z izvozom baze podatkov in verifikacijo smiselnosti vpisov v tekstovne datoteke baze, preverjanjem lastništva datotek in imenikov volumnov in vsebine datoteke /etc/iarc.conf.

6.2 Možni zapleti pri nadgradnji

Opis pogostega zapleta 1

Ob poizkusu nadgradnje se pojavi napaka:

```
error: can't create transaction lock on /var/lib/rpm/.rpm.lock (Permission denied)
```

Razlog zapleta 1

Uporabnik, ki izvaja nadgradnjo, nima ustreznih pravic.

Rešitev zapleta 1

Za nadgradnjo se je potrebno prijaviti kot korenski uporabnik ali pa mora uporabnik uporabljati pripomočke, ki mu zagotovijo pravice ekvivalentne korenskemu (sudo).

Opis pogostega zapleta 2

Ob poizkusu nadgradnje se pojavi opozorilo:

```
Changing ownership of volume mountpoint "/iarc/vol/vol00" recursively to iarc:iarc (this may take a while).
WARNING: Operation failed. You will need to grant access to directories and objects for user iarc group iarc manually.
```

Razlog zapleta 2

Ob urejanju pravic volumen na lokaciji /iarc/vol/vol00 ni bil dosegljiv ali obstaja nek drug razlog, ki preprečuje korenskemu uporabniku spremembo lastništva datotek in imenikov volumna.

Rešitev zapleta 2

Potrebno je povezati diskovne pogone, kjer se nahajajo manjkajoči volumni (v našem primeru je med njimi /iarc/vol/vol00) in ročno nastaviti lastništvo nad imeniki in datotekami z ukazom:

[user1@iarc ~]# **sudo chown <iarc uporabnik>:<iarc skupina><pot> -R**

V našem primeru:

[user1@iarc ~]# **sudo chown iarc:iarc /iarc/vol/vol00 -R**

Opis pogostega zapleta 3

Ob poizkusu nadgradnje se pojavi napaka:

ERROR: IMiS/ARChive Storage Server BDB Database consistency check reported an error in one of the database entities. Manually run '/opt/IS/imisarc/dbtool -a check' from directory /iarc/db to get extended error information. IMiS/ARChive upgrade can proceed only when database is consistent. You need to manually verify and remove any inconsistency of the database. UPGRADE ABORTED!

Razlog zapleta 3

Interna baza strežnika IMiS®/ARChive Server ni dostopna ali pa je okvarjena.

Rešitev zapleta 3

Preverimo ali imenik, kjer se nahaja podatkovna baza strežnika obstaja. Preverimo tudi, da imenik ni prazen ter da so v njem urejene pravice uporabnika, ki izvaja procese strežnika.

Nato zaženemo ukaz:

[user1@iarc ~]# **sudo su – iarc –s /bin/bash –c "cd /opt/IS/imisarc && ./dbtool –f <pot do strežniške konfiguracijske datoteke> –h <pot do interne baze> -w <pot do interne baze> -a check**

V našem primeru:

[user1@iarc ~]# **sudo su – iarc –s /bin/bash –c "cd /opt/IS/imisarc && ./dbtool –f /etc/iarc.conf –h /iarc/db -w /iarc/db -a check**

Ukaz nam bo izpisal več podatkov o okvari baze. V primeru, da napaka presega obseg znanj administratorja, ki upravlja s produktom, lahko uporabi poti, ki jih določa sklenjena vzdrževalna pogodba ali drug dogovor in napako po nasvetu s strani proizvajalca odpravi s pomočjo vzdrževalnega osebja proizvajalca.

7 ODSTRANITEV

Postopek odstranitve strežnika IMiS®/ARChive Server lahko opravi korenski uporabnik ali uporabnik z ekvivalentnimi pravicami (administrator, šolano osebje, ...).

7.1 Postopek odstranitve

Korak 1

V rpm bazi preverimo verzijo nameščenega strežnika IMiS®/ARChive Server (izpis se lahko spreminja v odvisnosti od uporabljene distribucije):

```
[user1@iarc ~]# sudo rpm –q imisarc
imisarc-9.1.1406-600.i386
[user1@iarc ~]#
```

Korak 2

Ustavimo IMiS®/ARChive Server strežniško storitev.

To dejanje implicitno izvede sicer tudi dejanje odstranitve v primeru, da zazna delujočo in zagnano storitev.

Korak 3

Z rpm ukazom izvedemo deinstalacijo namestitvenega paketa strežnika IMiS®/ARChive Server.

Potrebno je vpisati celoten naziv iz rpm baze, kot je to razvidno iz Koraka 1:

```
[user1@iarc ~]# sudo rpm –e imisarc-9.1.1406-600.i386
This uninstall action WILL NOT:
  * remove IMiS/ARChive configuration file (e.g.: /etc/iarc.conf)
  * remove IMiS/ARChive database files
  * remove IMiS/ARChive log files (location set in /etc/iarc.conf)
  * remove IMiS/ARChive pid file (location in /var/run/iarc or overriden in /etc/iarc.conf)
  * remove IMiS/ARChive stored objects (on all your volume mountpoints)
  * remove IMiS/ARChive process user (iarc) and group (iarc) accounts from /etc/passwd and /etc/group

Above actions should be performed manually if required!

Uninstall complete.

[user1@iarc ~]#
```

8 UPRAVLJANJE PRODUKTA

S strežnikom IMiS®/ARChive Server lahko upravlja korenski uporabnik ali uporabnik z ekvivalentnimi pravicami (administrator, šolano osebje, ...). Zaradi enostavnosti uporabe lahko nekatere nastavitve upravlja tudi preko odjemalca IMiS®/Client :

- pravice dostopa uporabnikov in skupin do entitet in uporabniško določenih atributov;
- uporabniško določeni atributi;
- nabor vrednosti atributov;
- globine drevesa entitet v načrtu razvrščanja gradiva in načina zapisa klasifikacijske oznake za posamezno vrsto entitete na določenem nivoju;
- uporabniki in skupine s pripadajočimi podatki o uporabniku, avtentikaciji, vlogah in članstvu v skupinah;
- predloge z uporabniško določenimi atributi;
- profili strežnika;
- volumni strežnika.

Podrobnerjše infomacije so na voljo v poglavju Konfiguracija strežnika v uporabniškem priročniku IMiS®/Client.

8.1 Postopek zagona in zaustavitve

Za zagon in zaustavitev strežnika IMiS®/ARChive Server uporabimo zagonsko skripto.
Skripta `iarcd` se v primeru uporabe distribucije RHEL ali CentOS nahaja v imeniku `/etc/rc.d/init.d`, v primeru SLES ali OpenSuSE pa v imeniku `/etc/init.d`.
Zagonsko skripto uporabimo z orodjem `service` na naslednji način:

```
[user1@iarc ~]# sudo service iarcd <ukaz>
```

Veljavne vrednosti opcije `<ukaz>` zagonske skripte so:

start Ukaz zažene strežnik IMiS®/ARChive Server.
V primeru uspešnega zagona skripta izpiše:
Starting IMiS/ARChive HSM Storage Server: [OK],
v primeru neuspešnega zagona pa:
Starting IMiS/ARChive HSM Storage Server: [FAILED].

stop Ukaz zaustavi delovanje strežnika IMiS®/ARChive Server.
V primeru uspešne zaustavitve skripta izpiše:
Shutting down IMiS/ARChive HSM Storage Server: [OK],
v primeru neuspešne zaustavitve pa:
Shutting down IMiS/ARChive HSM Storage Server: [FAILED].

restart Ukaz izvede ponovni zagon (angl. Restart) strežnika IMiS®/ARChive Server.
Dejansko gre za sosledje ukazov `start` in `stop` zato so tudi izpisi na konzolo enaki, kot če bi zaporedno izvedli oba ukaza.

status Ukaz izpiše stanje servisa strežnika IMiS®/ARChive Server.
V primeru, da ta deluje izpiše tudi identifikacijski številki procesa:
Status of IMiS/ARChive HSM Storage Server: iarcd (pid 6222 6216) is running ...
v primeru, da je servis ustavljen pa:
Status of IMiS/ARChive HSM Storage Server: iarcd is stopped

Več informacij o morebitnih težavah pri zagonu strežnika IMiS/ARChive Server je na voljo v poglavju 9. Odpravljanje težav.

8.2 Beleženje dogodkov delovanja

Beleženje dogodkov je namenjeno preverjanju delovanja, ki jo občasno, oziroma po potrebi izvaja administrator strežnika ali administrator strežnika IMiS®/ARChive Server. IMiS®/ARChive Server beleži dogodke glede na nastavitev nivoja beleženja v nastavitveni datoteki `/etc/iarc.conf`. Privzeta lokacija dnevnikov je v imeniku `/var/log/iarc`. Aktivni dnevnik, v katerega IMiS®/ARChive Server beleži trenutne dogodke, se nahaja na lokaciji `/var/log/iarc/iarc.log`. Starejši dogodki so shranjeni v arhivskih datotekah dnevnika, ki se v odvisnosti od nastavitev ustvarjajo po potrebi z naslovno shemo `/var/log/iarc/iarc.XX.log` (XX = sekvenca arhivske datoteke, večja številka predstavlja dogodke, ki so se zgodili dlje v zgodovini). Beleženje v dnevnike IMiS®/ARChive Server izvaja po FILO principu »FILO – first in/last out« (prvi notri – zadnji ven).

Število arhivskih dnevnikov in njihovo velikost nastavimo v nastavitveni datoteki `/etc/iarc.conf`, v razdelku `[Log]`. Čas, ko informacija iz dnevnika izpade,
prilagajamo z nastavitvijo števila in velikosti dnevnikov glede na količino vpisov.
Dobra praksa narekuje nastavitve, ki omogočajo hranjenje informacij nivoja 6 najmanj tri mesece.
Nivojev beleženja je 7. Vsak nivo pomeni stopnjo podrobnosti informacij,
ki jih strežnik IMiS®/ARChive Server beleži v dnevnik.

Nivo 0 – Emergency

Zapisi t.i. "nultega"nivoja so napake, ki onemogočajo nadaljnje delovanje strežnika IMiS®/ARChive Server. Pomenijo težko napako ob verjetno okrnjeni celovitosti podatkov strežnika, zato se slednji ob nastanku takšne napake sam takoj zaustavi in nadaljevanje ni mogoče brez posega administratorja strežnika.
Vzroki za te napake so ponavadi zunanje narave npr. odpoved ključnih delov strojne opreme strežnika.

Nivo 1 – Alert

Zapisi prvega nivoja so napake pri katerih ni nujno, da strežnik IMiS®/ARChive Server preneha delovati. Delovati preneha, če bi ob nadaljevanju lahko prišlo do okvare na interni bazi strežnika ali objektih.
Vzrok za tako napako je lahko odpoved strojne opreme, zaznano nepravilno delovanje funkcij operacijskega sistema, preobremenitev strežnika ali poseganje drugega programa v okolje strežnika IMiS®/ARChive Server ali nepravilen poizkus konfiguriranja profilov in volumnov.

Nivo 2 – Critical

Zapisi drugega nivoja so napake pri katerih IMiS®/ARChive Server preneha delovati, saj bi lahko ob nadaljevanju prišlo do okvare na interni bazi strežnika ali objektih. Vzrok za tako napako je lahko zaznano nepravilno delovanje funkcij operacijskega sistema, pomanjkanje virov operacijskega sistema, preobremenitev strežnika ali zaznana napaka v delovanju strežnika.

Nivo 3 – Error

Zapisi tretjega nivoja so napake pri katerih je strežnik IMiS®/ARChive Server zaznal napako pri delovanju, ki pa ni kritična napaka in ne pomeni možnosti okvare podatkov. Vzroki za tako napako so lahko na strani odjemalca, posledica uporabe napačnih ali neprimernih nastavitvenih parametrov strežnika ali posledica napačnih vpisov v bazo.

Nivo 4 – Warning

Zapisi četrtega nivoja so opozorila pri katerih je strežnik IMiS®/ARChive Server zaznal nepravilnost, ki pa bistveno ne vpliva na delovanje strežnika in so največkrat posledica neregularnih zahtevkov odjemalcev, redkeje posledica napačnih vpisov v bazo strežnika ali nastavitveno datoteko.

Nivo 5 – Notice

Zapisi petega nivoja so zapisi o pomembnih regularnih (normalnih) dogodkih na strežniku IMiS®/ARChive Server, ki bi utegnile zanimati administratorsko osebje.

Nivo 6 – Info

Zapisi šestega nivoja so zapisi o manj pomembnih regularnih (normalnih) dogodkih na strežniku IMiS®/ARChive Server, ki bi utegnile zanimati administratorsko osebje.

Nivo 7 – Debug

Zapisi sedmega nivoja so razširjeni zapisi o vseh dogodkih na strežniku IMiS®/ARChive Server, ki jih uporabljamo pri zbiranju natančnejših podatkov o delovanju strežnika kadar vzroki za napako ali opozorilo niso evidentni.

Administrator strežnika IMiS®/ARChive Server mora biti pozoren na pojavljanje sporočil nivojev 4 do 0, saj ti lahko odkrivajo težave pri delovanju strežnika in delovanju strežnika ter komunikaciji z odjemalci.

8.3 Konfiguriranje

Konfiguriranje opravi korenski uporabnik `root` ali uporabnik s pravicami ekvivalentnimi korenskemu uporabniku preko orodja `sudo` tako, da se izvajajo s poverilnicami uporabniškega računa strežnika IMiS®/ARChive Server,
saj lahko v nasprotnem primeru pride do okvare interne baze.

8.3.1 Predvidena opravila

Konzolna orodja za delo z interno bazo strežnika IMiS®/ARChive Server so:

`dbtool` Orodje administratorju strežnika omogoča upravljanje z interno bazo podatkov. Proces izvažanja baze lahko izvajamo pri delujočem strežniku. Proces uvažanja baze pa se lahko izvaja le, ko strežnik ne deluje

Sintaksa:

usage: dbtool [-f db_config_file] [-q(uiet)] [-w working_dir] [-v version] [-o override][-a | -t tables[:name],...] exp | imp | init | del

Tables:
acle - acl entry table
aclv - acl entry validity table
at - attribute table
ag - attribute group table
av - attribute value table
ab - attribute binary table
dt - attribute date-time table
db - attribute double table
fi - attribute file table
i4 - attribute int32 table
i8 - attribute int64 table
i16 - attribute int128 table
i16i - attribute int128 index table
sm - attribute max string table
s10 - attribute value string 10
s20 - attribute value string 20
s30 - attribute value string 30
s40 - attribute value string 40
s50 - attribute value string 50
s100 - attribute value string 100
s200 - attribute value string 200
s20i - attribute value string 20 index
s30i - attribute value string 30 index
s40i - attribute value string 40 index
s50i - attribute value string 50 index
s100i - attribute value string 100 index
s200i - attribute value string 200 index
u4 - attribute value uint32
u8 - attribute value uint64
u16 - attribute value uint128
u16i - attribute value uint128 index
au - audit
cx - compression
mi - content type
cnt - counter table
dsc - digital certificate
cbdy - certificate body
ajts - aip job timestamp
dsi - digital signature
de - directory entry
dea - directory entry alias
dg - directory entry group
ent - entity
cl - entity class
do - entity document

fl - entity folder
lo - lookup table
h128 - 128 bit hash table
h160 - 160 bit hash table
h224 - 224 bit hash table
h256 - 256 bit hash table
h384 - 384 bit hash table
h512 - 512 bit hash table
rets - revocation data/timestamp bind table
resig - revocation data/digital signature bind table
tsaip - timestamp/aip bind table
cxl - compression library
pr - profile
prop - property
sd - storage driver
tm - template
ta - template attribute
tb - template bind
aip - archival information package
aipq - archival information package queue
caq - create aip queue
revo - revocation data
fti - full text indexing queue record
ts - timestamp
tsr - timestamp rule
vo – volume
fd - file description
null - attribute value null value
revq - review queue

GetStorageInfo

Orodje za konfiguracijo profilov in volumnov ter podatke o zasedenosti.

Sintaksa:

usage: GetStorageInfo [path-to-iarc.conf]

(skupaj s potjo, če se ta ne nahaja na privzeti lokacij – /etc].)

8.3.2 Postopki konfiguriranja s konzolnimi orodji

Izvoz interne baze strežnika opravimo z ukazom (izpis je odvisen od konfiguracije strežnika):

[user1@iarc ~]# **sudo su - iarc -s /bin/bash -c "cd /opt/IS/imisarc && ./dbtool -f /etc/iarc.conf -w /iarc/db/ -a exp"**

Using 'sl_SI.UTF-8' locale settings.
Document Root: /iarc/db/
Port: 21553

Exporting ACLENTRY (33 records) to aclentry.txt ... OK.
Exporting ACLENTRYVALIDITY (2 records) to aclentryvalidity.txt ... OK.
Exporting ATTRIBUTE (84 records) to attribute.txt ... OK.
Exporting ATTRIBUTEGROUP (0 records) to attributegroup.txt ... OK.
Exporting ATTRIBUTEVALUE (4569935 records) to attributevalue.txt ... OK.
Exporting ATTRIBUTEVALUEBINARY (1 records) to attributevaluebinary.txt ... OK.
Exporting ATTRIBUTEVALUEDATETIME (1232675 records) to attributevaluedt.txt ... OK.
Exporting ATTRIBUTEVALUEDOUBLE (0 records) to attributevaluedouble.txt ... OK.
Exporting ATTRIBUTEVALUEFILE (1310789 records) to attributevaluefile.txt ... OK.
Exporting ATTRIBUTEVALUEINT32 (820991 records) to attributevalueint32.txt ... OK.
Exporting ATTRIBUTEVALUEINT64 (821009 records) to attributevalueint64.txt ... OK.
Exporting ATTRIBUTEVALUEINT128 (1 records) to attributevalueint128.txt ... OK.
Exporting ATTRIBUTEVALUEINT128IDX (1 records) to attributevalueint128idx.txt ... OK.
Exporting ATTRIBUTEVALUESTRING (242028 records) to attributevaluemaxstring.txt ... OK.
Exporting ATTRIBUTEVALUESTRING10 (0 records) to attvalstr10.txt ... OK.
Exporting ATTRIBUTEVALUESTRING20 (1 records) to attvalstr20.txt ... OK.
Exporting ATTRIBUTEVALUESTRING30 (1 records) to attvalstr30.txt ... OK.
Exporting ATTRIBUTEVALUESTRING40 (1 records) to attvalstr40.txt ... OK.
Exporting ATTRIBUTEVALUESTRING50 (619562 records) to attvalstr50.txt ... OK.
Exporting ATTRIBUTEVALUESTRING100 (39 records) to attvalstr100.txt ... OK.
Exporting ATTRIBUTEVALUESTRING200 (649035 records) to attvalstr200.txt ... OK.
Exporting ATTRIBUTEVALUESTRING20IDX (1 records) to attvalstr20idx.txt ... OK.
Exporting ATTRIBUTEVALUESTRING30IDX (8 records) to attvalstr30idx.txt ... OK.
Exporting ATTRIBUTEVALUESTRING40IDX (1 records) to attvalstr40idx.txt ... OK.
Exporting ATTRIBUTEVALUESTRING50IDX (1 records) to attvalstr50idx.txt ... OK.
Exporting ATTRIBUTEVALUESTRING100IDX (68 records) to attvalstr100idx.txt ... OK.
Exporting ATTRIBUTEVALUESTRING200IDX (31426 records) to attvalstr200idx.txt ... OK.
Exporting ATTRIBUTEVALUEUINT32 (2565834 records) to attvaluint32.txt ... OK.
Exporting ATTRIBUTEVALUEUINT64 (30944 records) to attvaluint64.txt ... OK.
Exporting ATTRIBUTEVALUEUINT128 (1 records) to attvaluint128.txt ... OK.
Exporting ATTRIBUTEVALUEUINT128IDX (1 records) to attvaluint128idx.txt ... OK.
Exporting audit log data (9544057 records) to auditlog.bin ... OK.
Exporting DIGITALCERTIFICATE (40 records) to digitalcert.txt ... OK.
Exporting COUNTER (4756 records) to counter.txt ... OK.
Exporting DIGITALSIGNATURE (476 records) to digitalsig.txt ... OK.
Exporting DIRECTORYENTRY (89 records) to direntry.txt ... OK.

Exporting DIRECTORYENTRYALIAS (42 records) to direntryalias.txt ... OK.
Exporting DIRECTORYENTRYGROUP (80 records) to direntrygroup.txt ... OK.
Exporting ENTITY (1245268 records) to entity.txt ... OK.
Exporting ENTITYCLASS (127 records) to entityclass.txt ... OK.
Exporting ENTITYDOCUMENT (1237320 records) to entitydocument.txt ... OK.

```
Exporting ENTITYFOLDER (7787 records) to entityfolder.txt ... OK.
Exporting PROFILE (4 records) to profile.txt ... OK.
Exporting PROPERTY (15 records) to property.txt ... OK.
Exporting TEMPLATE (33 records) to template.txt ... OK.
Exporting TEMPLATEATTRIBUTE (142 records) to templateatt.txt ... OK.
Exporting TEMPLATEBIND (22 records) to templatebind.txt ... OK.
Exporting ARCHIVALINFORMATIONPACKAGE (589815 records) to aip.txt ... OK.
Exporting ARCHIVALINFORMATIONPACKAGEQUEUE (0 records) to aipq.txt ... OK.
Exporting CREATEAIPQUEUE (0 records) to createaipqueue.txt ... OK.
Exporting AIPJOBTIMESTAMP (0 records) to aipjobts.txt ... OK.
Exporting REVOCATIONDATA (1034 records) to revodata.txt ... OK.
Exporting CONTENTPROCESSINGQUEUE (0 records) to cpqueue.txt ... OK.
Exporting CONTENTPROCESSINGERROR (1 records) to cperror.txt ... OK.
Exporting TIMESTAMP (476 records) to timestamp.txt ... OK.
Exporting VOLUME (12 records) to volume.txt ... OK.
Exporting LOOKUPTABLE (12 records) to lookup.txt ... OK.
Exporting HASH128 (1 records) to hash128.txt ... OK.
Exporting HASH160 (524494 records) to hash160.txt ... OK.
Exporting HASH224 (1 records) to hash224.txt ... OK.
Exporting HASH256 (65322 records) to hash256.txt ... OK.
Exporting HASH384 (1 records) to hash384.txt ... OK.
Exporting HASH512 (1 records) to hash512.txt ... OK.
Exporting REVOCATIONDATATIMESTAMP (657 records) to revodatats.txt ... OK.
Exporting REVOCATIONDATADIGITALSIGNATURE (0 records) to revodatadsig.txt ... OK.
Exporting ENTITYRETENTIONDISPOSITIONSCHEDULE (86 records) to rds.txt ... OK.
Exporting TIMESTAMPARCHIVALINFORMATIONPACKAGE (589814 records) to tsaip.txt ... OK.
Exporting CERTIFICATEBODY (14302 records) to certbody.txt ... OK.
Exporting FILEDESCRIPTION (348 records) to filedesc.txt ... OK.
Exporting ATTRIBUTEVALUENULL (124334 records) to attvalnull.txt ... OK.
Exporting REVIEWQUEUE (0 records) to reviewqueue.txt ... OK.
Exporting CONTENTBIND (31 records) to contentbind.txt ... OK.
Export command completed  with no errors.
```

Opomba: Številke (XX records) pri vsaki vrstici se lahko spreminjajo, odvisno od števila zapisov.

Rezultat je 74 tekstovnih datotek, ki predstavljajo posamezne tabele v bazi, binarna datoteka, ki predstavlja revizijsko sled ter dodatni podatki (shranjeni v datotekah brez končnic, datoteke se začnejo s črko, kateri sledi zaporedna številka – primer: »p0« za blob iz PROPERTY tabele), ki vsebujejo del interne baze strežnika IMiS®/ARChive Server.

8.4 Administracija

Administracijska opravila lahko bistveno vplivajo na delovanje produkta.
Pravilna namestitev in konfiguracija sistema lahko zagotovi stabilno in pričakovano delovanje produkta z malo ali nič vzdrževalnimi posegi, v primeru napačne ali neskladne konfiguracije pa sistem lahko kompromitira in naredi nestabilnega, počasi delujočega ali varnostno ranljivega.
Zato morajo biti posegi v administracijo produkta omejeni na izobražene administratorje, ki so v podrobnosti seznanjeni z navodili proizvajalca in splošnimi dobrimi praksami pri načrtovanju in vzdrževanju informacijskih sistemov.
Ob nakupu svetujemo tudi vzpostavitev vzdrževalne pogodbe s proizvajalcem, ki zagotavlja brezhibno in neprekinjeno delovanje »mission-critical« sistema, kar naj bi arhivski sistem predstavljal.
Administrator strežnika IMiS®/ARChive Server nastavlja parametre v nastavitveni datoteki glede na porabo virov strežnika, ki jih spremlja s periodičnim pregledovanjem stanja strežnika, zapisov v dnevnikih in zahtev aplikacijskega okolja.

Administrator po navodilih proizvajalca upravlja z nastavitvenimi parametri sistema, ki vplivajo na načrt razvrščanja gradiva, predloge entitet, metapodatkovne sheme atributov, nastavitve vtičnikov za različne podporne storitve sistema, itd. Ti posegi se izvajajo s pomočjo administracijskega vmesnika v okviru produkta IMiS®/Client, ki je opisan v priročniku produkta.
Administratorjem uporabnikov nudimo izobraževanja za upravljanje s produktom, strankam pa predlagamo vzdrževalno pogodbo, saj izvajamo profesionalne storitve upravljanja s produktom. Te zagotavljajo nemoteno delovanje strežnika skozi daljše obdobje.

Pred vsakim posegom je smiselno poskrbeti za varnostno kopijo interne baze strežnika in nastavitvene datoteke v primeru potrebe po restavraciji starih nastavitev zaradi nepravilne prilagoditve nastavitev in/ali baze.

8.4.1 Konfiguracijska datoteka iarc.conf

Nastavitveni parametri strežnika IMiS®/ARChive Server so nastavljivi v datoteki /etc/iarc.conf.

Za spodnje opise so uporabljene privzete vrednosti.

Parametri so razvrščeni v sekcije po namembnosti in jih podrobneje obravnavamo v nadaljevanju, v nadaljevanju pa so razdeljene po principu:

Ključ:	Opis (opcijsko nabor veljavnih, minimalnih, maksimalnih, priporočenih vrednosti)

Sekcija [Server]

Path:	Označuje absolutno pot do izvajalnih datotek (programov) in knjižnic strežnika. Privzeta vrednost je /opt/IS/imisarc
ConnChilds:	Označuje število hkratnih povezovalnih procesov strežnika. Privzeta in priporočena vrednost je 1, vrednost lahko povečamo, v kolikor število zahtevanih hkratnih sej z odjemalci naraste čez 1024 za vsakih 1024 novih sej. Vrednost navzgor ni omejena, vrednosti nad 10 niso performančno smiselne in lahko vodijo do težav.
ReqThreads:	Označuje število niti, ki izvajajo zahtevke. Privzeta vrednost je 7 in navzgor ni omejena. Priporočena vrednost znaša dvakratnik števila niti, ki jih je strežnikov procesor zmožen izvajati hkrati in je odvisen od števila procesorskih jeder.
StatisticsCycle:	Označuje število sekund med opravilom izračuna statistik, ki jih IMiS®/ARChive Server vodi o delovanju. Privzeta in priporočena vrednost je 180000. Spreminjanje vrednosti navzdol lahko vpliva na odzivnost strežnika. Najmanjša vrednost je 1, največja 16777216.
ClientTimeout:	Označuje čas neaktivnosti prijavljenega odjemalca v sekundah, ki mora preteči preden strežnik prekine sejo. Privzeta vrednost je 3600 sekund. Najmanjša možna vrednost je 1800 in največja 86400.
NoAuthClientTimeout:	Označuje čas neaktivnosti neprijavljenega (anonimnega) odjemalca v sekundah, ki mora preteči preden strežnik prekine sejo. Privzeta vrednost je 120 sekund. Najmanjša možna vrednost je 5 in največja 3600. Skozi anonimne seje je možno s strežnikom izmenjati le najosnovnejše in neobčutljive informacije (naziv arhiva, opis, ipd.).

IdentPassword:	Označuje šifrirano zgoščeno vrednost gesla, ki ga strežnik uporablja pri delovanju v procesih šifriranja. V kolikor trajno hranimo notranje identifikatorje za entitete v drugih sistemih, vrednosti ne smemo spreminjati po arhiviranju prve entitete, saj vrednost vpliva na šifrirni algoritem za generiranje notranjih entitetnih identifikatorjev.
Port:	Označuje številko TCP vrat, na katerih povezovalni proces strežnika pričakuje zahtevke odjemalcev. Privzeta vrednost je 16807.
Listen:	Označuje omrežni naslov na katerega strežnik veže TCP vrata na katerih pričakuje zahtevke odjemalcev. Vrednosti so lahko le v skladu IPv4 ali IPv6 naslovno shemo. Primeri veljavnih vrednosti: 192.168.92.32 fd00:192:168:92::32 192.168.92.32:16807 [fd00:192:168:92::32]:16807 [fd00:192:168:92:2340:efa1:1244:32] fd00:192:168:92:2340:efa1:1244:32 [fd00:192:168:92:2340:efa1:1244:32]:12345 [::ffff:192.168.92.12] ::ffff:192.168.92.12 [::ffff:192.168.92.12]:65743 localhost [::]
CXLib:	Označuje relativno pot do knjižnic, ki vsebujejo kompresijske metode za različne platforme odjemalcev, znotraj poti, ki jo označuje vrednost parametra DataPath opisanega spodaj.
PartialTimeout:	Označuje čas v sekundah, v katerem se mora odjemalec odzvati na zahtevek strežnika. Po preteku tega časa strežnik zaključi sejo. Najmanjša vrednost je 1, največja je 60 in priporočena 5.
PidPath:	Označuje pot do identifikacijske datoteke glavnega procesa strežnika. Privzeta vrednost je /var/run/iarc.

CountryLanguage:	Označuje regionalne nastavitve v POSIX formatu: xx_YY[.CHARSET[@variant]]; prvi dve črki xx predstavljata kodo jezika po standardu ISO-639, drugi dve črki predstavljata kodo države po standardu ISO-3166, CHARSET podatek (opcijski) določa kodno tabelo oziroma kodni razpored (seznam dostopen v imeniku /usr/share/i18n/charsets), podatek variant (opcijski) pa določa nacionalne posebnosti jezika. Seznam možnih nastavitev je možno pridobiti preko izpisa ukaza "locale –a". Vrednost določa nabor pravil s katerimi naj strežnik izvaja vse operacije z znakovnimi nizi, ki so lahko "občutljivi" na različne nacionalne znake (sortiranje, prevajanje, zlaganje, itd). Privzeta vrednost je "", kar efektivno določa sistemsko nastavljena pravila (glej sistemski ukaz "locale" http://www.linuxmanpages.com/man1/locale.1.php). *Posebnost: Zaradi zagotavljanja prenosljivosti shranjenih podatkov v notranji bazi storitev, ne glede na določene regionalne nastavitve, interno uporablja UTF-8 kot notranjo kodno tabelo. Ostale regionalne nastavitve ohrani (jezik, država, jezikovna posebnost).*
DataPath:	Označuje področje na disku, kjer strežnik hrani trajne podatke, povezane z arhivom kot so statistični podatki transakcij, ipd. Privzeta vrednost je /iarc/db.
MaxRequestSize:	Določa največji možen enkratni zahtevek odjemalca. Privzeta vrednost je 3276800 bajtov. Najmanjša možna vrednost je 32768 in največja 32768000.
JVMHome:	Označuje absolutno pot do javanskega okolja. Privzeta vrednost je /opt/IS/imisarc/jvm.
JVMOptions:	Označuje konfiguracijske parametre za javansko okolje.
Mode:	Določa način, kako lahko odjemalec komunicira s strežnikom. Vrednosti so naslednje: 0 – komunikacija poteka po nezaščitenem kanalu, 1 - strežnik omogoča komunikacijo tako po nezaščitenem kakor tudi po zaščitenem kanalu, 2 – komunikacija poteka samo po zaščitenem kanalu. Privzeta vrednost je 0.
SecureListen:	Označuje omrežni naslov na katerega strežnik veže TCP vrata, na katerih pričakuje zahtevke odjemalcev (zaščiten komunikacijski kanal). Za podrobnosti glej opis pod ključem Listen.
SecurePort:	Označuje številko TCP vrat, na katerih povezovalni proces strežnika pričakuje zahtevke odjemalcev (zaščiten komunikacijski kanal).

SecureCertKeyFile:	Označuje absolutno pot do ključa digitalnega potrdila, ki se uporablja v procesu vzpostavitve zaščitenega komunikacijskega kanala. Privzeta vrednost je /opt/IS/imisarc/cert.key.
SecureCertFile:	Označuje absolutno pot do digitalnega potrdila, ki se uporablja v procesu vzpostavitve zaščitenega komunikacijskega kanala. Privzeta vrednost je /opt/IS/imisarc/cert.crt.
SecureCertChainFile:	Označuje absolutno pot do datoteke, ki vsebuje zaupanja vredno verigo digitalnih potrdil, ki se preverjajo v procesu vzpostavitve zaščitenega komunikacijskega kanala.
SecureProtocol:	Označuje kriptografski protokol, ki se uporablja za zaščito komunikacijskega kanala. Privzeta vrednost je TLSv1.
SecureCiphers:	Označuje omogočene kriptografske algoritme za šifriranje podatkov.
SecureVerifyClientMode:	Določa način preverjanja odjemalčevega digitalnega potrdila v procesu vzpostavitve zaščitenega komunikacijskega kanala. Vrednosti so naslednje: 0 – odjemalčevo digitalno potrdilo ni potrebno, 1 – odjemalčevo digitalno potrdilo je opcijsko (če ga odjemalec posreduje ga strežnik preveri), 2 – odjemalec mora posredovati digitalno potrdilo ki se nato na strežniku preveri.
SecureHandshakeTimeout:	Označuje čas (v sekundah), v katerem morata odjemalec in strežnik vzpostaviti zaščiten komunikacijski kanal. Privzata vrednost je 60 sekund, najmanjša vrednost je 5 sekund, največja pa 3600.
SecureRevocationCheck:	Določa način preverjanja odjemalčevega digitalnega potrdila v povezavi s preklicnimi listami izdajateljev digitalnih potrdil. Vrednosti so naslednje: 0 – (OFF) preklicanosti odjemalčevega digitalnega potrdila se ne preverja, 1 – (OPTIONAL) preklicanost odjemalčevega digitalnega potrdila se preveri. V primeru kakršnekoli napake pri tem postopku, strežnik povezavo vseeno dovoli, 2 – (REQUIRED-CACHED) stanje preklicanosti odjemalčevega digitalnega potrdila se obvezno preveri. Strežnik lahko uporabi predhodno pridobljeno informacijo, v kolikor je še veljavna, 3 – (REQUIRED-FRESH) stanje preklicanosti odjemalčevega digitalnega potrdila se obvezno preveri. Strežnik vselej uporabi sveže informacije o preklicanosti.

Sekcija [Database]

ProviderBootstrapFile:	Označuje relativno pot do nastavitvene datoteke vtičnika za servis podatkovne baze. Relativna je glede na vrednost parametra [Server].Path.

Sekcija [Cache]

ReadPath:	Označuje pot do imenika, ki predstavlja medpomnilnik, kjer strežnik IMiS®/ARChive Server odlaga objekte, ko jih je potrebno tja začasno odložiti zaradi hitrejšega posredovanja objektov odjemalcem. Pravice na poti morajo biti urejene tako, da lahko uporabnik, ki izvaja procese strežnika, tam bere in zapisuje datoteke.
ReadSize:	Označuje najmanjšo velikost medpomnilnika. Strežnik to omejitev prilagaja dinamično, glede na potrebe. Spreminjanje vrednosti ni smiselno.
EditPath:	Označuje pot do imenika, ki predstavlja medpomnilnik, kjer strežnik začasno odlaga objekte, ki mu jih posredujejo odjemalci. Pravice na poti morajo biti urejene tako, da lahko uporabnik, ki izvaja procese strežnika, tam lahko bere in zapisuje datoteke.
EditSize:	Označuje najmanjšo velikost medpomnilnika. Strežnik to omejitev prilagaja dinamično, glede na potrebe. Spreminjanje vrednosti ni smiselno.

Sekcija [Log]

LogFile:	Označuje osnovno ime dnevniške datoteke, skupaj s potjo, v katero strežnik IMiS®/ARChive Server beleži dogodke. Pravice na poti in datoteki morajo biti urejene tako, da lahko uporabnik, ki izvaja procese strežnika, v njo zapisuje datoteke in po potrebi ustvarja nove datoteke.
MaxSize:	Označuje največjo velikost ene dnevniške datoteke v bajtih. Privzeta in priporočena vrednost je 1000000, najmanjša 65536 in največja 2147483648.
BackupCount:	Označuje število arhivskih dnevniških datotek, znotraj katerega strežnik zapisuje dogodke po algoritmu prvi notri – zadnji ven. Privzeta vrednost je 1, priporočena 9.
LogLevel:	Nivo dogodkov, ki jih strežnik zapisuje v dnevniško datoteko (glej poglavje 8.2 Beleženje dogodkov delovanja). Najmanjša vrednost je 1, največja 7, priporočena 6.

Sekcija [AuditLog]

AuthCryptoModes:	Označuje nabor možnih kriptografskih metod, ki jih strežnik dovoli uporabiti za šifriranje avtentikacijskih sporočil in kasnejše komunikacije z odjemalcem, ki želi izvesti vpogled v revizijsko sled. Identifikator predstavlja kombinacijo algoritma, dolžine ključa in kriptografski način obdelave paketa podatkov. Veljavne vrednosti so: aes-256-cbc aes-256-ecb aes-256-ofb aes-256-cfb aes-192-cbc aes-192-ecb aes-192-ofb aes-192-cfb aes-128-cbc aes-128-ecb aes-128-ofb aes-128-cfb
AuthPreSharedKey:	Predpomneni strežniški ključ, ki se uporabi za šifriranje avtentikacijskih sporočil in kasnejše komunikacije z odjemalcem, ki predstavlja pooblaščeno osebo za namen vpogleda v revizijsko sled beleženja dogodkov povezanih s sejami in/ali objekti.

Sekcija [Authentication]

Methods:	Označuje možen nabor metod za vzpostavitev seje odjemalca IMiS®s strežnikom IMiS®/ARChive Server. Veljaven nabor vrednosti je: basic advanced psk srp6a Osnovna (basic) predstavlja starejšo metodo vzpostavitve seje s strežnikom, brez posredovanja evidenčnih podatkov o odjemalcu. Pri napredni (advanced) je vključena zapletenejša (HMAC) metoda vzpostavitve seje s strežnikom, ki predvideva obvezne in neobvezne metapodatke o odjemalcu. Pri uporabi deljenega ključa (psk) napredno metodo dodatno zaščitimo s šifrirano izmenjavo omrežnih avtentikacijskih paketov. Uporaba prijavnih podatkov uporabnika »User Credentials« (srp6a) pa omogoča individualno prijavo uporabnikov, kjer se podatki o uporabniku pridobijo na podlagi uspešne avtentikacije uporabnika iz internih virov (imenika).

CryptoModes:	Označuje nabor možnih metod šifriranja, ki jih strežnik uporabi za šifrirano komunikacijo z odjemalcem. Identifikator predstavlja kombinacijo algoritma, dolžine ključa in kriptografski način obdelave paketa podatkov. Veljavne vrednosti so: aes-256-cbc aes-256-ecb aes-256-ofb aes-256-cfb aes-192-cbc aes-192-ecb aes-192-ofb aes-192-cfb aes-128-cbc aes-128-ecb aes-128-ofb aes-128-cfb
PreSharedKey:	Predpomnjeni strežniški ključ, ki se uporabi za šifriranje avtentikacijskih sporočil (prijavna metoda psk) in kasnejše komunikacije z odjemalcem, ki predstavlja navadnega, ne-priviligiranega IMiS® odjemalca.
RequiredParams:	Nabor zahtevanih podatkov, ki jih mora posredovati odjemalec pri odpiranju seje ali dogodka. V kolikor manjka en sam zahtevan podatek, strežnik zavrne vzpostavitev seje ali dogodka. Veljaven nabor zahtevanih podatkov lahko vsebuje vsaj enega izmed: username (uporabniško ime uporabnika, ki izvaja operacijo) computername (ime računalnika, iz katerega operacija izvira) message (razlog/sporočilo, ki ga uporabnik vpiše pri izvedbi operacije) Privzeta nastavitev je prazno – ne vsebuje nobenega iz nabora.

Sekcija [Admin]

Enabled:	Označuje vrednost, ki pove ali je administracijskih modul omogočen ali ne. Vrednosti so naslednje: 0 – administracijski modul ni omogočem, 1 – administracijski modul je omogočen. Privzata vrednost je 0.
IdleTimeout:	Označuje čas v sekundah, po katerem se administracijska seja zaradi nedejavnosti prekine. Privzeta vrednost je 900 sekund.
WebPath:	Označuje relativno pot, ki predstavlja spletni naslov administracijskega vmesnika. Privzeta vrednost je /admin.
Port:	Označuje številko TCP vrat, na katerih povezovalni proces strežnika pričakuje zahtevke odjemalcev za administracijo.
Listen:	Označuje omrežni naslov na katerega strežnik veže TCP vrata na katerih pričakuje zahtevke odjemalcev za administracijo. Za podrobnosti glej opis v sekciji Server pod ključem Listen.

MaxSessionsPerUser:	Označuje največje število vzporednih adminitracijskih sej za posameznega uporabnika. Privzeta vrednost je 5 administracijskih sej.
MaxRequestSize:	Določa največji možen enkratni zahtevek odjemalca. Privzeta vrednost je 1048576 bajtov. Najmanjša možna vrednost je 10240 in največja 10485760.
Authenticate:	Označuje vrednost, ki pove ali je za dostop do administracije potrebna avtentikacija: Vrednosti so naslednje: 0 – avtentikacija ni potrebna, 1 – avtentikacija je potrebna. Privzeta vrednost je 1.
Secure:	Označuje vrednost, ki pove ali komunikacije med odjemalcem in strežnikom poteka po zaščitenem kanalu ali ne. Vrednosti so: 0 – komunikacija med odjemalcem in strežnikom poteka po nezaščitenem kanalu, 1 – komunikacija med odjemalcem in strežnikom poteka po zaščitenem kanalu. Privzeta vrednost je 0.
SecureCertKeyFile:	Označuje absolutno pot do ključa digitalnega potrdila, ki se uporablja v procesu vzpostavitve zaščitenega komunikacijskega kanala. Privzeta vrednost je /opt/IS/imisarc/cert.key.
SecureCertKeyPassword:	Označuje geslo za dostop do ključa digitalnega potrdila.
SecureCertFile:	Označuje absolutno pot do digitalnega potrdila, ki se uporablja v procesu vzpostavitve zaščitenega komunikacijskega kanala. Privzeta vrednost je /opt/IS/imisarc/cert.crt.
SecureCertChainFile:	Označuje absolutno pot do datoteke, ki vsebuje zaupanja vredno verigo digitalnih potrdil. Preverjajo se v procesu vzpostavitve zaščitenega komunikacijskega kanala.
SecureProtocol:	Označuje kriptografski protokol, ki se uporablja za zaščito komunikacijskega kanala. Privzeta vrednost je TLSv1.
SecureCiphers:	Označuje omogočene kriptografske algoritme za šifriranje podatkov.
SecureVerifyClientMode:	Določa način preverjanja odjemalčevega digitalnega potrdila v procesu vzpostavitve zaščitenega komunikacijskega kanala. Vrednosti so naslednje: 0 – odjemalčevo digitalno potrdilo ni potrebno, 1 – odjemalčevo digitalno potrdilo je opcijsko (če ga odjemalec posreduje ga strežnik preveri), 2 – odjemalec mora posredovati digitalno potrdilo ki se nato na strežniku preveri.
SecureHandshakeTimeout:	Označuje čas (v sekundah), v katerem morata odjemalec in strežnik vzpostaviti zaščiten komunikacijski kanal. Privzata vrednost je 60 sekund, najmanjša vrednost je 5 sekund, največja pa 3600.

SecureRevocationCheck:	Določa način preverjanja odjemalčevega digitalnega potrdila v povezavi s preklicnimi listami izdajateljev digitalnih potrdil. Vrednosti so naslednje: 0 – (OFF) preklicanosti odjemalčevega digitalnega potrdila se ne preverja, 1 – (OPTIONAL) preklicanost odjemalčevega digitalnega potrdila se preveri. V primeru kakršnekoli napake pri tem postopku, strežnik povezavo vseeno dovoli, 2 – (REQUIRED-CACHED) stanje preklicanosti odjemalčevega digitalnega potrdila se obvezno preveri. Strežnik lahko uporabi predhodno pridobljeno informacijo, v kolikor je še veljavna, 3 – (REQUIRED-FRESH) stanje preklicanosti odjemalčevega digitalnega potrdila se obvezno preveri. Strežnik vselej uporabi sveže informacije o preklicanosti.

9 ODPRAVLJANJE TEŽAV

V primeru težav in napak je pomembno, da administratorji in uporabniki postopajo pravilno.
Z morebitnim nestrokovnim posegom lahko pride do dodatnega poslabšanja stanja strežnika IMiS®/ARChive Server, s tem pa tudi do težje odprave napake.
Uporabniki/Administratorji morajo biti seznanjeni s pravilnim načinom uporabe produkta in postopati v skladu z uporabniško dokumentacijo.
Priporočljivo je, da se ob morebitnih težavah obrnejo na ustrezno strokovno osebo v organizaciji (sistemske administratorje). Sistemskim administratorjem svetujemo,
da tudi s pomočjo dokumentacije ugotovijo mesto napake in se po potrebi o nadaljnjih korakih posvetujejo z našimi strokovnjaki.

9.1 Kako se težavam izognemo?

Redni periodični pregledi delovanja strežnika IMiS®/ARChive Server so bistvenega pomena pri pravočasnemu odkrivanju morebitnih težav in napak v delovanju.

Mednje sodijo tudi pregledi usklajenosti diskovnega sistema (samostojni disk ali diskovno polje) in datotečnega sistema. Težavam z diskovnim sistemom se izognemo tudi tako, da izberemo zanesljivo strojno opremo in poskrbimo, da je uporabljeno diskovje na strežnik priklopljeno lokalno s primerno redundanco.
Izogibamo se diskovnim sistemom NAS ali souporabi diskov na drugih strežnikih oziroma diskovju, ki je dosegljivo preko lokalne mreže.

Periodično preverjamo tudi konsistentnost interne baze strežnika. Več informacij o pogostih vzrokih je na voljo v poglavju 8.3 Konfiguriranje.

Bistvenega pomena je tudi opcijska veljavna vzdrževalna pogodba, ki zagotavlja minimalne odzivne čase v primeru težjih napak ali izpada sistema.

9.2 Pogoste težave

Opis pogoste težave 1

Na odjemalcih IMiS®/Scan ali IMiS®/View se ob poizkusu pregledovanja objekta shranjenega na strežniku IMiS®/ARChive Server pojavi »Napaka 61523«.
Drugi odjemalci (npr. IMiS®/Storage Connector) javljajo napako:

> IMiS/ARC Client <IASession.Open> Failed to establish connection to the cluster node <10.1.1.10, 16807> (Reason: Error <TimedOut> occurred while opening network connection.).

Strežnik je sicer omrežno dostopen, servis na vratih, kjer posluša, pa ni dostopen (preverjanje s telnet programom)

```
[user1@test ~]# ping iarc.acme.com
PING iarc.acme.com (10.1.1.10) 56(84) bytes of data.
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=1 ttl=64 time=0.653 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=2 ttl=64 time=0.190 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=3 ttl=64 time=0.186 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=4 ttl=64 time=0.183 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=5 ttl=64 time=0.164 ms
```

```
--- iarc.acme.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 0.164/0.450/1.530/0.540 ms

[user1@test ~]# telnet iarc.acme.com 16807
Trying 10.0.0.10...
.. (daljši premor) ...
telnet: connect to address 10.1.1.10: Connection timed out
[user1@test ~]#
```

Strežnik sicer deluje, kar preverimo s konzolnim ukazom na strežniku, kjer je nameščen:

```
[user1@iarc ~]# sudo service iarcd status
Status of IMiS/ARChive HSM Storage Server: iarcd (pid 23209 23203) is running...
[user1@iarc ~]#
```

Vzrok težave 1

Požarni zid na strežniku ali omrežju med odjemalcem in strežnikom preprečuje komuniciranje odjemalcev IMiS® s strežnikom IMiS®/ARChive Server preko TCP vrat 16807 ali drugih, v primeru drugačne nastavitve TCP vrat v /etc/iarc.conf datoteki.

Rešitev težave 1

Požarni zid je potrebno ponovno nastaviti tako, da bo dovoljeval komunikacijo odjemalcev IMiS® s strežnikom.

Opis pogoste težave 2

Ob poizkusu shranjevanja novega objekta, strežnik vrne odgovor:

> You cannot create an entity with template »%TEMPLATE_NAME%« under specified parent since it's not included in the list of allowed templates.

Strežnik je sicer omrežno dostopen, servis na vratih, kjer posluša, se odziva (preverjanje s telnet programom)

```
[user1@test ~]# ping iarc.acme.com
PING iarc.acme.com (10.1.1.10) 56(84) bytes of data.
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=1 ttl=64 time=0.653 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=2 ttl=64 time=0.190 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=3 ttl=64 time=0.186 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=4 ttl=64 time=0.183 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=5 ttl=64 time=0.164 ms
```

```
--- iarc.acme.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 0.164/0.450/1.530/0.540 ms

[user1@test ~]# telnet iarc.acme.com 16807
Trying 10.1.1.10...
Connected to iarc.acme.com.
Escape character is '^]'.
Connection closed by foreign host.
[user1@test ~]#
```

Strežnik deluje, kar dodatno preverimo s konzolnim ukazom na strežniku, kjer je nameščen:

```
[user1@iarc ~]# sudo service iarcd status
Status of IMiS/ARChive HSM Storage Server: iarcd (pid 23209 23203) is running...
[user1@iarc ~]#
```

Vzrok težave 2

Odjemalec skuša shraniti objekt z obstoječo predlogo, ki ni uvrščena v seznam dovoljenih predlog na nadrejeni entiteti.

Rešitev težave 2

Preveriti je potrebno seznam dovoljenih predlog na nadrejeni entiteti, klasifikacijsko oznako nadrejene entitete ter samo predlogo. V primeru, da je uporabljena napačna predloga ali pa je odjemalec poskušal uvrstiti entiteto na napačno mesto v načrtu razvrščanja gradiva, je potrebno popraviti napako na odjemalcu in znova shraniti objekt. Drugače je potrebno dodati predlogo v seznam dovoljenih predlog.

Opis pogoste težave 3

Ob poizkusu shranjevanja novega objekta na strežnik IMiS®/ARChive Server z odjemalcem IMiS®/Scan se pojavi »Napaka #14«. Drugi odjemalci (npr. IMiS®/Storage Connector) pri poizkusu shranitve objekta s strežnika dobijo naslednji odgovor:

```
Not enough space on storage profile »%PROFILE_NAME%«. Required %FILE_SIZE% bytes..
```

Strežnik je sicer omrežno dostopen, servis na vratih, kjer posluša, se odziva (preverjanje s telnet programom):

```
[user1@test ~]# ping iarc.acme.com
PING iarc.acme.com (10.1.1.10) 56(84) bytes of data.
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=1 ttl=64 time=0.653 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=2 ttl=64 time=0.190 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=3 ttl=64 time=0.186 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=4 ttl=64 time=0.183 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=5 ttl=64 time=0.164 ms


--- iarc.acme.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 0.164/0.450/1.530/0.540 ms

[user1@test ~]# telnet iarc.acme.com 16807
Trying 10.1.1.10...
Connected to iarc.acme.com.
Escape character is '^]'.
Connection closed by foreign host.
[user1@test ~]#
```

Strežnik deluje, kar dodatno preverimo s konzolnim ukazom na strežniku, kjer je nameščen:

```
[user1@iarc ~]# sudo service iarcd status
Status of IMiS/ARChive HSM Storage Server: iarcd (pid 23209 23203) is running...
[user1@iarc ~]#
```

Vzrok težave 3

Prišlo je do zapolnitve vseh volumnov znotraj profila za shranjevanje,

ki je bil uporabljen ob shranjevanju novega objekta.

Rešitev težave 3

Profilu strežnika, ki mi je zmanjkalo prostora dodamo ustrezno število novih volumnov.

Opis pogoste težave 4

Ob zagonu strežnika IMiS®/ARChive Server se na konzoli pojavi izpis:

```
[user1@iarc ~]# sudo service iarcd start

WARNING: Network subsystem not running or (RT)NETLINK interface not configured in this kernel. If you're sure
that your network is UP you can ignore this message. Continue loading IMiS/ARChive HSM Storage Server...

Starting IMiS/ARChive HSM Storage Server:                [  OK  ]
[user1@iarc ~]#
```

Strežnik ni omrežno dostopen:

```
[user1@test ~]# ping iarc.acme.com
PING iarc.acme.com (10.1.1.10) 56(84) bytes of data.
... (premor) ...
From 192.168.92.32 icmp_seq=2 Destination Host Unreachable
From 192.168.92.32 icmp_seq=3 Destination Host Unreachable
From 192.168.92.32 icmp_seq=4 Destination Host Unreachable
... (prekinemo test s CTRL-C) ...
^C
--- iarc.acme.com ping statistics ---
7 packets transmitted, 0 received, +3 errors, 100% packet loss, time 6937ms
[user1@test ~]#
```

Vzrok težave 4

Ob zagonu strežnika omrežni podsistem operacijskega sistema ni deloval.

Rešitev težave 4

Potrebno je vzpostaviti delovanje omrežnega podsistema in nato ponovno zagnati strežnik.

Če se sporočilo pojavi ponovno, gre verjetno za nezdružljivost strežnika z operacijskim sistemom.

Opis pogoste težave 5

Ob zagonu strežnika IMiS®/ARChive Server se na konzoli pojavi izpis:

```
[user1@iarc ~]# sudo service iarcd start

WARNING: Network subsystem not running or (RT)NETLINK interface not configured in this kernel. If you're sure
that your network is UP you can ignore this message. Continue loading IMiS/ARChive HSM Storage Server...

Starting IMiS/ARChive HSM Storage Server:                [  OK  ]
[user1@iarc ~]#
```

Strežnik ni omrežno dostopen:

```
[user1@test ~]# ping iarc.acme.com
PING iarc.acme.com (10.1.1.10) 56(84) bytes of data.
... (premor) ...
From 192.168.92.32 icmp_seq=2 Destination Host Unreachable
From 192.168.92.32 icmp_seq=3 Destination Host Unreachable
From 192.168.92.32 icmp_seq=4 Destination Host Unreachable
... (prekinemo test s CTRL-C) ...
^C

--- iarc.acme.com ping statistics ---
7 packets transmitted, 0 received, +3 errors, 100% packet loss, time 6937ms
[user1@test ~]#
```

V dnevniku se v sosledju pojavijo zapisi:

```
<datum in ura zapisa> [iarcd:<decimalna vrednost>:<decimalna vrednost>] INFO[6] Preforking 1 connection handling childs.
<datum in ura zapisa> [iarcd:<decimalna vrednost>:<decimalna vrednost>] WARN[4] Cannot bind socket 0 to address [10.1.1.10] on port [16807], error 99: Cannot assign requested address. Socket will be closed.
<datum in ura zapisa> [iarcd:<decimalna vrednost>:<decimalna vrednost>] ERR[3] Server was unable to open any configured listening socket.
<datum in ura zapisa> [iarcd:<decimalna vrednost>:<decimalna vrednost>]  INFO[6] Child 2922 exited with exit code 0.
<datum in ura zapisa> [iarcd:<decimalna vrednost>:<decimalna vrednost>]  INFO[6] Fatal error occured. Server is shutting down.
```

Vzrok težave 5

Ob zagonu strežnika omrežni podsistem operacijskega sistema ne deluje ali pa so v nastavitveni datoteki napačne mrežne nastavitve.

Rešitev težave 5

Potrebno je preveriti delovanje omrežnega podsistema in ustrezno urediti omrežne nastavitve v nastavitveni datoteki /etc/iarc.conf.

Opis pogoste težave 6

Ob zagonu strežnika IMiS®/ARChive Server se na konzoli pojavi sporočilo:

```
[user1@iarc ~]# sudo service iarcd start
Error accessing IMiS/ARChive Database directory (<pot-do-baze>). Check user iarc access to this directory (must be rwx)
[user1@iarc ~]#
```

Strežnik je sicer omrežno dostopen:

```
[user1@test ~]# ping iarc.acme.com
PING iarc.acme.com (10.1.1.10) 56(84) bytes of data.
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=1 ttl=64 time=0.653 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=2 ttl=64 time=0.190 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=3 ttl=64 time=0.186 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=4 ttl=64 time=0.183 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=5 ttl=64 time=0.164 ms

--- iarc.acme.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 0.164/0.450/1.530/0.540 ms
[user1@test ~]#
```

Vzrok težave 6

Izvajalni program strežnika ne more dostopati do svoje interne baze podatkov zaradi:

- napačne nastavitve v nastavitveni datoteki /etc/iadbprovider.xml, sekcija /IMiSARChive/Configuration/Database/DriverArguments/DatabaseLocation in/ali napačno nastavljenih dostopnih pravic in/ali lastništva nad nastavljenim imenikom;
- interna baza ni dosegljiva, ker disk, na katerem se interna baza strežnika nahaja, ni povezan na pravi imenik.

Rešitev težave 6

Preveriti je potrebno pravilnost nastavitve lokacije interne baze podatkov strežnika, določene v /etc/iadbprovider.xml,

Sekcija /IMiSARChive/Configuration/Database/DriverArguments/DatabaseLocation, če ta obstaja.

Privzeta vrednost v primeru, da te nastavitve ni v nastavitveni datoteki,

je /iarc/db. Preveriti je potrebno pravice in lastništva imenika in datotek znotraj imenika, ki je naveden kot imenik, ki vsebuje datoteke interne baze podatkov. Uporabnik, ki izvaja procese strežnika (privzeto iarc), mora imeti pravice branja, pisanja in ustvarjanja novih datotek v tem imeniku. Skupini, kateri pripada uporabnik, ki izvaja procese strežnika (privzeto iarc) zadošča pravica branja.

V kolikor je imenik /iarc prazen, je najverjetnejši vzrok nedosegljivost diska, na katerem se po privzetih nastavitvah v imeniku /iarc/db sicer nahaja interna baza strežnika in je potrebno najprej zagotoviti dosegljivost tega diska.

Opis pogoste težave 7

Po ponovnem zagonu (angl. Restart) celotnega strežnika IMiS®/ARChive Server se na odjemalcih IMiS®/Scan in IMiS®/View ob poizkusu pregledovanja objektov shranjenih na strežniku pojavi »Napaka 61523«. Drugi odjemalci (npr. IMiS®/Storage Connector) javljajo napako:

IMiS/ARC Client <IASession.Open> Failed to establish connection to the cluster node <10.1.1.10, 16807> (Reason: Error <TimedOut> occurred while opening network connection.).

Strežnik je sicer omrežno dostopen.

```
[user1@test ~]# ping iarc.acme.com
PING iarc.acme.com (10.1.1.10) 56(84) bytes of data.
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=1 ttl=64 time=0.653 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=2 ttl=64 time=0.190 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=3 ttl=64 time=0.186 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=4 ttl=64 time=0.183 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=5 ttl=64 time=0.164 ms

--- iarc.acme.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 0.164/0.450/1.530/0.540 ms

[user1@test ~]#
```

Strežnik sicer deluje, status strežnika kot storitve pa javlja naslednje:

```
[user1@iarc ~]# sudo service iarcd status
Status of IMiS/ARChive HSM Storage Server: iarcd is stopped
[user1@iarc ~]#
```

Vzrok težave 7

Zagonska skripta strežnika ni aktivirana v sekvenci za zagon strežniških servisov.

Rešitev težave 7

Urediti je potrebno samodejni zagon strežnika kot storitve ob zagonu operacijskega sistema:

```
[user1@iarc ~]# sudo chkconfig iarcd on
[user1@iarc ~]#
```

Preverimo uspešnost izvedbe ukaza:

```
[user1@iarc ~]# sudo chkconfig iarcd --list
iarcd           0:off   1:off   2:on    3:on    4:on    5:on    6:off [user1@iarc ~]#
```

Nato strežnik zaženemo z ukazom:

```
[user1@iarc ~]# sudo service iarcd start
Starting IMiS/ARChive HSM Storage Server:                [  OK  ]
[user1@iarc ~]#
```

Opis pogoste težave 8

Ob zagonu strežnika IMiS®/ARChive Server se na konzoli pojavi izpis:

```
[user1@iarc ~]# sudo service iarcd start
Starting IMiS/ARChive HSM Storage Server:
WARNING: Maximum number of file handles (ulimit -n) allowed for
user iarc or group iarc is 1024. Set allowable maximum to
at least 4096 by adding following two lines to /etc/security/limits.conf:
iarc          hard          nofile        4096
iarc          soft          nofile        4096
 or
@iarc         hard          nofile        4096
@iarc         soft          nofile        4096
If you still recieve this message after modifying /etc/security/limits.conf
check if Pluggable Authentication Modules (PAM) include module
pam_limits.so in session service for user iarc and/or group iarc
(see Linux-PAM system administrators guide on how to manage modules)
IMiS/ARChive will continue to run normally with current setting...
                                         [  OK  ]
[user1@iarc ~]#
```

Storitev po zagonu sicer normalno deluje. Čez čas postane strežnik nedosegljiv za seje novih odjemalcev. V dnevniku se pojavijo zapisi:

```
<datum in ura zapisa> [iarcd:<decimalna vrednost>:<decimalna vrednost>] CRIT[2] No child process can accept
new connection.
```

Vzrok težave 8

Strežnik je dosegel največje možno število odprtih datotek in zato ne more več sprejemati novih povezav. Vsako povezavo namreč operacijski sistem zazna kot »odprto datoteko«.

Rešitev težave 8

Potrebno je preveriti sistemsko nastavitev največjega možnega števila odprtih datotek za uporabnika `iarc`, pod katerim strežnik teče.

9.3 Redkejše težave

Opis redke težave 1

Ob poizkusu pregledovanja objekta shranjenega na strežniku IMiS®/ARChive Server se na odjemalcu IMiS®/View ali IMiS®/Scan pojavi »Napaka 11«.
Drugi odjemalci (npr. IMiS®/Storage Connector) pri poizkusu pridobivanja objekta s strežnika dobijo naslednji odgovor:

```
iavol OpenObject error 0x564e4f56..
```

Strežnik je sicer omrežno dostopen, servis na vratih, kjer posluša, se odziva (preverjanje s telnet programom)

```
[user1@test ~]# ping iarc.acme.com
PING iarc.acme.com (10.1.1.10) 56(84) bytes of data.
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=1 ttl=64 time=0.653 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=2 ttl=64 time=0.190 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=3 ttl=64 time=0.186 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=4 ttl=64 time=0.183 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=5 ttl=64 time=0.164 ms

--- iarc.acme.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 0.164/0.450/1.530/0.540 ms


[user1@test ~]# telnet iarc.acme.com 16807
Trying 10.1.1.10...
Connected to iarc.acme.com.
Escape character is '^]'.
Connection closed by foreign host.
[user1@test ~]#
```

Strežnik deluje, kar dodatno preverimo s konzolnim ukazom na strežniku,
kjer je nameščen:

```
[user1@iarc ~]# sudo service iarcd status
Status of IMiS/ARChive HSM Storage Server: iarcd (pid 23209 23203) is running...
[user1@iarc ~]#
```

Vzrok težave 1

Odjemalec skuša odpreti objekt, ki je pravilno vpisan v interni bazi strežnika,
vendar vsebina objekta ni na svojem mestu ali manjka.

Rešitev težave 1

Potrebno je pridobiti:

- podatke o identifikatorju objekta iz aplikacije, ki arhivski sistem uporablja (npr.: 4c9f36d38b4d6985b1ec111a5a14a7e9db89edd0cb36923010b6624c667ef142);
- vsebino parametra IdentPassword iz nastavitvene datoteke strežnika /etc/iarc.conf;
- podatke o kupcu.

Vse skupaj je potrebno poslati po e-pošti na naslov: podpora@imis.si.

Naše tehnično osebje nato dešifrira identifikator objekta, ki je osnova za informacijo in nadaljnje postopke restavracije iz varnostnih kopij ali iskanje po datotečnem sistemu, v kolikor se ne nahaja na izvornem mestu. To je možno samo v primeru, ko ga je nekdo s pravicami upravljalca strežnika premaknil ali izbrisal iz izvornega mesta.

Opis redke težave 2

Ob poizkusu pregledovanja objekta shranjenega na strežniku IMiS®/ARChive Server se na odjemalcih IMiS®/Scan ali IMiS®/View pojavi »Napaka pri branju IMiS objekta«. Drugi odjemalci (npr. IMiS®/Storage Connector) pri poizkusu pridobivanja objekta s strežnika dobijo naslednji odgovor:

```
Unable to locate database record for entity <decimalna vrednost>
```

Strežnik je sicer omrežno dostopen, servis na vratih, kjer posluša, se odziva (preverjanje s telnet programom)

```
[user1@test ~]# ping iarc.acme.com
PING iarc.acme.com (10.1.1.10) 56(84) bytes of data.
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=1 ttl=64 time=0.653 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=2 ttl=64 time=0.190 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=3 ttl=64 time=0.186 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=4 ttl=64 time=0.183 ms
64 bytes from iarc.acme.com (10.1.1.10): icmp_seq=5 ttl=64 time=0.164 ms
```

```
--- iarc.acme.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 0.164/0.450/1.530/0.540 ms

[user1@test ~]# telnet iarc.acme.com 16807
Trying 10.1.1.10...
Connected to iarc.acme.com.
Escape character is '^]'.
Connection closed by foreign host.
[user1@test ~]#
```

Strežnik deluje, kar dodatno preverimo s konzolnim ukazom na strežniku, kjer je nameščen:

```
[user1@iarc ~]# sudo service iarcd status
Status of IMiS/ARChive HSM Storage Server: iarcd (pid 23209 23203) is running...
[user1@iarc ~]#
```

Vzrok težave 2

Odjemalec skuša odpreti objekt, ki ni vpisan v interni bazi strežnika.

Rešitev težave 2

Potrebno je pridobiti:

- podatke o identifikatorju objekta iz aplikacije, ki arhivski sistem uporablja (npr.: 4c9f36d38b4d6985b1ec111a5a14a7e9db89edd0cb36923010b6624c667ef142);
- vsebino parametra IdentPassword iz nastavitvene datoteke strežnika /etc/iarc.conf;
- interno podatkovno bazo (vsebino imenika /iarc/db) v stisnjeni obliki ali njeno tekstualno obliko;
- podatke o kupcu.

Vse skupaj je potrebno poslati po e-pošti na naslov: podpora@imis.si.

Naše tehnično osebje nato dešifrira identifikator objekta, ki je osnova za informacijo in nadaljnje postopke ugotavljanja stanja interne podatkovne baze in vzroka za izpad zapisa, ki je osnova za pravilne operacije z objektom v inventarju strežnika.

9.4 Seznam napak storitve, ki se beležijo v dnevnik delovanja

9.4.1 Nivo 0 – Emergency

EMERG: »Unknown exception caught.«
EMERG: Exception caught <opis>
Napaka se zgodi v primeru, ko pride do težke napake pri delovanju strežnika IMiS®/ARChive Server, ki pa ni evidentirana oziroma predvidena kot možna napaka.
Razlogi so različni: od stanja okolja do morebitnih napak v aplikacijski kodi storitve.

9.4.2 Nivo 1 – Alert

ALERT: »Out of memory.«
Napaka pomeni, da je strežniku IMiS®/ARChive Server zmanjkalo razpoložljivega delovnega spomina. Kljub temu strežnik pri obstoječem stanju lahko nadaljuje z delom, čeprav je stanje kritično. V primeru daljše izpostavljenosti takem okolju lahko storitev preneha delovati.

ALERT: »Thread <identifikator niti> error number <koda napake>. Exiting...«
Napaka se pojavi, ko pri delovanju ene izmed niti pride do nepopravljive napake.
Odvisno od teže napake strežnik IMiS®/ARChive Serverv tem primeru prekine delovanje niti ali celotnega izvajalnega programa, saj bi nadaljevanje lahko ogrozilo konsistenco persistentnih podatkov.

ALERT: »Maximum number of child processes reached.«
Obvestilo pomeni, da strežnik IMiS®/ARChive Server ne more več zagnati novega povezovalnega pod-procesa in zato ne more obdelati novih zahtevkov.
V tem primeru strežnik teče naprej, dokler se viri ne sprostijo ali pa povečamo število pod-procesov (nastavitev v `/etc/iarc.conf` datoteki).

ALERT: »Shared memory (hnd = <št.>, ptr = <št.>) error <koda napake>«
Obvestilo o napaki se pojavi v primeru, da je strežnik IMiS®/ARChive Server zaznal napako ali nepravilnost pri delu z delom spomina v souporabi, ki se uporablja za komunikacijo med procesi. Delovanje strežnika se takoj zaustavi zaradi nezmožnosti komunikacije med procesi družine IMiS®/ARChive storitve.

9.4.3 Nivo 2 – Critical

CRIT: »Out of memory. Cannot continue.«
Napaka pomeni, da je strežniku IMiS®/ARChive Server zmanjkalo razpoložljivega delovnega spomina. Kljub temu, da je stanje kritično, lahko pri obstoječemu stanju nadaljuje z delom.
V primeru daljše izpostavljenosti takem okolju, lahko storitev preneha delovati.

CRIT: »Error is unrecoverable. The process will terminate.«
Prišlo je do napake zaradi katere proces ne more nadaljevati z delom.
To sporočilo je ponavadi le posledica neke druge napake, ki se je zgodila tik pred tem in je tudi zabeležena v dnevniku napak.

CRIT: »Unsupported client address structure at <št. procesa>.«
Strežnik IMiS®/ARChive Server je zaznal nepodprt tip naslova odjemalca na procesu <št. procesa>. Napaka pomeni prekinitev komunikacije odjemalca s strežnikom.

CRIT: »Signal SIGSEGV occured. Process will shut down ...«
Pri delu z delovnim spominom strežnika IMiS®/ARChive Server je prišlo napake. Strežnik takoj prekine z delom. Potreben je poseg administratorja strežnika, navadno ponoven zagon storitve.

CRIT: »No child process can accept new connection.«
Komunikacijski pod-procesi ne morejo več sprejemati novih zahtevkov.
Napaka je ponavadi posledica druge napake ali pomanjkanja virov strežnika.

CRIT: »Cannot initialize LC_COLLATE setting.«
CRIT: »Cannot initialize LC_CTYPE setting.«
Strežnik IMiS®/ARChive Server ne more nastaviti želenih regionalnih nastavitev za razvrščanje in prevajanje znakov, ki presegajo obseg ACSII kodne tabele.
V kolikor je nastavitev ena izmed ponujenih, ko izvedemo ukaz locale –a,
je napaka zgolj teoretična.
CRIT: »New process couldn't accept new connection.«
Strežniku IMiS®/ARChive Server je zmanjkalo virov. Zagnal je nov komunikacijski proces, a ni dovolj virov, da bi sprejemal nove zahtevke.

CRIT: »Error <koda napake> while recording session close to audit log.«
CRIT: »Error <koda napake> while recording session open to audit log.«
Pri beleženju v dnevnik nadzora je prišlo do napake, zapis ni bil uspešno posredovan in zapisan v interno bazo podatkov. Razlogi so različni. Potreben je poseg administratorja.

CRIT: »Locale parameter '<vrednost parametra>' for LC_COLLATE in not formated according to POSIX standard and cannot be used. Use format ll_CC[.CHARSET[@variant]].«
CRIT: »Locale parameter '<vrednost parametra>' for LC_CTYPE in not formated according to POSIX standard and cannot be used. Use format ll_CC[.CHARSET[@variant]].«
Podan parameter regionalnih nastavitev za sortiranje in prevajanje znakov, ki presegajo obseg ACSII kodne tabele, ni pravilnega POSIX formata.
Glej nastavitev CountryLanguage v sekciji [Server] nastavitvene datoteke.

CRIT: »Error <koda napake> while recording server's session to audit log.«
Napaka pri vnosu strežniške seje v revizijsko sled sej interne baze podatkov.
Razlogi so različni. Potreben je poseg administratorja. Napaka je sicer zgolj teoretična.

CRIT: »Error <koda napake> opening file <ime datoteke>. Terminating process.«
CRIT: »Error <koda napake> while reading file <ime datoteke>. Terminating process.«
CRIT: »File <ime datoteke> is too small (<število> bytes). Terminating process.«
CRIT: »Invalid header data in file <ime datoteke>. Terminating process."CRIT: "Error <koda napake> seeking in file <ime datoteke>. Terminating process.«
CRIT: »File <ime datoteke> is too large. Terminating process."CRIT: "File <ime datoteke> has invalid size (<število> bytes). Terminating process.«
CRIT: »Error <koda napake> while reading file <ime datoteke>. Terminating process.«
CRIT: »Audit log config check: Crypt engine error. Terminating process.«
CRIT: »Audit log config check: Data size doesn't match. Terminating process.«
CRIT: »Audit log config check: Checksum error. Terminating process.«
CRIT:«Audit log config check: Crypt engine error. Terminating process."CRIT: "Error <koda napake> while writing to file <ime datoteke>. Terminating process.«
CRIT: »Audit log config check: Database error. Terminating process."CRIT: "Audit log config check: Unexpected unknown error. Terminating process.«
Zgornje napake vse nakazujejo na težave pri sistemu za detekcijo sprememb nastavitev revizijske sledi. Napako lahko odpravimo z odstranitvijo datoteke /iarc/db/iaalcc.bin, vendar je napako potrebno obravnavati resno in raziskati njen vzrok. Odstranitev omenjene datoteke bo ob ponovnem zagonu strežnika povzročila vnos »sprememb« nastavitev revizijske sledi, čeprav do njih ni prišlo.
Trenutne nastavitve bodo označene kot spremenjene, saj storitev nima na voljo stanja nastavitev pred zagonom (te so shranjene v šifrirani binarni obliki v omenjeni datoteki).

CRIT: »Error while recording server's session to audit log.«
CRIT: »Malformed Entity identification configuration. Reason: '<razlog>'«

9.4.4 Nivo 3 – Error

ERR: »Read returned with error: <koda napake>.«
Pri branju z vtiča za komunikacijo z odjemalcem je prišlo do napake. Napaka ni kritična in ponavadi pomeni nenadno, neregularno prekinitev seje s strani odjemalca.

ERR: »Write returned with error: <koda napake>.«
Pri pisanju na vtič za komunikacijo z odjemalcem je prišlo do napake. Napaka ni kritična in ponavadi pomeni nenadno, neregularno prekinitev seje s strani odjemalca.

ERR: »No select file descriptor available.«
Doseženo je največje možno število odprtih datotek. Zato ni mogoče uporabiti novega vozla (i-node), ki ga strežnik IMiS®/ARChive Server potrebuje za delo z objektom.

ERR: »IDFromIdentShort: Unknown ObjectID version information.«
Odjemalec je zahteval objekt, katerega struktura identifikatorja strežniku IMiS®/ARChive Server ni znana, ga ne more dešifrirati ali ga nima evidentiranega v svoji interni bazi. Zahtevek odjemalca strežnik zavrže.

ERR: »Cannot create object file <koda/ime datoteke>.«
Pri zapisovanju objekta na volumen je prišlo do napake, ker datoteka z objektom že obstaja. Potreben je poseg strokovnjaka proizvajalca in razrešitev vprašanja, zakaj na tem mestu datoteka z identifikatorjem objekta že obstaja.

ERR: »Unknown file handling error.«
Pri delu z objektom je prišlo do nepredvidene napake. Zahtevek strežnik IMiS®/ARChive Server zavrže.

ERR: »Cannot open object file <ime datoteke objekta>.«
Odjemalec je od strežnika IMiS®/ARChive Server zahteval objekt, ki je sicer vpisan v inventarju, vendar pa objektna datoteka ne obstaja.

ERR: »ObjRemove error <koda napake>.«
Strežnik IMiS®/ARChive Serverje prejel regularen zahtevek za brisanje objekta, vendar brisanje ni izvedljivo. Potreben je poseg strokovnjaka proizvajalca za razrešitev vprašanja, zakaj postopek izbrisa ni možen (navadno vzrok zaradi pravic nad datotečnim sistemom in datotekami HSM inventarja).

ERR: »Cannot open object file <ime objektne datoteke>.«
Pri poizkusu branja objekta je prišlo do napake. IMiS®/ARChive Server objekta ne more odpreti. Potreben je poseg strokovnjaka proizvajalca, saj gre za objekt, ki je sicer vpisan v inventar objektov vendar ostaja vprašanje, zakaj postopek ni možen (navadno vzrok zaradi pravic nad datotečnim sistemom in datotekami HSM inventarja)

ERR: »Not enough space available in profile <id profila>.«
Na volumnih, ki pripadajo profilu <id profila> je zmanjkalo prostora.

ERR: »Invalid profile number.«
V zahtevku odjemalca je bila uporabljena napačna ali neobstoječa številka profila.

ERR: »Unexpected FIN!«
Odjemalec IMiS® je nepričakovano zaključil sejo ali pa je poslal signal za zaključek seje po tem, ko je zaradi neaktivnosti sejo zaključil že strežnik IMiS®/ARChive Server sam.

ERR: »Error in ConnInfoGetLib request (req->seq). Skipping processing.«
Odjemalec IMiS® je strežniku IMiS®/ARChive Server poslal nepravilen zahtevek za komunikacijsko knjižnico.

ERR: »Cannot open file <ime knjižnice>.«
Odjemalec IMiS® je strežniku IMiS®/ARChive Server poslal pravilen zahtevek za komunikacijsko knjižnico, vendar te ni na svojem mestu. Napaka navadno pomeni nepopolno namestitev, napačno nastavitev v nastavitveni datoteki /etc/iarc.conf
ali problem pravic za uporabnika iarc.

ERR: »Unknown object handle <koda/ročica>.«
Operacijski sistem je strežniku IMiS®/ARChive Server posredoval neregularno ročico datoteke objekta. Napaka je najverjetneje posledica napačnega delovanja operacijskega sistema ali datotečnega sistema.

ERR: »Unknown transmission handle.«
Operacijski sistem je strežniku IMiS®/ARChive Server posredoval neregularno ročico vozla (i-node) povezave. Napaka je najverjetneje posledica napačnega delovanja operacijskega sistema.

ERR: »Unknown ConnInfo request: <koda> - ignoring!«
Strežnik IMiS®/ARChive Server je prejel neregularen zahtevek za podatke o povezavi z odjemalcem. Zahtevek strežnik zavrže.

ERR: »Unknown External ID request size (velikost zahtevka).«
Strežnik IMiS®/ARChive Server je prejel zahtevek za identifikacijsko številko za nov objekt za t.i. zunanji sistem (npr. SAP/R3). Velikost zahtevane identifikacijske številke pa ni regularna. Zahtevek strežnik zavrže kot neveljaven.

ERR: »Invalid request size: <velikost zahtevka>.«
Strežnik IMiS®/ARChive Server je prejel zahtevek, katerega velikost ni regularna. Zahtevek strežnik zavrže kot neveljaven.

ERR: »Unknown request <koda zahtevka> received. Closing connection.«
Strežnik IMiS®/ARChive Server je prejel neregularen zahtevek in zaprl odprto povezavo. Največkrat je to posledica poizkusa vzpostavitve povezave preko TCP vrat strežnika s strežniku nepoznanim protokolom.

ERR: »Socket <koda vtiča> closed for reading on client side. Connection closed.«
Strežnik IMiS®/ARChive Server je zaznal, da je bil vtič za komunikacijo z odjemalcem zaprt, zato je tudi sam povezavo na svoji strani zaprl.

ERR: »Socket <koda vtiča> write error <koda napake>.«
Strežnik IMiS®/ARChive Server preko vtiča ne more komunicirati z odjemalcem.

ERR: »Error reading message queue (errno: <koda napake>).«
Pri izmenjavi podatkov med procesi družine IMiS®/ARChive Server storitve je pri branju podatkov iz čakalne vrste za med-procesno komunikacijo prišlo do napake.

ERR: »Ident(): Initializing crypto engine.«
Napaka pri inicializaciji sistema šifriranja identifikatorjev objektov.
Napaka je zgolj teoretična in posledica napake programerja, potreben je poseg strokovnjaka proizvajalca.

ERR: »Ident(): Setting internal key.«
Napaka pri sestavljanju ključa za prvo stopnjo šifriranja identifikatorjev objektov.
Napaka je zgolj teoretična in posledica napake programerja, potreben je poseg strokovnjaka proizvajalca.

ERR: »Ident(): Setting external key.«
Napaka pri sestavljanju ključa za drugo stopnjo šifriranja identifikatorjev objektov.
Napaka je zgolj teoretična in je posledica napake programerja, potreben je poseg strokovnjaka proizvajalca.

ERR: »Ident(): Internal encrypting.«
Napaka pri prvi stopnji šifriranja identifikatorja objekta. Napaka je zgolj teoretična in je posledica napake programerja, potreben je poseg strokovnjaka proizvajalca.

ERR: »Ident(): External encrypting.«
Napaka pri drugi stopnji šifriranja identifikatorja objekta. Napaka je zgolj teoretična in je posledica napake programerja, potreben je poseg strokovnjaka proizvajalca.

ERR: »IDFromIdent(): Initializing crypto engine.«
Napaka pri inicializaciji sistema dešifrira identifikatorjev objektov. Napaka je zgolj teoretična in je posledica napake programerja, potreben je poseg strokovnjaka proizvajalca.

ERR: »IDFromIdent(): Setting external key.«
Napaka pri sestavljanju ključa za prvo stopnjo dešifriranja identifikatorjev objektov.
Napaka je zgolj teoretična in je posledica napake programerja, potreben je poseg strokovnjaka proizvajalca.

ERR: »IDFromIdent(): Setting internal key.«
Napaka pri sestavljanju ključa za drugo stopnjo dešifriranja identifikatorjev objektov.
Napaka je zgolj teoretična in je posledica napake programerja. Potreben je poseg strokovnjaka proizvajalca.

ERR: »IDFromIdent(): External decrypting.«
Prišlo je do napake pri prvi stopnji dešifriranja identifikatorja objekta. Do napake lahko pride zaradi nepravilnega identifikatorja poslanega s strani odjemalca.

ERR: »IDFromIdent(): Internal decrypting.«
Prišlo je do napake pri drugi stopnji dešifriranja identifikatorja objekta. Do napake lahko pride zaradi nepravilnega identifikatorja poslanega s strani odjemalca.

ERR: »IdentShort(): error <koda napake>while cyphering internal block.«
Pri prvi stopnji šifriranja kratkega identifikatorja objekta je prišlo do napake.
Napaka je zgolj teoretična in zahteva pregled strokovnjaka proizvajalca.

ERR: »IdentShort(): error <koda napake>while cyphering external block.«
Pri drugi stopnji šifriranja kratkega identifikatorja objekta je prišlo do napake.
Napaka je zgolj teoretična in zahteva pregled strokovnjaka proizvajalca.

ERR: »IDFromIdentShort: 1st Server id (<identifikator strežnika>) does not match.«
Odjemalec je posredoval nepravilno vrednost za kratko obliko identifikatorja objekta.

ERR: »IDFromIdentShort: 2nd Server id (<identifikator strežnika>) does not match.«
Odjemalec je posredoval nepravilno vrednost za kratko obliko identifikatorja objekta.

ERR: »IDFromIdentShort: Unknown ObjectID version information (<decimalna vrednost>).«
Odjemalec je posredoval nepravilno vrednost za kratko obliko identifikatorja objekta, ali pa je vrednost nastala z novejšo različico strežnika IMiS®/ARChive Server in je strežnik ne zna dešifrirati.

ERR: »IDFromIdentShort(): error <koda napake> while decyphering external block.«
ERR: »IDFromBytesL(): error <koda napake> while decyphering external block.«
Prišlo je do napake pri prvi stopnji dešifriranja kratke oblike identifikatorja objekta.
Do napake pride zaradi nepravilnega identifikatorja kratke oblike poslanega s strani odjemalca.

ERR: »IDFromIdentShort(): error <koda napake> while decyphering internal block.«
Prišlo je do napake pri drugi stopnji dešifriranja kratke oblike identifikatorja objekta.
Do napake pride zaradi nepravilnega identifikatorja kratke oblike poslanega s strani odjemalca.

ERR: »IDFromIdentShort: Invalid ID data.«
Napaka zaradi napačne vsebine kontrolnih podatkov po dešifriranju kratke oblike identifikatorja objekta. Do napake pride zaradi nepravilnega identifikatorja kratke oblike poslanega s strani odjemalca.

ERR: »Volume client error.«
Prišlo je do neidentificirane napake pri delovanju modula za upravljanje z diskovnimi mediji.
Vzrok za napako je navadno nepravilno delovanje operacijskega sistema zaradi pomanjkanja sistemskih sredstev.

ERR: »Invalid audit query size <decimalna vrednost>«
Strežnik je prejel zahtevek za iskanje po revizijski sledi, katerega velikost je nepravilna.
Vzrok za napako je nepravilno delovanje odjemalca, napaka je sicer zgolj teoretična.

ERR: »Invalid sess_cond.type (<decimalna vrednost>)«
Strežnik je prejel zahtevek za iskanje po revizijski sledi, v katerem je nepravilna vrednost za določanje kriterija iskanja sej. Vzrok za napako je nepravilno delovanje odjemalca, napaka je sicer zgolj teoretična.

ERR: »Invalid sess_cond.offset (<decimalna vrednost>)«
Strežnik je prejel zahtevek za iskanje po revizijski sledi, v katerem je nepravilna struktura kriterija za iskanje sej. Vzrok za napako je nepravilno delovanje odjemalca, napaka je sicer zgolj teoretična.

ERR: »Invalid ts_cond.offset (<decimalna vrednost>)«
Strežnik je prejel zahtevek za iskanje po revizijski sledi, v katerem je nepravilna struktura kriterija za časovno obdobje dogodkov. Vzrok za napako je nepravilno delovanje odjemalca, napaka je sicer zgolj teoretična.

ERR: »Invalid objid_cond.offset (<decimalna vrednost>)«
Strežnik je prejel zahtevek za iskanje po revizijski sledi, v katerem je nepravilna struktura kriterija za določanje identifikatorjev objektov. Vzrok za napako je nepravilno delovanje odjemalca, napaka je sicer zgolj teoretična.

ERR: »AuditQuery::GetNextAddress(), line <decimalna vrednost>, error <decimalna vrednost>«
Strežnik je prejel zahtevek za iskanje po revizijski sledi, v katerem kriterij za iskanje po omrežnih naslovih odjemalcev vsebuje podatke nepravilne oblike. Vzrok za napako je nepravilno delovanje odjemalca, napaka je sicer zgolj teoretična.

ERR: »ObjectsQueryArray::FindEvents(); Error decrypting object id.«
Strežnik je prejel zahtevek za iskanje po revizijski sledi, ki vsebuje identifikator objekta nepravilne vrednosti. Vzrok za napako je nepravilno delovanje odjemalca, napaka je sicer zgolj teoretična.

ERR: »msgctl(<sistemski identifikator>, IPC_RMID) error <koda napake>: <opis sistemske napake>«
Prišlo je do sistemske napake pri vzpostavitvi čakalne vrste za komunikacijo med procesi, konkretno pri poizkusu odstranitve obstoječe čakalne vrste. Vzrok za napako je navadno nepravilno delovanje operacijskega sistema in je bolj natančno opisan z opisom v <opis sistemske napake>.

ERR: »Could not connect to iavol server. Session canceled.«
Prišlo je to napake pri povezovanju z modulom za delo z diskovnimi mediji.
Vzrok za napako je navadno pomanjkanje sistemski sredstev, oziroma nezmožnost operacijskega sistema za serviranje doseženega števila sej odjemalcev.

ERR: »BuildVolTree(): Volume <identifikacija volumna> not mounted.«
Prišlo je do napake pri uporabi določenega volumna. Vzrok za neuporabnost je ena izmed napak, ki je bila predhodno zabeležena v dnevnik.

ERR: »Error closing file descriptor.«
Prišlo je do sistemske napake pri zapiranju datoteke. Napaka je navadno posledica napake programerja, vzrok zanjo pa je lahko tudi nepravilno delovanje operacijskega sistema.

ERR: »accept() returned error <koda napake>.«
ERR: »accept() error <koda napake>: <opis sistemske napake>.«
Napaka pri vzpostavljanju nove seje z odjemalcem. Vzrok za napako je najverjetneje preveliko število trenutno vzpostavljenih sej z odjemalci, pri drugi obliki pa je natančnejši vzrok opisan z <opis sistemske napake>.

ERR: »Passed fd <decimalna vrednost> is not a listen socket.«
Napaka pri čakanju zahtevkov za vzpostavitev novih sej z odjemalci. Vzrok za napako je posledica napake v programu, je zgolj teoretična. Potreben je poseg strokovnjaka proizvajalca.

ERR: »Error creating thread: 0x<šestnajstiška vrednost>.«
Napaka pri vzpostavljanju nove procesne niti programa. Vzrok za napako je navadno pomanjkanje sistemskih sredstev.

ERR: »Stats counter error <koda napake>: "<opis sistemske napake>.«
Napaka pri shranjevanju statističnih podatkov o številu dostopov do objektov v določenem časovnem obdobju. Napaka je posledica nepravilnega delovanja operacijskega sistema.

ERR: »Profile(<šestnajstiška vrednost>): No volumes on level <številka nivoja>. Emergency migration skipped.«
ERR: »Profile(<šestnajstiška vrednost>): No volumes on level <številka nivoja>. Scheduled migration skipped.«
Napaka pri poizkusu migracije objektov profila na nivo višje (<številka nivoja>) v hierarhiji volumnov. Do napake je prišlo, ker profil na tem nivoju nima določenih volumnov, na volumnih nivoja nižje pa primanjkuje prostora.

ERR: »mkstemp("<ime datoteke>") error <koda napake>.«
Prišlo je do napake pri ustvarjanju začasne datoteke z imenom <ime datoteke>.
Napaka je navadno posledica pomanjkanja prostora na disku, pravic uporabnika iarc, lahko pa je tudi posledica nepravilnega delovanja operacijskega sistema.

ERR: »unlink("<ime datoteke>") error <koda napake>.«
Prišlo je do napake pri brisanju datoteke z imenom <ime datoteke>.
Napaka je lahko posledica napake v programu, verjetneje pa gre za napako pravic dostopa za uporabnika iarc ali nepravilno delovanje operacijskega sistema.

ERR: »User '<uporabnik>' at connection <število> denied due to authentication failure - Authentication request sequence mismatch (request subid:proto:stage = COPN_CLIAUTH:SRP6A:<število>, valid COPN_CLIAUTH:SRP6A:<število>, session stage = <število>);«
ERR: »Error creating server SRPC-6a evidence for user '<uporabnik>'. Connection <število> denied (reason: <razlog>).«
ERR: »Connection <število> denied due to authentication failure - Authentication request sequence mismatch (request subid:proto:stage = COPN_CLIAUTH:SRP6A:<število>, valid COPN_CLIAUTH:SRP6A:[<število>/<število>], session stage = <število>);«
ERR: »Connection <število> denied (reason: Unknown Authentication mode requested (id = <število>))«
Napaka pri avtentikaciji uporabnika. Razlogi so različni, od posredovanja napačnih poverilnic do (redko) napačne uporabe avtentikacijskega protokola. Dostop takšnim sejam je zavrnjen.

ERR: »Level of volume in profile exceeds maximal allowed value.«
Trenutni nivo volumna je presegel najvišjo dovoljeno vrednost (16).

ERR: »GetVolumeInfo error <opis napake>.«
Napaka pri odpiranju volumna storitve skladišča digitalnih vsebin.

ERR: »"Error connecting to database. Reason: '<opis napake>' at'<datoteka>:<vrstica>'«
Napaka pri povezovanju s podatkovno bazo. V primeru napake je potreben pregled strokovnjaka proizvajalca.

ERR: »Error opening archive.«
Napaka pri odpiranju arhiva. Najverjetnejši razlog je napaka v podatkih ali nekonsistenca nastavitev arhiva. V primeru napake je potreben pregled strokovnjaka proizvajalca.

ERR: »IAVolume: Unable to retrieve a database session!«
Težava pri pridobivanju podatkovne seje. V primeru napake je potreben pregled strokovnjaka proizvajalca.

ERR: »Database exception <opis napake>!«
Težava pri dostopu do podatkovne baze. Rešitev težave je odvisna od opisa napake.
V primeru, da rešitev ni razvidna iz opisa je potreben pregled strokovnjaka proizvajalca.

ERR: »Access mode '<način dostopa>' is unsupported.«
ERR: »Unsupported access mode '<način dostopa>'.«
V zahtevku je določen neveljaven način dostopa. Dovoljeni vrednosti sta: »RO« in »RW«.
Napaka na odjemalcu IMiS®/Client.

ERR: »ACL entry for '<imeniška entiteta>' is internal and cannot be updated.«
Prepovedano spreminjanje liste dostopnih pravic na sistemskih imeniških entitetah.

ERR: »Address '< omrežni naslov>' structure is not supported.«
Napačen format omrežnega naslova. Vpisati je potrebno ali veljavno ime gostitelja, veljaven IPv4 naslov ali veljaven IPv6 naslov z opcijskim dodatkom omrežnih vrat. Ločilo med omrežnim naslovom in omrežnimi vrati je znak »:«.
Napaka pri nastavitvi produkta.

ERR: »Attribute 'destination' containes the value '%s' which resolves to an unknown attribute«
ERR: »Attribute 'destination' identifies the attribute '%s' which type '%u' is not supported for entity identification storage«
Napaka pri nastavitvi števcev za samodejno številčenje entitet.

ERR: »AuditQuery::GetNextAddress(), line %d, error %d«
Napaka pri določanju IP naslova iz zahtevka za vpogled v revizijsko sled.

ERR: »Binary, File and StringMax are unsortable.«
Razvrščanje atributov tipa »Binary«, »File« in »StringMax« ni mogoče.
Napačna uporaba funkcionalnosti sortiranja na odjemalcu IMiS®/Client.

ERR: »Both IP addresses need to be of the same length and same protocol.«
Napaka pri določanju razpona IP naslovov iz zahtevka za vpogled v revizijsko sled.
Obe strani razpona morata biti iste družine. Napačno posredovana vrednost razpona iz odjemalca IMiS®/Client.

ERR: »Can't move an entity into its own subtree.«
Premik entitete v lastno podrejeno entiteto ni mogoč.

ERR: »Classification code '<klasifikacijska oznaka>' exceeds maximum length of 20 characters.«
Največja dovoljena dolžina relativne klasifikacijske oznake je 20 znakov.

ERR: »Collection handle '<šestnajstiška vrednost>' is invalid.«
Referenca (handle) na zbirko je neveljavna. Možni razlogi za napako so:

- zbirka je bila predhodno zaprta
- seja iz katere izvira referenca je bila zaprta in ponovno odprta nova seja
- referenca ni nikoli obstajala.

Napaka na odjemalcu IMiS®/Client

ERR: »Configuration requires client computer name to be provided.«
Strežnik IMiS®/ARChive Server je nastavljen tako, da pri odpiranju nove seje zahteva ime računalnika s katerega je bila vzpostavljena seja.
Napaka v nastavitvi, če ime računalnika ni obvezen podatek ali napaka odjemalca IMiS®/Client, če je ime računalnika obvezen podatek.

ERR: »Configuration requires client username to be provided.«
Strežnik IMiS®/ARChive Server je nastavljen tako, da pri odpiranju nove seje zahteva uporabniško ime s katerega je bila vzpostavljena seja v primeru uporabe avtentikacijskih metod pred SRP-6a.
Napaka v nastavitvi, če uporabniško ime ne bi smelo biti obvezen podatek ali napaka odjemalca, če je uporabniško ime obvezen podatek.

ERR: »Counter definition '<definicija števca >' does not contain a required variable '<ime spremenljivke>'.«
Napaka v določevanju števca. Ime spremenljivke ni bilo moč najti v določitvi števca za avtomatično določanje klasifikacijskih oznak.

ERR: »Decrypted Authentication payload starting sequence is invalid.«
Napaka pri avtentikaciji. Napaka na odjemalcu IMiS®/Client.

ERR: »Destination is already a parent of the specified entity. Move is not possible.«
Premik ne spremeni umestitve entitete v načrtu razvrščanja gradiva in posledično ni smiseln. Premik se ne izvede. Napaka na odjemalcu IMiS®/Client.

ERR: »Element count must be greater than 0.«
Število elementov na strani zbirke entitet mora biti pozitivno število.
Napaka na odjemalcu IMiS®/Client.

ERR: »Empty request recieved.«
Prejet je bil zahtevek brez vsebine. Napaka na odjemalcu IMiS®/Client.

ERR: »Encoding (<število>) not recognized.«
Zahtevano šifriranje (encoding) za enolične oznake entitet v rezultatu poizvedbe revizijske sledi ni bilo prepoznano. Dovoljene vrednosti so 1 (BASE 16), 2 (BASE 64) in 3 (BASE 85).

ERR: »Entity handle '<šestnajstiško število>is invalid.«
ERR: »Entity handle is invalid.«
Referenca (handle) na entiteto je neveljavna. Možni razlogi za napako so:

- entiteta je bila predhodno zaprta
- seja iz katere izvira referenca je bila zaprta in ponovno odprta nova seja.
- Referenca ni nikoli obstajala.

Napaka na odjemalcu IMiS®/Client.

ERR: »Entity handle is invalid or you're trying to open a binary object in read-write mode while the parent entity is opened in read-only mode.«
Referenca ni veljavna (za zgornji opis napake (glej Entity handle is invalid.) ali poizkus odprtja binarnega objekta v načinu za pisanje, čeprav je nadrejena entiteta odprta v načinu za branje.

ERR: »Entity handle must be provided.«
Zahtevek brez reference na entiteto ni veljaven. Napaka na odjemalcu IMiS®/Client.

ERR: »Entity id type cannot be NONE.«
Tip enoličnega identifikatorja entitete v zahtevku ne sme biti N (NONE).
Dovoljene vrednosti so I (šifriran interni), E (eksterni) ali C (klasifikacijska oznaka).
Napaka na odjemalcu IMiS®/Client.

ERR: »Entity identifier not specified.«
ERR: »Entity unique identificator (id or handle) must be provided.«
Zahtevek ne vsebuje veljavnega enoličnega identifikatorja entitete.
Napaka na odjemalcu IMiS®/Client.

ERR: »Entity type '<tip>' is unknown.«
Zahtevek ne vsebuje veljavnega tipa entitete. Dovoljene vrednosti so C (razred), F (zadeva) in D (dokument). Napaka na odjemalcu IMiS®/Client.

ERR: »Entity type '<število>' is not a valid template entity type.«
ERR: »Template enumeration type '<število>' is not a valid entity type.«
Predloga vsebuje neveljavno vrednost tipa entitete, ki jo opisuje. Dovoljene vrednosti so v razponu od 1 do 5. Napaka na odjemalcu IMiS®/Client.

ERR: »Error converting IP address«
Napaka pri pretvorbi znakovnega niza v veljaven IP naslov.
Napaka na odjemalcu IMiS®/Client.

ERR: »Exactly 2 id tags are required in mv request.«
Zahtevek za premik mora vsebovati natančno dve enolični oznaki entitete.
Napaka na odjemalcu IMiS®/Client.

ERR: »Insufficient rights to create file property in the specified enity.«
ERR: »Insufficient rights to create subentities in the destination entity.«
ERR: »Insufficient rights to create subentities under the specified parent.«
Uporabnik nima pravice izvesti zahtevane operacije. Če bi uporabnik moral imeti pravico izvajanja operacije je potrebno preveriti listo dostopnih pravic in jo po potrebi spremeniti.

ERR: »Invalid acl scope request '<znak>'.«
Neveljavna oznaka okvira poizvedbe liste dostopnih pravic. Dovoljene vrednosti so:
N (brez), E (entitete) in A (atributi). Napaka na odjemalcu IMiS®/Client.

ERR: »Invalid date part (<del datuma>).«
ERR: »Invalid date part.«
ERR: »Invalid time part <del časa>).«
ERR: »Invalid time part.«
ERR: »Invalid timestamp.«
Napaka pri pretvorbi znakovnega niza v datum/čas.

ERR: »Invalid old password.«
ERR: »New password required.«
ERR: »Old password required.«
Napaka pri spremembi gesla. Ali staro geslo ni podano (old password required),
ni pravilno (invalid old password) ali pa ni podano novo geslo (new password required). Napaka na odjemalcu IMiS®/Client.

ERR: »Invalid property scope '<znak>' requested.«
Neveljavna vrednost za obseg vrnjenih atributov v »rd« zahtevku. Veljavne vrednosti so:
N (ne vračaj), S (naštete), A (vse) in P (javne). Napaka na odjemalcu IMiS®/Client.

ERR: »Invalid tag %u.«
ERR: »Invalid tag name %u.«
Napaka pri izvajanju zahtevka. Neveljavna XML etiketa.
Napaka na odjemalcu IMiS®/Client.

ERR: »Invalid value id '<število>' cannot be opened.«
Neveljaven identifikator vrednosti binarne vsebine.
Napaka na odjemalcu IMiS®/Client.

ERR: »Invalid XML request: Unknown root tag '<ime>' found.«
Neznana korenska etiketa (root tag) v zahtevku.
Napaka na odjemalcu IMiS®/Client.

ERR: »Invalid/Unsupported key type <identifikator>.«
Nepodprt tip avtentikacijskega ključa.
Napaka na odjemalcu IMiS®/Client.

ERR: »Malformed classification code '<koda>' (<napaka>).«
Nepravilno oblikovana klasifikacijska oznaka.

ERR: »Malformed counter definition '%s'.«
Nepravilna konfiguracija števca generiranja klasifikacijske oznake.
Napaka na odjemalcu IMiS®/Client.

ERR: »Malformed expression '%s'.«
Nepravilen izraz v tabeli šifrantov. Napaka v konfiguraciji.

ERR: »Newly created Binary object identified by '%llu' was not created from this session and cannot be assigned to the property '%s'.«
ERR: »Newly created Binary object identified by '%llu' was not created from this session.«
V zahtevku je posredovan neveljaven identifikator binarnega objekta.
Napaka na odjemalcu IMiS®/Client.

ERR: »No legacy archival configuration for profile '%s'«
Arhivski profil ni vključen v konfiguracijo za arhiviranje.

ERR: »Page element count '<število elementov>' exceeds maximum allowed size of '< število elementov >' elements.«
Število elementov na strani zbirke entitet mora biti manjše od najvišjega dovoljenega števila elementov (10000). Napaka na odjemalcu IMiS®/Client.

ERR: »Parent required but not specified.«
Zahtevek mora vsebovati enolično identifikacijsko številko nadrejene entitete pod katero se ustvari nova entiteta. Napaka na odjemalcu IMiS®/Client.

ERR: »Property '<ime atributa>' is not a streamable property.«
Napačen tip atributa. Podatkovni tok je mogoč samo na atributih tipa »Binary«, »File« in »StringMax«. Napaka na odjemalcu IMiS®/Client.

ERR: »Property code '<ime atributa>': Malformed boolean value '<vrednost>'.«
Nepravilna vrednost binarnega atributa na zahtevku. Dovoljene vrednosti so 0, 1, true ali false. Napaka na odjemalcu IMiS®/Client.

ERR: »Property name must be provided.«
Zahtevek mora vsebovati ime atributa. Napaka na odjemalcu IMiS®/Client.

ERR: »Provided client application name contains invalid UTF-16 characters.«
Ime odjemalca v zahtevku vsebuje nedovoljen UTF-16 znak. Napaka na odjemalcu IMiS®/Client.

ERR: »Provided client application name exeeds maximum length of <število> UTF-16 characters.«
Ime odjemalca v zahtevku je predolgo. Napaka na odjemalcu IMiS®/Client.

ERR: »Provided client local address '<IP naslov>' cannot be converted into its binary form (Detail: <opis napake>)«
Napaka pri pretvorbi internega IP naslova. Navadno označuje napačno obliko posredovanega naslova. Napaka na odjemalcu IMiS®/Client.

ERR: »Provided client local address contains invalid UTF-8 characters.«
Odjemalčev lokalni IP naslov v zahtevku vsebuje nedovoljen UTF-8 znak. Napaka na odjemalcu IMiS®/Client.

ERR: »Provided client local address exeeds maximum length of <število> UTF-8 characters.«
Odjemalčev lokalni IP naslov v zahtevku je predolg. Napaka na odjemalcu IMiS®/Client.

ERR: »Provided computer name contains invalid UTF-16 characters.«
Ime uporabnikovega računalnika v zahtevku vsebuje nedovoljen UTF-16 znak.
Napaka na odjemalcu IMiS®/Client.

ERR: »Provided computer name exeeds maximum length of <število> UTF-16 characters.«
Ime uporabnikovega računalnika v zahtevku je predolgo.
Napaka na odjemalcu IMiS®/Client.

ERR: »Provided username contains invalid UTF-16 characters.«
Uporabniško ime v zahtevku vsebuje nedovoljen UTF-16 znak.
Napaka na odjemalcu IMiS®/Client.

ERR: »Provided username exeeds maximum length of <število> UTF-16 characters.«
Uporabniško ime v zahtevku je predolgo. Napaka na odjemalcu IMiS®/Client.

ERR: »Reason for deletion required but missing or empty.«
Uporabnik mora obvezno podati razlog za brisanje. Napaka na odjemalcu IMiS®/Client.

ERR: »Server was unable to decrypt authentication payload using any configured crypto contexts.«
Težava pri avtentikaciji na starem vmesniku, saj ni bilo mogoče dešifrirati avtentikacijskega paketa. Napaka na odjemalcu IMiS®/Client.

ERR: »Start index '<začetni indeks>' exceeds collection element count '<število elementov>'.«
Neveljaven začetni indeks na zbirki. Začetni indeks ne sme biti višji od števila elementov v zbirki. Napaka na odjemalcu IMiS®/Client.

ERR: »Template '<ime>' is internal and cannot be used.«
Uporaba internih predlog ni dovoljena. Napaka na odjemalcu IMiS®/Client.

ERR: »The password is not encoded in correct UTF-8 sequence.«
Geslo ni veljaven UTF-8 znakovni niz. Napaka na odjemalcu IMiS®/Client.

ERR: »The SRP6A group you provided is not supported (id=<število>).«
SRP6A skupina ni podprta.

ERR: »Time must be in range!«
ERR: »Timestamp must be a range.«
Zahtevek za poizvedbo po revizijski sledi mora obvezno vsebovati časovni razpon iskanih zahtevkov. Napaka na odjemalcu IMiS®/Client.

ERR: »Unable to make query without any parameters.«
Zahtevek za poizvedbo po revizijski sledi mora obvezno vsebovati vsaj en parameter.
Napaka na odjemalcu IMiS®/Client.

ERR: »Unable to search objects by object id range.«
ERR: »We cannot have ranges of objects«
Zahtevek za poizvedbo po revizijski sledi ne sme vsebovati razpona po enoličnih oznakah objekta. Napaka na odjemalcu IMiS®/Client.

ERR: »Unknown collection management operation '<znak>'.«
Zahtevek vsebuje nedovoljeno vrednost za upravljanje z zbirko. Dovoljeni vrednosti sta C (briši zbirko) in I. Napaka na odjemalcu IMiS®/Client.

ERR: »Unknown property '<ime atributa>'.«
ERR: »Unknown property code '<ime atributa>'.«
Atribut ne obstaja. Napaka na odjemalcu IMiS®/Client.

ERR: »Unknown stream handle (<šestnajstiško število>). Ignoring request.«
Nepravilna referenca na podatkovni niz. Napaka na odjemalcu IMiS®/Client.

ERR: »Unknown value id '<število>' requested or error opening a stream.«
Neveljaven identifikator vrednosti binarne vsebine. Napaka na odjemalcu IMiS®/Client.

ERR: »Unknown XML Parser error.«
Neznana napaka pri procesiranju XML zahtevka. Napaka na odjemalcu IMiS®/Client.

ERR: »Unknown xml tag encountered: '<število>'.«
ERR: »Unknown xml tag encountered: '<string>'«
Pri procesiranju XML zahtevka se je pojavila nepoznana XML etiketa.
Napaka na odjemalcu IMiS®/Client.

ERR: »Unsupported object id size.«
Za šifrirane enolične oznake entitete sta dovoljeni samo dve dolžini: 24 in 32 znakov.
Napaka na odjemalcu IMiS®/Client.

ERR: »XML Parser error: domain=%d, code=%d, msg='%s', level='%s', line=%d«
Problem pri procesiranju XML zahtevka. Napaka na odjemalcu IMiS®/Client.

ERR: »You cannot assing an existing Binary object to the property '%s' from another entity. The offending value is '%llu'.«
Istega binarnega atributa ni mogoče dodeliti dvema različnima entitetama.

ERR: »You cannot create a <tip entitete> from template '<ime predloge>' (Reason: Template is not of type <tip entitete>).«
Predloge ni mogoče uporabiti na danem tipu entitete.
Napaka na odjemalcu IMiS®/Client.

ERR: »Unable to open the listening port on address '[<naslov>]:<port>' (Reason: '<razlog>').«
ERR: »Unable to bind the listening socket to address '[%s]:%u' (Reason: '%s').«
Napaka pri odpiranju omrežnega naslova za serviranje zahtevkov. Razlog je naveden v sporočilu »Reason«.

ERR: »Access denied. System entities can not be opened.«
Prepovedano operacija »odpri« na sistemski entiteti.

ERR: »Classification code cannot be set for existing entities.«
Prepovedano spreminjanje klasifikacijske oznake na obstoječi entiteti.

ERR: »Classification code is set to be generated automatically.«
Klasifikacijska oznaka se generira samodejno. Ročno nastavljanje ni dovoljeno.

ERR: »Invalid stream object for CS_OBJ_UPDATE request.«
Neveljaven zahtevek za odpiranje »stream-a« na starem vmesniku.

ERR: »Only end templates can be removed.«
Predloga, ki jo želite odstraniti ne sme imeti podrejenih predlog.

ERR: »Entites based on template '<ime predloge>' exist. Template cannot be removed.«
Prepovedano odstranjevanje predloge, ki je uporabljena v vsaj eni entiteti.

ERR: »Invalid CS_OBJ_CREATE request packet size: <dolžina>.«
ERR: »Invalid CS_OBJ_OPEN request packet size: '<dolžina>'.«
ERR: »Invalid CS_OBJ_UPDATE request packet size: '<dolžina>'.«
Neveljavna dolžina zahtevka na starem vmesniku.

ERR: »<ime atributa>' property not found.«
Atribut ne obstaja.

ERR: »Default storage profile is not set.«
Napaka v nastavitvi. Potrebno je določiti privzeti arhivski profil.

ERR: »Storage profile '<oznaka>' doesn't exist.«
Napaka v nastavitvi. Arhivski profil z dano oznako ne obstaja.

ERR: »Error acquiring an instance of Full Text Index Service provider.«
Napaka v nastavitvi.

ERR: »Classification code is set to be generated automatically but generator is not configured for the entity's hierarchy level.«
Napaka v nastavitvi. Samodejno generiranje klasifikacijskih oznak ni nastavljeno za vse nivoje v načrtu razvrščanja gradiva.

ERR: »Classification code must be set by creator and cannot be empty.«
Klasifikacijska oznaka je obvezen podatek.

ERR: »Classification code must be set to be generated automatically for the destination entity's subentities.«
ERR: »Classification code must be set to be generated automatically for the whole moving entity's hierarchy.«
ERR: »Missing classification code generator. Entity type: '<tip entitete>', Absolute level: <nivo>, Relative level: <nivo>.2
Napaka v nastavitvah. Operacija premik obvezno potrebuje nastavljeno samodejno generiranje klasifikacijskih oznak na ciljnem drevesu.

ERR: »Directory entity with name '<ime>' (id=<število>) is deleted.«
ERR: »Group '<ime>' (id=<število>) has been deleted.«
Poizkus brisanja entitete, ki je bila že izbrisana.

ERR: »Directory entity '<ime>' (id=<število>) has been deleted while being edited.
Group '<ime>' (id=<število>) has been deleted while being edited.«
Poizkus spreminjanj entitete, ki je bila že izbrisana.

ERR: »Directory entity '<ime>' (id=<število>) is not being edited.«
ERR: »Group '<ime>' (id=<število>) is not being edited.«
Poizkus spreminjanja entitete, ki ni bila odprta v načinu za spreminjanje.

ERR: »Non-empty entities cannot be deleted.«
Brisana entiteta ne sme vsebovati vsebovanih entitet. Najprej je potrebno izbrisati podrejene entitete.

ERR: »Entity '<id>' cannot be opened in exclusive mode.«
Entiteta je že odprta v načinu za pisanje v neki drugi seji.

ERR: »Value '<id>' is currently being edited. Close all editable streams before opening a new one.«
ERR: »Value '<id>' is currently being edited through stream '<referenca streama>'. Close it before opening a new one.«
Atribut tipa podatkovni tok je že odprt za pisanje.

ERR: »Closed entities cannot be edited.«
Status dotične entitete (ali nadrejene) je »Closed«. Kakršno koli spreminjanje zaprte entitete je prepovedano.

ERR: »Access denied. Request requires user-credentials authenticated session or higher.«
Za izvajanje želene operacije je potrebno vzpostaviti sejo z avtentikacijo.

ERR: »Adding new entities under closed entities is not allowed.«
Status dotične entitete (ali nadrejene) je »Closed«. Dodajanje novih podrejenih entitet pod zaprto entiteto ni dovoljeno.

ERR: »Destination status is 'Closed'. Adding new entities under it is not allowed.«
Kakršnokoli spreminjanje entitete s statusom »Closed« je prepovedano. To vključuje tudi dodajanje novih podrejenih entitet.

ERR: »Access denied. (You do not have the right to delete the property '<ime>')«
Uporabnik nima pravice brisanja atributov na entiteti.

ERR: »Access denied. (You do not have the right to create the property '<ime>')«
Uporabnik nima pravice dodajanja atributov na entiteti.

ERR: »Access denied. (You do not have the right to edit the property '<ime>')«
Uporabnik nima pravice spreminjanja vrednosti atributov na entiteti.

ERR: »Query result set is to large. Try to decrease datetime range.«
Rezultat poizvedbe revizijske sledi zajame več rezultatov kot jih je sistem zmožen obdelati. Spremeniti je potrebno parametre iskanja, da bo vrnjenih manj rezultatov. Najlažje se to stori tako, da se zmanjša časovni razpon iskanih dogodkov.

ERR: »Malformed entity id«
ERR: »Unable to decrypt entity id«
Napaka pri dešifriranju enolične oznake entitete. Napaka na odjemalcu IMiS®/Client.

9.4.5 Nivo 4 – Warning

WARN: »File descriptor <decimalna vrednost> is too big for select(). Just closing.«
Prišlo je do klica funkcije za zapiranje seje uporabnika z napačnim parametrom.
Vzrok za opozorilo je napaka v programu. Napaka je zgolj teoretična in je potreben pregled strokovnjaka proizvajalca.

WARN: »Object header: Illegal server ID.«
Objekt je bil verjetno prenesen z drugega strežnika ali pa je datoteka objekta okvarjena. Možno je tudi, da je bil strežnik IMiS®/ARChive Server nadgrajen, vendar iz paketa z drugim strežniškim identifikatorjem. V vseh treh primerih je potreben poseg/pregled in mnenje strokovnjaka proizvajalca.

WARN: »Object header: Illegal head.«
Objektna datoteka je najverjetneje okvarjena, oziroma je bila mimo strežnika IMiS®/ARChive Server nadomeščena z datoteko napačne vsebine. Kar se tiče strežnika je ta datoteka neuporabna oziroma neberljiva.

WARN: »Object header: Object ID mismatch (ObjID: < šestnajstiška vrednost>; Header: <šestnajstiška vrednost>).«
Objekt ima v glavi vpisan napačen identifikator objekta. Lahko gre za okvarjen objekt ali pa je napaka posledica napake pri določanju identifikatorja objekta. Potreben je poseg strokovnjaka proizvajalca. Kar se tiče strežnika IMiS®/ARChive Server je ta datoteka neuporabna oziroma neberljiva.

WARN: »No available volume found in profile <desetiška vrednost>.«
Profilu za shranjevanje je zmanjkalo prostora. Potrebno je dodati nov volumen ali pa povečati obstoječe volumne profila.
WARN: »Seek ofset underflow. Repositioning.«
Prišlo je do poizkusa dostopa do objekta na neveljavni lokaciji. Vzrok za napako je nepravilno delovanje odjemalca IMiS®/Client. Napaka je zgolj teoretična.

WARN: »Unsupported Cipher algorithm requested for 128bit key strength (alg_id=<desetiška vrednost>)", alg_id).«
WARN: »Invalid Cipher mode requested (id=<desetiška vrednost>)", mode_id).«
WARN: »Unsupported Cipher key strength requested (key_strength=<desetiška vrednost>)", key_strength).«
WARN: »Unsupported block size identifier (<desetiška vrednost>)", bs).«
WARN: »Block size identifier (<desetiška vrednost>) doesn't match crypto context.", bs).«
WARN: »Crypto exception occured (details: <opis podrobnosti>)", e.what()).«
Do navedenih opozoril pride, ko odjemalec IMiS®/Client zahteva način šifriranja,
ki v konfiguraciji strežnika ni omogočen. Vzrok za napako je nekompatibilna nastavitev šifrirnega podsistema odjemalca, oziroma neustrezna nastavitev strežnika v kombinaciji z nastavitvijo odjemalcev.

WARN: »Unknown exception occured while setting up crypto context.«
Pri vzpostavljanju okolja za šifrirano komunikacijo z odjemalcem IMiS®/Client je prišlo do nepričakovane napake. Napaka je zgolj teoretična. Potreben pregled strokovnjaka proizvajalca.
WARN: »Audit Log Query session denied by configuration settings.«
Strežnik IMiS®/ARChive Server je zavrnil zahtevek za vzpostavitev seje za pregled revizijske sledi, ker poizkuša odjemalec IMiS®/Client vzpostaviti sejo z nepodprtim naborom šifrirnih parametrov.

WARN: »Unsupported key type (type=<vrednost>).«
Tip ključa, ki ga odjemalec IMiS®/Client poizkuša uporabiti pri vzpostavitvi seje za pregled revizijske sledi, ni podprt ali dovoljen. Potrebno je preveriti nastavitve strežnika in odjemalca ter ju uskladiti.

WARN: »Illegal length of data recieved (<desetiška vrednost>). Ignoring request <desetiška vrednost>.«
Pri komunikaciji z odjemalcem IMiS®/Client je strežnik IMiS®/ARChive Server prejel zahtevek, katerega dolžina ni pravilna. Zahtevka strežnik ne upošteva in ga zavrne. Napaka pomeni napako v odjemalcu IMiS®/Client in je zgolj teoretična.

WARN: »Error deleting object (<desetiška vrednost>).«
Pri izvajanju zahtevka za brisanje objekta na strežniku IMiS®/ARChive Server je prišlo do napake. Vzrok so lahko nepravilno spremenjene pravice dostopa do objektov datoteke ali pa datoteke ni na svojem mestu.

WARN: »Request <šestnajstiška vrednost> not expected. Ignored.«
Tip zahtevka, ki ga je strežnik IMiS®/ARChive Server prejel od odjemalca IMiS®/Client ni pričakovan v stanju seje, katero zazna ročica seje odjemalca oziroma ni regularen.
Zahtevek strežnik zavrže kot neveljaven. Navadno so to zahtevki, ki jih odjemalci pošiljajo ne-zavedajoč se, da je prišlo do ponovnega zagona strežnika.
Zato njihovo zavračanje ne povzroči nepravilnega delovanja.

WARN: »Volume "<opis volumna>" has no profile assigned.«
Volumen, ki je sicer regularno vpisan v interni bazi strežnika IMiS®/ARChive Server, ni dodeljen profilu. Vzrok je najverjetneje neskladje v interni bazi strežnika.
Potreben je poseg strokovnjaka proizvajalca.

WARN: »Cannot create a socket (out of file descriptors?), error <vrednost>: <vrednost>.«
Strežnik IMiS®/ARChive Server je izkoristil največje dovoljeno število odprtih datotek.
Ustrezno je potrebno povečati sistemsko nastavitev za največje dovoljeno število odprtih datotek za uporabnika, s čigar privilegiji teče strežnik (privzeto `iarc`).

WARN: »Configuration parameter '<ime parametra>' has invalid structure and will be ignored.«
Namestitveni parameter v `/etc/iarc.conf` datoteki je napačno vpisan oziroma ne ustreza pričakovanemu naboru vrednosti. Strežnik IMiS®/ARChive Server nastavitve ne upošteva in namesto tega uporabi privzeto vrednost.

WARN: »Error <koda napake> while getting object id for external id <identifikator>.«
V podatkovni bazi ni objekta, ki bi bil predhodno povezan z zunanjim identifikatorjem vrednosti <identifikator>.

WARN: »Error <koda napake> while setting external id for object <identifikator objekta>.«
Pri povezovanju obstoječega objekta z zunanjim identifikatorjem je prišlo do napake.
Napaka je zgolj teoretična, lahko da je vzrok tudi nepravilnost v interni bazi podatkov.
O opozorilu je potrebno obvestiti strokovnjaka proizvajalca.

WARN: »User <ime uporabnika> from <ime računalnika> did not authenticate with Audit Log Query permissions. Query denied!«
Uporabnik je poslal zahtevek za iskanje po revizijski sledi, vendar za to nima pravic ker uporabnikova seja ni bila ustrezno avtentificirana. Vzrok je lahko napačna konfiguracija odjemalca IMiS®/Client, lahko pa pomeni opozorilo o neavtoriziranem delovanju določenega uporabnika.

WARN: »semop() ended with error <koda napake>.«
Prišlo je do sistemske napake pri sinhronizaciji procesov. Vzrok je najverjetneje nepravilno delovanje operacijskega sistema ali pomanjkanje sistemskih sredstev. Posledično je lahko moteno normalno delovanje strežnika IMiS®/ARChive Server,
ni pa nevarnosti za že shranjene podatke. O opozorilu je priporočljivo obvestiti strokovnjaka proizvajalca in ponovno zagnati operacijski sistem in s tem storitev.

WARN: »dup(<desetiška vrednost>) error <koda napake>: <sistemski opis napake>.«
Prišlo je do sistemske napake pri podvajanju sistemskega identifikatorja.
Vzrok je najverjetneje nepravilno delovanje operacijskega sistema ali pomanjkanje sistemskih sredstev in je podrobneje opisan v <sistemski opis napake>.
Posledično je lahko moteno normalno delovanje strežnika IMiS®/ARChive Server,
ni pa nevarnosti za že shranjene podatke. O opozorilu je priporočljivo obvestiti strokovnjaka proizvajalca in ponovno zagnati operacijski sistem in s tem storitev.

WARN: »Cannot put socket <desetiška vrednost> in listen mode, error <koda napake>: <sistemski opis napake>. Skipping to next...«
Prišlo je do sistemske napake pri čakanju na zahtevke za nove povezave odjemalcev.
Vzrok je lahko pomanjkanje sistemskih sredstev ali pa napačna nastavitev strežnika IMiS®/ARChive Server.

WARN: »Message queue full. Increase number of serving threads.«
Čakalna vrsta zahtevkov je zapolnjena. Priporočljivo je povečati število procesnih niti v nastavitvi.

WARN: »Configured service '<ime storitve>' cannot be resolved to a discreet port number (error: <sistemski opis napake>). Falling back to default '<ime storitve>'...«
WARN: »Default service '<ime storitve>' cannot be resolved to a discreet port number (error: <sistemski opis napake>). Default port will not be configured.«
WARN: »Default service '<ime storitve>' resolves to a unsupported protocol family. Default port will not be configured.«
WARN: »Configured service '<ime storitve>' resolves to a unsupported protocol family. Falling back to default '<ime storitve>'...«
WARN: "Address '<omrežni naslov>', service '<ime storitve>' skipped since it cannot be resolved (error: <sistemski opis napake>).«
Opozorila pomenijo napako pri inicializaciji sistema za omrežne povezave.
Vzrok za napako je lahko napačna konfiguracija strežnika IMiS®/ARChive Server ali pa nepravilno delovanje operacijskega sistema.

WARN: »Maximum number of listening sockets reached (max = <decimalna vrednost>). Additional addresses will not be used!«
Opozorila pomenijo napako pri inicializaciji sistema za omrežne povezave.
Vzrok za napako je lahko napačna nastavitev strežnika IMiS®/ARChive Server ali pa nepravilno delovanje operacijskega sistema.

WARN: »Unable to locate template <desetiška vrednost> in entity inventory. Template:Attribute bind record <desetiška vrednost> : <desetiška vrednost> will be ignored.«
Iskane entitete ni v podatkovni bazi.

WARN: »Unable to locate attribute <desetiška vrednost> in attribute inventory. Template:Attribute bind record <desetiška vrednost> : <desetiška vrednost> will be ignored.«
WARN: »Configuration[ContentParsers]: Content parsing option DISABLED. Reason: '<opis>'«
WARN: »Configuration[FullTextIndex]: FullTextIndex option DISABLED. Reason: '<opis>'«
WARN: »Configuration[<xml etiketa>]: Attribute '<attribute name>' required but missing or empty. Configuration record skipped.«
WARN: »Configuration[<xml etiketa>]: Attribute '<attribute name>' contains an unsupported value '<string>'. Configuration record skipped.«
WARN: »Configuration[<xml etiketa>]: Template '<string>' not found or error accessing it. Configuration record skipped.«
WARN: »Configuration[<xml etiketa>]: Invalid attribute type '<desetiška vrednost>' for '<string>' (FILE required). Configuration record skipped.«

WARN: »Configuration[<xml etiketa>]: Configuration for template '<string>' already exists. Using first configuration record with attribute '<string>'.«
WARN: »Configuration[<xml etiketa>]: Unknown configuration parameter '<string>' encountered. Skipping...«
WARN: »Legacy Achival option DISABLED. Reason: '<string>'«
WARN: »Using default Security Options. Reason: '<string>'«
WARN: »Configuration[<xml etiketa>]: Unknown configuration parameter '<string>' encountered. Skipping...«
WARN: »Invalid XML request: Unknown request tag '<ime etikete>' found, ignored.«
Zahtevek vsebuje nepričakovano XML etiketo. Napaka ni kritična in ne vpliva na delovanje strežnika IMiS®/ARChive Server. Nepričakovano XML etiketo strežnik ignorira. Opozorilo je pričakovano v primeru, ko je verzija odjemalca novejša od verzije strežnika.

WARN: »Volume info: Volume <število> not found in database!«
Volumna ni bilo mogoče najti v podatkovni bazi. Preveriti je potrebno nastavitve volumnov.

