

IMiS^(R) / ARChive Server

verzija 9.1.1406

IMAGING
SYSTEMS

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

KAZALO VSEBINE

1	PREDGOVOR.....	7
1.1	O dokumentaciji	7
1.2	Ciljno občinstvo	7
1.3	Konvencije	7
1.4	Kratice	8
2	UVOD	12
2.1	Predstavitev	12
2.2	Verzioriranje in označevanje.....	13
2.3	Funkcionalnosti.....	14
3	TEHNIČNA DOKUMENTACIJA.....	16
3.1	Arhitektura strežnika.....	16
3.2	Atribut	23
3.2.1	Vrste.....	24
3.2.2	Zmožnosti	31
3.2.3	Povezava s predlogami.....	32
3.2.4	Poimenovanje.....	36
3.2.5	Sistemske atributi	36
3.2.6	Atributi dokumentov elektronske pošte.....	48
3.2.7	Atributi upravljanja s fizičnim gradivom.....	51
3.2.8	Atributi prenesenega gradiva.....	54
3.3	Entiteta.....	57
3.3.1	Vrste.....	57
3.3.2	Hierarhija.....	58
3.3.3	Komponente	59
3.3.4	Predloge.....	60
3.3.5	Dostopi.....	60
3.3.6	Identifikatorji.....	71
3.3.7	Življenjski cikel.....	74
3.3.8	Revizijska sled.....	83
3.4	Razvrščanje	93
3.4.1	Klasifikacijske oznake.....	94
3.4.2	Nastavitve načrta razvrščanja gradiva	94
3.4.3	Premikanje gradiva v načrtu razvrščanja (reklasifikacija)	96
3.5	Iskanje	97
3.5.1	Varnost in zaščita podatkov pri iskanju.....	97
3.5.2	Pravila iskalnega niza	98
3.6	Avtentičnost.....	101
3.6.1	Predpogoji	101
3.6.2	Koncept.....	102

3.6.3	Shramba digitalnih potrdil	104
3.6.4	ERS.....	109
3.6.5	AIP.....	129
3.6.6	Časovno žigosanje	136
3.6.7	Pravila	140
3.7	Imeniške storitve	141
3.7.1	Tipi.....	142
3.7.2	Preddefinirane (sistemsko) določene entitete.....	143
3.7.3	Komponente entitete	144
3.7.4	Sinonimi.....	146
3.7.5	Avtentikacija.....	146
3.7.6	Življenjski cikel entitet imenika.....	152
3.8	Varnostno kopiranje in obnovitev podatkov.....	154
3.8.1	Varnostno kopiranje.....	158
3.8.2	Obnovitev podatkov	158
3.8.3	Primer	160
3.8.4	Težave pri obnovitvi podatkov	161
4	SISTEMSKE ZAHTEVE	162
4.1	Strojna oprema	162
4.1.1	Načrtovanje procesorske moči strežnika	162
4.1.2	Načrtovanje pomnilniških kapacitet strežnika	163
4.1.3	Načrtovanje diskovnih kapacitet strežnika	163
4.1.4	Komunikacijske poti.....	165
4.1.5	Priklop na mrežno opremo.....	165
4.1.6	Administratorske pravice	165
4.1.7	Nadzor delovanja strojne opreme	166
4.1.8	Minimalne zahteve.....	166
4.1.9	Priporočene zahteve	166
4.2	Programska oprema	167
4.2.1	Operacijski sistemi.....	167
4.2.2	Seznam obveznih sistemskih orodij.....	168
4.2.3	Seznam obveznih sistemskih knjižnic.....	168
4.2.4	Minimalne zahteve.....	169
5	NAMESTITEV	169
5.1	Postopek namestitve	169
5.2	Ponamestitveni postopki.....	170
5.2.1	Nastavitev števila hkrati odprtih datotek	171
5.2.2	Nastavitev samodejnega zagona	172
5.3	Preizkušanje namestitve in nastavitev	172
6	NADGRADNJA	173
6.1	Postopek nadgradnje.....	173

6.2	Možni zapleti pri nadgradnji.....	174
7	ODSTRANITEV.....	176
7.1	Postopek odstranitve.....	176
8	UPRAVLJANJE PRODUKTA.....	177
8.1	Postopek zagona in zaustavitve.....	177
8.2	Beleženje dogodkov delovanja.....	178
8.3	Konfiguriranje.....	180
8.3.1	Predvidena opravila.....	180
8.3.2	Postopki konfiguriranja s konzolnimi orodji.....	182
8.4	Administracija.....	184
8.4.1	Konfiguracijska datoteka iarc.conf.....	185
9	ODPRAVLJANJE TEŽAV.....	191
9.1	Kako se težavam izognemo?.....	192
9.2	Pogoste težave.....	192
9.3	Redkejšje težave.....	200
9.4	Seznam napak storitve, ki se beležijo v dnevnik delovanja.....	203
9.4.1	Nivo 0 – Emergency.....	203
9.4.2	Nivo 1 – Alert.....	203
9.4.3	Nivo 2 – Critical.....	204
9.4.4	Nivo 3 – Error.....	206
9.4.5	Nivo 4 – Warning.....	225

KAZALO SLIK

Slika 1: Shematični prikaz arhitekture strežniške komponente IMiS®/ARChive Server	16
Slika 2: Logične povezave med atributi, predlogami in entitetami.....	33
Slika 3: Delovanje ACL.....	61
Slika 4: Hierarhija z razredom, zadevo in dvema dokumentoma.....	68
Slika 5: Življenjski cikel entitete.....	75
Slika 6: Življenjski cikel instance entitete v delovnem spominu strežnika	76
Slika 7: Umeščanje entitete v postopku za zagotavljanje avtentičnosti dolgoročno arhiviranih podatkov.....	101
Slika 8: Delovanje zgoščevalne funkcije	102
Slika 9: Generiranje in preverjanje elektronskega podpisa.....	103
Slika 10: Veriga digitalnih potrdil.....	105
Slika 11: Postopek dodajanja digitalnega potrdila v shrambo	106
Slika 12: Ustvarjanje AIP-ja za proces ustvarjanja dokazil	109
Slika 13: Podaljševanje zanesljivosti dokazil	110
Slika 14: Primer preverjanja avtentičnosti podatkov.....	111
Slika 15: Postopek časovnega žigosanja posameznega AIP-ja.....	111
Slika 16: Postopek časovnega žigosanja z uporabo Merklvega drevesa	112
Slika 17: Postopek enostavnega podaljševanja v kombinaciji z ustvarjanjem dokazil	113
Slika 18: Del procesa kompleksnega podaljševanja, opisanega v točkah 2,3 in 4	114
Slika 19: Primer Merklvega drevesa	116
Slika 20: Primer reduciranega drevesa za vozlišči H1 in H2.....	117
Slika 21: Enostavno podaljševanje arhivskega časovnega žiga.....	124
Slika 22: Postopek obdelave AIP in pripadajočega ERS.....	126
Slika 23: Postopek kompleksnega podaljševanja z Merklvim drevesom.....	127
Slika 24: Koncept vtičnikov	137
Slika 25: Pripadnost uporabnika v uporabniških skupinah.....	142
Slika 26: Primer gnezdenja uporabniških skupin	143
Slika 27: Primer napada na IMiS®/ARChive Server z vrinjenim napadalcem.....	148
Slika 28: Povezava med procesi v življenjskem ciklu entitete v imeniku.....	152

KAZALO TABEL

Tabela 1: Uporaba različnih stilov v dokumentaciji.....	7
Tabela 2: Uporaba kratic v dokumentaciji	10
Tabela 3: Seznam uporabljenih pojmov v dokumentaciji.....	11
Tabela 4: Tabela operacij in pravic	65
Tabela 5: Omejitev pravic vpogleda v javne metapodatke.....	66
Tabela 6: XML elementi etikete »EvidenceRecord«	118
Tabela 7: XML elementi etikete »ArchiveTimeStampChain«	119
Tabela 8: XML elementi etikete »ArchiveTimeStamp«	120
Tabela 9: XML elementi etikete »Sequence«	120
Tabela 10: XML elementi etikete »TimeStamp«	121
Tabela 11: Podprti tipi časovnih žigov	122
Tabela 12: Podprti tipi kriptografskih elementov	122
Tabela 13: XML elementi etikete »Header«.....	130
Tabela 14: Interpretacija AIP v odvisnosti od vrednosti atributa »Version«	130
Tabela 15: XML elementi etikete »Attribute«	131
Tabela 16: XML elementi etikete »Content«.....	132
Tabela 17: XML elementi etikete »ContentValue«	132
Tabela 18: Algoritmi in pripadajoči URI-ji	133
Tabela 19: XML elementi etikete »Signature«	133
Tabela 20: XML elementi etikete »Certificate«.....	134
Tabela 21: XML elementi etikete »RevocationData«	136
Tabela 22: Podprti tipi informacij o preklicu digitalnih potrdil.....	136
Tabela 23: Tabele z opisom podatkov za varnostno kopiranje in obnavljanje.....	157
Tabela 24: Povprečne velikosti skeniranega dokumenta pri uporabi različnih metod stiskanja.....	164

1 PREGOVOR

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)
<u>Navadno podčrtano</u>	dodatne možnosti izbora podpoglavij znotraj posameznega nivoja
»Navadno«	nazivi funkcij ali akcij v okviru možnosti izbora
<i>Navadno ležeče</i>	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)
OCSF	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 Rivest, Adi Shamir, Leonard Adleman (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

Kratika	Opis
TCP/IP	(algoritmi za izračun prstnega odtisa vsebine) Transmission Control Protocol / Internet Protocol (družina omrežnih protokolov)
TSA	Time Stamp Authority (agencija za izdajo kvalificiranih časovnih žigov)
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 arhivirano gradivo 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 unikatni 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.ARCHITECTURE.rpm`

Primer: `imisarc.9.1.1406-600.0001.rdm.e14.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 maj 2012 je v RELEASE identifikatorju označena kot 1205.
- **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 unikatni 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 vsebovano 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 8, 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 hranjene vsebine.

- Hramba metapodatkov in vsebine 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 dolgoročne avtentičnosti hranjenega gradiva preko ustvarjanja in vzdrževanja dokaznih elementov o nespremenjenosti hranjenega gradiva (LTANS).
- Imeniške storitve za podporo prijavi uporabnika, upravljanjem z dostopnimi pravicami do hranjenih vsebin in razširitvi 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).

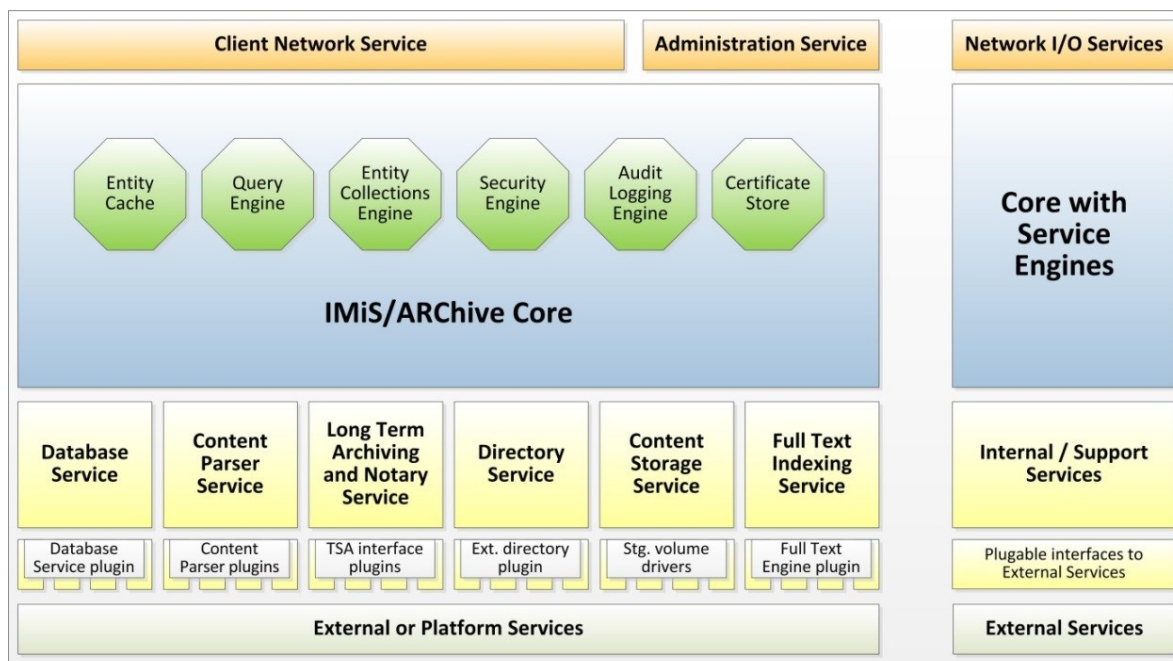
3 TEHNIČNA DOKUMENTACIJA

3.1 Arhitektura strežnika

IMiS®/ARChive Server predstavlja strežniško komponento sistema za upravljanje z gradivom v elektronski in fizični 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
 - opsijsko 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, ...

Strukturirane so 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: Storitve sistemu nudi nivo podatkovne zbirke tabel in indeksov, potrebnih za delovanje. Storitve 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 opsijsko 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 štiti 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 je na voljo v poglavju 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 je na voljo v poglavju 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 attribute entitet, digitalnih vsebin pa ne.

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

Pomen: Neobvezen

Opis: Storitve 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 [podsystem 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 [podsystem 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 podsystem pogosto uporabljajo funkcije odjemalcev za hierarhični vpogled v drevo arhiviranih entitet (vse pod-entitete neke starševske entitete, ipd.) in podsystem 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 [podsystem 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. Dostopne pravice omogočajo upravljanje z dostopnimi pravicami, njihovim nivojem, upravljanjem z dostopnimi pravicami in življenjskim ciklom podrejenih entitet.

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 nastavitvev, 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 nastavitvev 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 [podsystem za izmenjavo informacij med jedrom in odjemalci sistema]

Pomen: Kritičen

Opis: Podsystem 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. Storitve omogoča povezovanje preko več nastavljivih TCPIP vrat in naslovov obeh IPv4 in IPv6 skladov.

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, jih je potrebno uokviriti in jim predpisati pravila.

To v strežniku IMiS®/ARChive Server dosežemo z uporabo koncepta Atribut, katerih skupine pripišemo Entiteti ([glej poglavje 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 ([glej poglavje 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 algebr.

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: -9223372036854775807.8, Max: 9223372036854775808.7

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 9223372036854775807 bajtov,

opis z velikostjo do 4096 bajtov UTF-8 znakov.

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

3.2.2 Zmožnosti

Atribut poleg njegovega tipa opisujejo še naslednji podatki:

- »Name«: naziv, obvezen unikatni niz UTF-8 znakov, dolg do 256 zlogov UTF-8 znakov.
- »Description«: opis atributa, opcijski niz UTF-8 znakov, dolg do 512 zlogov UTF-8 znakov.
- »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.
- »Parameters«: dodatni parametri atributa, ki poleg njegovega podatkovnega tipa dodatno uokvirjajo pripisane vrednosti: »Searchable«, »Unique« in »PickList«.

Parameter atributa »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 odločiti, 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.

Parameter atributa »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.

Parameter atributa »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 odvezemo. Dodatno lahko vsaki vrednosti določimo:

- **Atribute:**
 - "alias": sinonim, uporabniku prijazen naziv vredosti, ki ga odjemalci lahko uporabijo za izbire vrednosti.
 - "alias.xx_YY": preveden sinonim, uporabniku prijazen naziv vredosti, 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.

Primeri:

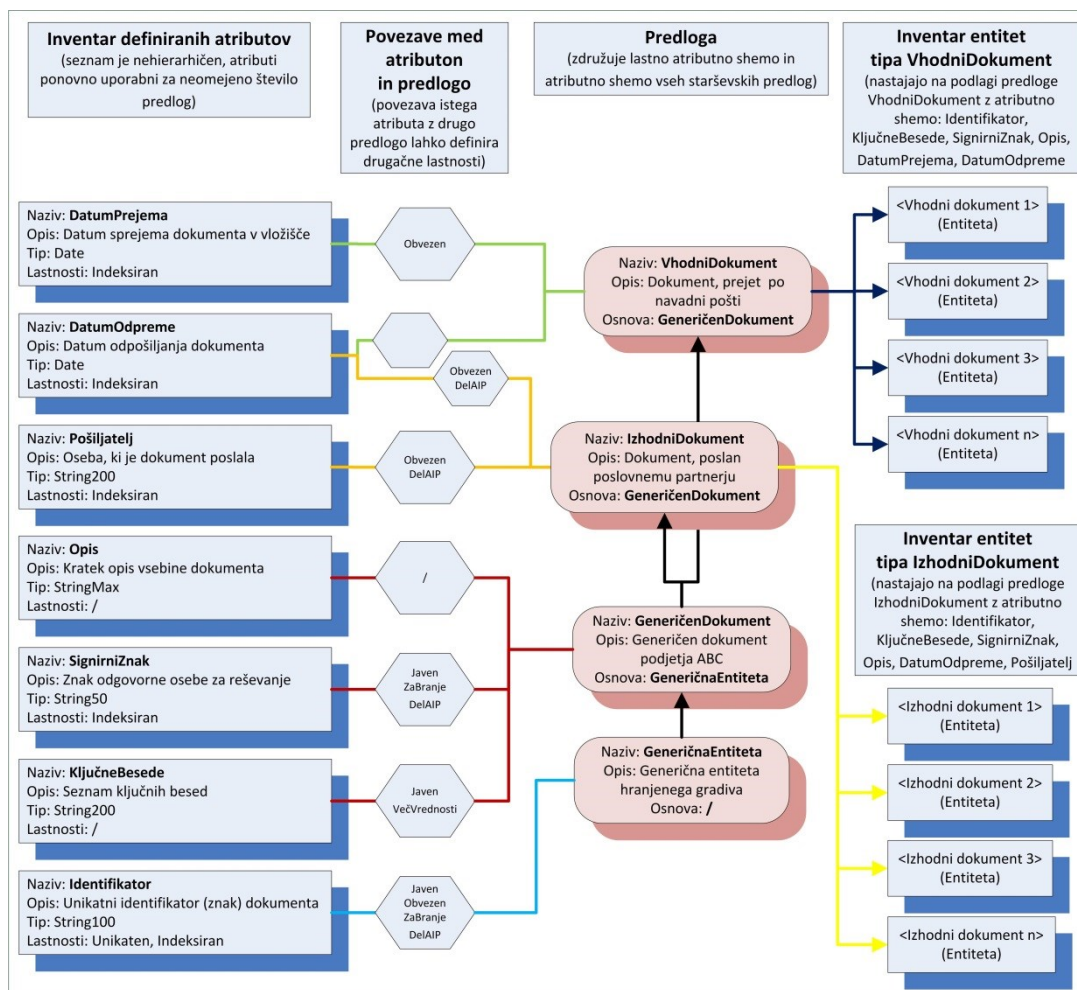
- "1:{default,alias="Opened",alias.sl_SI="Odprto"} – Vrednost atributa »1« identificirata njegova sinonima "Opened" in preveden v slovenščino "Odprto" in je privzet za nove entitete. Pogojev, kdaj je na voljo, nima definiranih.
- "2:{alias="Closed",alias.sl_SI="Zaprto"}:["":"1"] – Vrednost atributa »2« 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 ([glej poglavje Predloge](#)).

V povezavi z atributom se je potrebno zavedati tudi koncepta dedovanja predlog, saj 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 preddefiniranih 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 [Zmožnosti, odstavek Parameter atributa »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 kreirana 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

Atribut ob shranitvi entitete ne sme biti prazen. 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 kreirana 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 ([glej poglavje 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.

3.2.4 Poimenovanje

Vsak atribut je potrebno poimenovati s poljubnim unikatnim nizom UTF-8 znakov, dolgim do 256 zlogov 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:«. »imp:« so rezervirane in uporaba ni dovoljena. Poizkus ustvarjanja tako imenovanega atributa bo zavržen.
- 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, opsijski niz UTF-8 znakov, dolg do 512 zlogov 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 attribute 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

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

Uporaba: Razred, Zadeva, Dokument

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

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

Uporaba: Razred, Zadeva, Dokument

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, [Required pri izbrisanih entitetah]

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

Uporaba: Razred, Zadeva, Dokument

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

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

Uporaba: Dokument

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

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

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 definira. 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

Omejitve: Veljavne vrednosti:

- **1, alias "Opened"**: entiteto je dovoljeno spreminjati in pod njo kreirati podrejene entitete.
- **2, alias "Closed"**: entiteta ne dovoljuje spreminjana in kreiranja podrejenih entitet.

Prisotnost atributa je obvezna v vseh entitetah, ki so podrejene entitete razreda.

Uporaba: Razred, Zadeva, Dokument

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 Življenjski cikel](#)).

Atribut »sys:Creator«

Tip: DirectoryEntity

Lastnosti: Searchable, Public, Required, ReadOnly

Omejitve: Vrednost določa strežnik in ni spremenljiva. Vrednost vedno predstavlja registrirano entiteto iz imenika.

Uporaba: Razred, Zadeva, Dokument

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

Omejitve: Dovoljene vrednosti so identifikatorji registriranih entitet iz imenika.

Uporaba: Razred, Zadeva, Dokument

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, Required, PickList

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 "Normal"**: Običajna entiteta brez posebne pomembnosti.
- **5, 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, ...).

Prisotnost atributa obvezna v vseh entitetah, kjer je uporabljen.

Uporaba: Zadeva, Dokument

Opis: Atribut predstavlja pomembnost entitete v kontekstu lastnika arhiva.

Atribut »sys:Opened«

Tip: DateTime

Lastnosti: Searchable, Public, ReadOnly

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

Uporaba: Razred, Zadeva, Dokument

Opis: Vrednost predstavlja datum in čas ko je atribut »sys:Status« dotične entitete dobil vrednost »Opened« ([glej poglavje 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

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

Uporaba: Razred, Zadeva, Dokument

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

Omejitve: Vrednost določa strežnik in ni spremenljiva.

Uporaba: Dokument

Opis: Vrednost predstavlja vsakokratni dnevnik dogodkov samodejnih dejanj preverjanja strežnika ob shranitvi dokumenta. 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:MoveReason«

Tip: String200

Lastnosti: Searchable, MultiValue, ReadOnly, AppendOnly

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:MoveAgent«, »sys:MoveDateTime« in »sys:MoveClassificationCode« 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 Življenjski cikel entitete.](#)

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

Atribut »sys:MoveAgent«

Tip: DirectoryEntity

Lastnosti: Searchable, MultiValue, ReadOnly, AppendOnly

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. Istoležne vrednosti lahko povežemo z atributoma »sys:MoveReason«, »sys:MoveDateTime« in »sys:MoveClassificationCode« 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 Življenjski cikel entitete.](#) Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:MoveDateTime«

Tip: DateTime

Lastnosti: Searchable, MultiValue, ReadOnly, AppendOnly

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. Istoležne vrednosti lahko povežemo z atributoma »sys:MoveReason«, »sys:MoveAgent« in »sys:MoveClassificationCode« 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 Življenjski cikel entitete.](#) Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:MoveClassificationCode«

Tip: String200

Lastnosti: Searchable, MultiValue, ReadOnly, AppendOnly

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.

Istoležne vrednosti lahko povezujemo z atributoma »sys:MoveReason«, »sys:MoveAgent« in »sys:MoveDateTime« 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 Življenjski cikel entitete](#). Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys>DeleteReason«

Tip: String200

Lastnosti: Searchable, Public, Required, ReadOnly

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

Uporaba: Razred, Zadeva, Dokument

Opis: Vrednost predstavlja razlog za izbris entitete. Gre za vrednost, ki je posredovana s strani uporabnika ob izbrisu entitete. [Za več informacij glej poglavje Življenjski cikel entitete](#). Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys>DeleteAgent«

Tip: DirectoryEntity

Lastnosti: Searchable, Public, Required, ReadOnly

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 izbris entitete. Gre za vrednost, ki jo strežnik samodejno določi iz uporabniške seje, oziroma prijavnih podatkov uporabnika, ki je izbris izvedel. [Za več informacij glej poglavje Življenjski cikel entitete](#). Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys>DeleteDateTime«

Tip: DateTime

Lastnosti: Searchable, Public, Required, ReadOnly

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

Uporaba: Razred, Zadeva, Dokument

Opis: Vrednost predstavlja datum in čas, ko je izveden izbris entitete. Gre za vrednost, ki jo strežnik samodejno določi. [Za več informacij glej poglavje Življenjski cikel entitete.](#)

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

Atribut »sys>DeleteClassificationCode«

Tip: String200

Lastnosti: Searchable, Public, Required, ReadOnly

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

Uporaba: Razred, Zadeva, Dokument

Opis: Vrednost predstavlja klasifikacijsko oznako entitete v času izbrisa.

Gre za vrednost, ki jo strežnik samodejno zajame. [Za več informacij glej poglavje Življenjski cikel entitete.](#) Atribut nima vpliva na poslovno logiko strežnika pri operacijah z entitetami, služi le kot nosilec informacije.

Atribut »sys:SecurityClassChangeReason«

Tip: String200

Lastnosti: Searchable, MultiValue, ReadOnly, AppendOnly

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 povežemo z atributoma

»sys:SecurityClassChangeAgent«, »sys:SecurityClassChangeDateTime«,

»sys:SecurityClassChangeFrom« in »sys:SecurityClassChangeTo« 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:SecurityClassChangeAgent«

Tip: DirectoryEntity

Lastnosti: Searchable, MultiValue, ReadOnly, AppendOnly

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 stopnje tajnosti entitete. Gre za vrednost, ki jo strežnik samodejno določi iz uporabniške seje, oziroma iz prijavnih podatkov uporabnika, ki je stopnjo tajnosti vsakokrat izvedel. Istoležne vrednosti lahko povežemo z atributoma »sys:SecurityClassChangeReason«, »sys:SecurityClassChangeDateTime«, »sys:SecurityClassChangeFrom« in »sys:SecurityClassChangeTo« 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:SecurityClassChangeDateTime«

Tip: DateTime

Lastnosti: Searchable, MultiValue, ReadOnly, AppendOnly

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. Istoležne vrednosti lahko povežemo z atributoma »sys:SecurityClassChangeReason«, »sys:SecurityClassChangeAgent«, »sys:SecurityClassChangeFrom« in »sys:SecurityClassChangeTo« 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:SecurityClassChangeFrom«

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 definirati vrednost za "0", ki je prednastavljena 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:SecurityClass" pred njegovo spremembo. Istoležne vrednosti lahko povezujemo z atributoma »sys:SecurityClassChangeReason«, »sys:SecurityClassChangeAgent«, »sys:SecurityClassChangeDateTime« in »sys:SecurityClassChangeTo« 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:SecurityClassChangeTo«

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 definirati vrednost za »0«, ki je prednastavljena 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:SecurityClass« po njegovi spremembi. Istoležne vrednosti lahko povezujemo z atributoma »sys:SecurityClassChangeReason«, »sys:SecurityClassChangeAgent«, »sys:SecurityClassChangeDateTime« in »sys:SecurityClassChangeFrom« 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 preddefinirani in z njihovimi lastnostmi ni možno upravljati.

Atribut »eml:MessageId«

Tip: String100

Lastnosti: Searchable

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 »eml:Date«

Tip: DateTime

Lastnosti: Searchable, Required

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 »eml:From«

Tip: String200

Lastnosti: Searchable, Required

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 »eml:To«

Tip: String200

Lastnosti: Searchable, Required, MultiValue

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 »eml:ToCC«

Tip: String200

Lastnosti: Searchable, MultiValue

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 »eml:ToBCC«

Tip: String200

Lastnosti: Searchable, MultiValue

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 »eml:Subject«

Tip: String200

Lastnosti: Searchable, Required

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 »eml:Priority«

Tip: String20

Lastnosti: Searchable, PickList

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 »eml:Signed«

Tip: Bool

Lastnosti:

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. attribute 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 beležijo. 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 systemske 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 »prm:Identifier«

Tip: String100

Lastnosti: Searchable

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 »prm:Description«

Tip: String200

Lastnosti: Searchable, MultiValue

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 »prm:Status«

Tip: String20

Lastnosti: Searchable

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 »prm:CurrentLocation«, »prm:Custodian« in »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 »prm:StatusChange«.

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

Atribut »prm:StatusChange«

Tip: DateTime

Lastnosti: Searchable, ReadOnly

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 »prm:Status« in ni spremenljiva.

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

Atribut »prm:HomeLocation«

Tip: String100

Lastnosti: Searchable

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 »prm:CurrentLocation«

Tip: String100

Lastnosti: Searchable

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 »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 »prm:Custodian«

Tip: String100

Lastnosti: Searchable

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 »prm:Status« je »CheckedIn«), je to navadno skrbnik fizičnega gradiva. Če je zunanja (vrednost atributa »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 »prm:ReturnDue«

Tip: Date

Lastnosti: Searchable

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 »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 attribute prenesenega gradiva hraniti v t.i. sistemskih atributih, da zagotavljamo kontinuiteto življenjskega cikla hranjenega gradiva. Nekateri sistemski atributi tretjih sistemov se vnašajo v sistemske attribute 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« ali »Prenosa« določa prisotnost vloge »Transfer«. Atributi prenesenega gradiva nimajo vpliva na poslovno logiko strežnika v smislu operacij nad entitetami, služijo le kot nosilci informacij.

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

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

Atribut »trf:ClassificationCode«

Tip: String100

Lastnosti: ReadOnly

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

Atribut »trf:Imported«

Tip: DateTime

Lastnosti: ReadOnly

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

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

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).

Vsaka entiteto določa:

- 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 podzadeva
- 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 dobavnice pa v drugem)
- lastniku vsebovanih dokumentov (npr. kadrovski oddelek ima svoje dokumente v enem razredu, prodaja pa v drugem).

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

Zadeva

Zadeva predstavlja skupino entitet (podzadeve, 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 začneja z več debli (razredi na prvem nivoju) in vejami, ki jih predstavljajo posamezne entitete.

Končni razredi vsebujejo entitete vsebine (zadeva, podzadeva, dokument).

Strežnik IMiS®/ARChive Server ne omejuje globine hierarhije načrta razvrščanja gradiva. Pri določanju načrta razvrščanja gradiva je potrebno upoštevati spodaj naštetna 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 Dostopi](#)), razen kadar so dostopne pravice natančneje določene v samih podrejenih entitetah ([glej podpoglavje Eksplicitna dovoljenja ali prepovedi](#)). Istočasno hierarhija določa še stopnjo tajnosti (če ni eksplicitno določena; [glej poglavje Stopnje tajnosti](#)) ter obveznost statusa entitete (sistemski atribut »sys:Status«).

Tako je določitev statusa entitete obvezna v naslednjih primerih:

- za vse razrede
- za vse zadeve uvrščene podrazrede
- za vse dokumente uvrščene podrazrede.

Statusa ne določamo:

- podrejenim zadevam
- dokumentom uvrščenim v zadevo/podrejeno zadevo.

Te entitete podedujejo status iz svoje prve nadrejene entitete, ki določa status.

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.

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 attribute.

Hkrati so osnova za izdelavo specializiranih predlog entitet istega tipa, ki podedujejo atributno shemo.

Vsaka predloga vsebuje vgrajene in vnaprej določene sistemske attribute.

Ti so nujno potrebni za pravilno delovanje strežnika IMiS®/ARChive Server in jih ni mogoče spreminjati. Njihovi nazivi prihajajo iz naslovnega področja »int:«, »sys:«, »trf:«, ...

Vse ostale attribute lahko administrator poljubno dodaja in briše iz predloge, dokler ni kreirana prva entiteta. Potem so možnosti spremembe predloge omejene, omejitve so navedene v nadaljevanju.

Ob dodajanju atributa na predlogo administrator določi:

- ali je obvezen oziroma neobvezen
- ali ima lahko več kot eno vrednost
- ali lahko uporabnik spreminja že zapisan atribut oziroma, če dovoli samo dodajanje vsebine atributa brez brisanja že obstoječih vrednosti.

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 kreira 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:

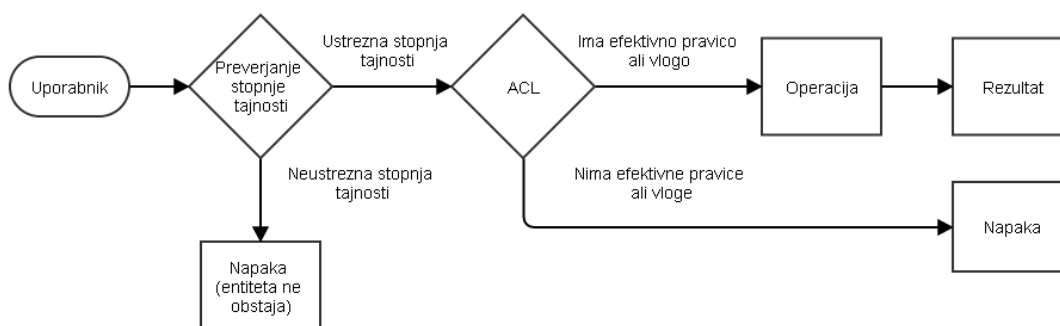
- celovitost (angl. Integrity): zagotavlja nespremenjenosti informacij
- zaupnost (angl. Confidentiality): zagotavlja, da so informacije dostopne samo pooblaščenim osebam
- razpoložljivost (angl. Availability): zagotavlja razpoložljivosti informacij pooblaščenim osebam.

Preverjanje dostopa lahko razdelimo na dva dela:

- preverjanje stopnje tajnosti (angl. Security Class)
- preverjanje dostopa do entitet.

Stopnja tajnosti je predpogoj, da lahko uporabnik vidi obstoj entitet. Zato mora imeti uporabnik najmanj enako (ali višjo) stopnjo tajnosti, kot jo imajo entitete, bodisi eksplicitno določene ali podedovane iz nadrejene entitete. V nasprotnem primeru je uporabniku vsakršno delo z entitetami onemogočeno. Če uporabniku stopnja tajnosti omogoča videti obstoj entitet, se preverja dostop do le-teh. Za preverjanje dostopov do entitet, strežnik uporablja IMiS®/ARChive Server koncept liste dostopnih pravic (angl. Access Control Lists - ACL). Za vsako operacijo na strežniku, ki jo izvaja avtoriziran uporabnik preveri ali jo ima le-ta 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

Gre za 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 manjšo ali enako stopnjo tajnosti na entiteti, kot jo ima nadrejena entiteta v hierarhiji (če nadrejena entiteta nima nastavljene stopnje tajnosti, potem se lahko nastavi poljubna 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 enako kot entiteta, ki ji dvigujemo stopnjo tajnosti.
- Pri spuščanju, se stopnja tajnosti spusti vsem podrejenim entitetam, ki imajo večje stopnje tajnosti kot vrednost, ki jo uporabnik nastavlja.

Primer 1: za primer vzemimo hierarhijo, ki jo predstavlja [slika v poglavju Dostopi – Efektivne pravice](#). Predpostavimo, da nobena od entitet nima eksplicitno nastavljene stopnje tajnosti (stopnja tajnosti vseh entitet je 0). Razredu nastavimo stopnjo tajnosti na 2 (Restricted). Zaradi dedovanja se vsem podentitetam nastavi stopnja tajnosti na 2.

Primer 2: zadevi iz prejšnjega primera hočemo dvigniti stopnjo tajnosti na 3 (Confidential). Ker ima nadrejeni razred stopnjo tajnosti 2, dvigovanje stopnje tajnosti ni mogoče.

Primer 3: zadevi iz prejšnjega primera spustimo stopnjo tajnosti na 1 (Unrestricted). Posledično se obema dokumentoma stopnja tajnosti spusti na stopnjo 1 zaradi dedovanja.

Primer 4: razredu dvignemo stopnjo tajnosti na 3 (Confidential). Zadevi in podrejenim dokumentom se stopnja tajnosti ne spremeni, saj se pri dvigovanju stopnje tajnosti upoštevajo samo entitete, ki imajo enako stopnjo tajnosti kot entiteta, ki ji stopnjo tajnosti spreminjamo.

Primer 5: zadevi dvignemo stopnjo tajnosti na 2 (Restricted). Posledično se tudi stopnja tajnosti dokumentov dvigne na 2. Nato razredu spustimo stopnjo tajnosti na 1 (Unrestricted). Posledično se stopnja tajnosti spusti tudi zadevi in dokumentom, saj se stopnja tajnosti spremeni vsem podrejenim entitetam, ki imajo višjo stopnjo tajnosti od vrednosti, ki se jo nastavlja.

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 lahko dodeli listo dostopnih pravic uporabniku ali uporabniški skupini. Dostopne pravice razdelimo v dve skupini:

- dostopne pravice za entitete
- specializirane dostopne pravice za attribute.

Vsaki skupini pravic lahko določimo eksplicitno dovoljenje (angl. Grant) ali prepoved (angl. Denial) in velja za posamezno pravico znotraj skupine.

V kombinaciji s podedovanimi pravicami s tem kontroliramo efektivne pravice (angl. Effective rights), ki določajo, ali ima uporabnik na strežniku pravico do zahtevane operacije.

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.
- **ChangeRights:** pravica spreminjanja liste dostopnih pravic (ACL).
Uporabniku je dovoljeno spreminjati dostopne pravice za entitete, metapodatke in vloge. Spreminjanje dostopnih pravic je mogoče samo, ko je entiteta odprta za urejanje, zato mora uporabnik imeti tudi pravice za urejanje entitete.
- **CreateSubentities:** pravica kreiranja novih podentitet.
Uporabnikom, ki je pravica dodeljena, je potrjen obstoj entitete, pod katero se kreira nova entiteta, ne glede na uporabnikovo pravico branja vsebine entitete in ne glede na globalno nastavitvev dostopa v primeru, ko uporabnik do entitete nima dostopa (potrditev obstoja entitete ali njeno skrivanje).
- **SetSecurityClass:** pravica spreminjanja stopnje tajnosti.
Uporabnikom, ki 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.

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

Operacija	Pravica						
	Read	Write	Delete	Move	Change-Rights	Create-Subentities	SetSecurity-Class
Odpiranje entitete v načinu za branje	✓						
Urejanje entitete	✓	✓					
Brisanje entitete		✓	✓				
Urejanje ACL	✓	✓			✓		
Re-klasifikacija (izvorna entiteta)	✓	✓		✓			
Re-klasifikacija (tarčna entiteta)						✓	
Kreiranje entitet (nadrejena entiteta)						✓	
Spreminjanje stopnje tajnosti		✓					✓

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 dodatno omejimo dostop in urejanje metapodatkov, ki niso javni. Tako lahko uporabniku s pravico branja na določeni entiteti preprečimo 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

Izjema pri nastavitvah dovoljenj ali prepovedi so javni 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 nastavitvev na strežniku.

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 nastavitvev
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 nastavitvev tudi pravico branja javnih metapodatkov.	✓	N/A

Tabela 5: Omejitev pravic vpogleda v javne metapodatke

3.3.5.2.4 Vloge

Vloge so skupek pravic, ki uporabniku omogočajo izvajanje določene operacije na strežniku. Določene so naslednje vloge:

- **AuditLogQuery:** vloga omogoča pridobivanje revizijske sledi
- **Transfer:** vloga omogoča izvoz in uvoz arhiviranih vsebin
- **Deletion:** vloga omogoča prikaz seznama entitet, ki so kandidati za izbris
- **Reports:** vloga omogoča prikaz sistemskih poročil o uvozih, izvozih, prenosih, izbrisih, ...
- **Import:** vloga omogoča uvoz vsebin.

3.3.5.2.5 Eksplicitna dovoljenja ali prepovedi

Z nastavljanjem eksplicitnih dovoljenj ali prepovedi upravljamo učinkovite 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.

Dovoljenja ali prepovedi brez časovne omejitve

Standardni način uporabe dovoljenj ali prepovedi. Takšna dovoljenja in prepovedi veljajo vedno, ne glede na to, kdaj se uporabljajo pri izračunu učinkovitih pravic.

Dovoljenja ali prepovedi z začetkom časovne omejitve

Dovoljenja ali prepovedi imajo določen začetek veljavnosti, nimajo pa določenega konca. Takšna dovoljenja in prepovedi se upoštevajo pri izračunu učinkovitih pravic v primeru, da je trenutni datum večji ali enak datumu začetka veljavnosti dovoljenja ali prepovedi.

Dovoljenja ali prepovedi s koncem časovne omejitve

Dovoljenja ali prepovedi imajo določen konec veljavnosti, nimajo pa določenega začetka. Takšna dovoljenja ali prepovedi se upoštevajo pri izračunu učinkovitih 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 učinkovitih 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.

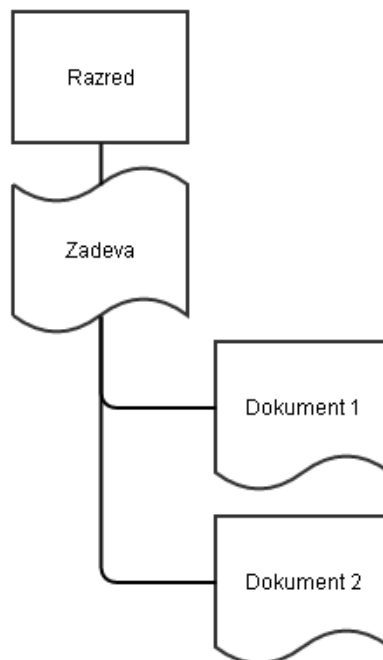
3.3.5.2.6 Efektivne pravice

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 učinkovite 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 učinkovite 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 učinkovitih 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 učinkovitih pravic, saj ima eksplicitna prepoved pisanja prednost pred eksplicitnim dovoljenjem. Posledično ima uporabnik enake učinkovite 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 učinkovito pravico urejanja ACL na zadevi ter na dokumentu »Dokument 2«. Na dokumentu »Dokument 1« uporabnik še vedno ne more urejati ACL-a, 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 1.4.2014 (brez konca).

Uporabnik tako izgubi efektivne pravice brisanja metapodatkov z dnem 1.4.2014, pred tem datum pa uporabnik lahko normalno briše metapodatke.

Če nastavimo konec časovne omejitve na 1.4.2014 (brez začetka), potem uporabnik do 1.4.2014 nima efektivne pravice brisanja metapodatkov, po 1.4.2014 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 1.4.2014 do 15.4.2014.
- V ACL zadeve nastavimo eksplicitno dovoljenje pisanja in kreiranja podrejenih entitet, ki je časovno omejeno od 6.4.2014 do 12.4.2014.

- V ACL dokumenta »Dokument 1« nastavimo eksplicitno prepoved pisanja, ki je časovno omejeno od 8.4.2014 do 12.4.2014.

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

- Med 1.4.2014 in 15.4.2014 ima uporabnik pravico odpiranja vseh entitet v hierarhiji.
- Med 6.4.2014 in 12.4.2014 ima uporabnik pravico urejanja zadeve in dokumenta »Dokument 2« ter kreiranja podrejenih entitet v zadevi.
- Urejanje dokumenta »Dokument 1« je dovoljeno uporabniku med 6.4.2014 in 8.4.2014, saj z 8.4.2014 začne veljati eksplicitna prepoved pisanja dokumenta, ki ima prednost pri računanju efektivnih pravic in posledično izniči efektivno pravico pisanja.
- Pred 1.4.2014 in po 15.4.2014 uporabnik nima pravice odpiranja in urejanja entitet v hierarhiji. Še vedno pa ima pravico branja javnih metapodatkov, če mu globalna varnostna nastavitvev 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 zlogov (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 AES-256 enkripcijo, 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 zlogov in $1 : 8,636 \times 10^{70}$ pri identifikatorjih dolžine 32 zlogov.

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 zlogov in 64 znakov za identifikator velikosti 32 zlogov.
- 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 zlogov in 43 znakov za identifikator velikosti 32 zlogov.
- 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 zlogov in dolžino 40 znakov za identifikatorje velikosti 32 zlogov.

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 takšne 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 kode 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 polne klasifikacijske oznake. 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

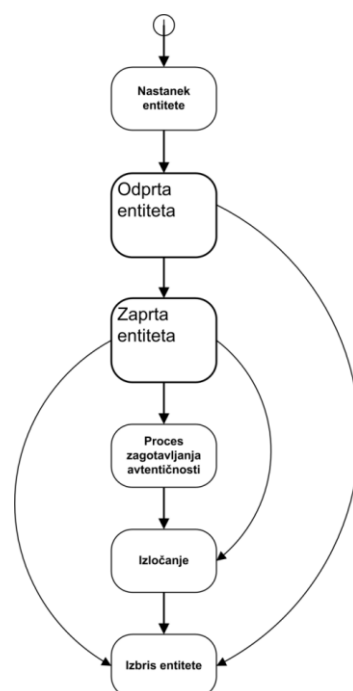
Ob nastanku entiteta vedno dobi status »Opened« (odprto), kar omogoča operacije, ki rezultirajo v njeni spremembi, oziroma dodajanju njenih podrejenih entitet.

Uporabnik spremeni status v »Closed« šele, ko je povsem prepričan, da entitete ne bo več potrebno spreminjati in pod njo ne uvrščati novih entitet.

Entiteti namreč ni več mogoče povrniti statusa »Opened« potem, ko se ji je status enkrat spremenil v »Closed«.

Na zaprti entiteti je mogoče (ni pa obvezno) sprožiti postopek zagotavljanja avtentičnosti ([glej poglavje 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, vendar je brisanje pred postopkom izločanja 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). Za spremembo statusa iz »Opened« v »Closed« je potrebno entiteto odpreti v načinu za pisanje in izvesti spremembo atributa »sys:Status«.

Opozorilo: Sprememba statusa je nepovratna operacija, ki lahko pomeni tudi začetek roka za odbiranje in izločanje.

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.

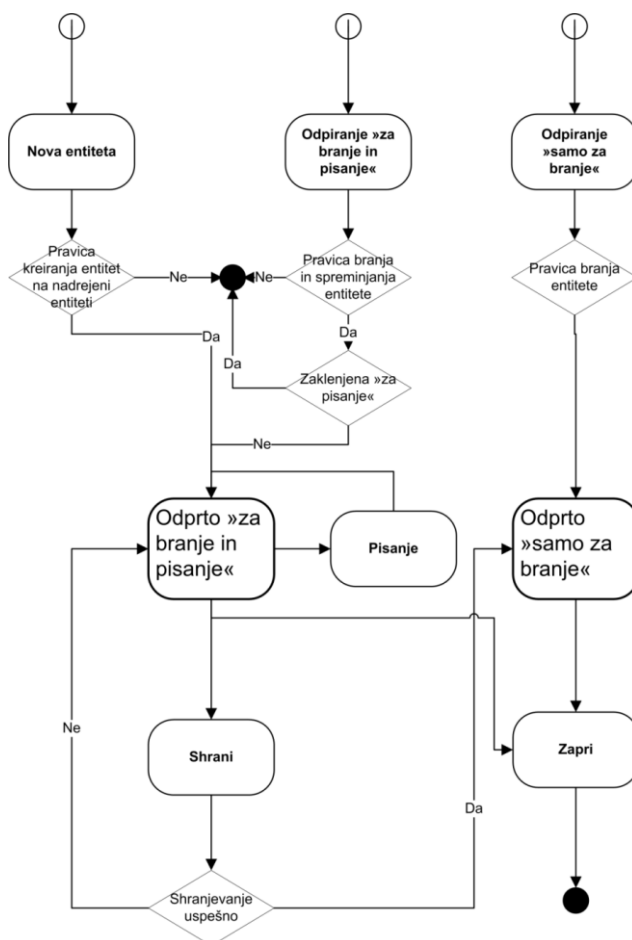
V kolikor je strežnik IMiS®/ARChive Server nastavljen za zagotavljanje avtentičnosti gradiva, strežnik v seznam entitet, primernih za generiranje dokaznih elementov avtentičnosti, take entitete vključi.

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. 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

Neshranjena 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 kreiranja novih podrejenih entitet na nadrejeni entiteti
- status nadrejene entitete ne sme biti »Closed«
- 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«

Zahtevak 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 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 kreiranja, zadnje spremembe, efektivne pravice uporabnika nad entiteto, ...) in 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).

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 attribute 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 Zmožnosti](#)).

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 Instanca entitete](#)).

Zahtevek mora vsebovati vsaj:

- referenco na instanco entitete.

Obenem se preveri drugi del veljavnost vrednosti atributov ([glej poglavje Zmožnosti](#)).

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.

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 njihovem celotnem življenjskem 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 neshranjena 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 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 pogoje:

- pravica dostopa do entitete iz naslova stopnje tajnosti
- pravica uporabnika do spremembe stopnje tajnosti na dotični entiteti
- pravica so 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 attribute 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. Zahtevk 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 katerekoli 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:Logs^C=Deletion«, kjer se zbirajo vsi izbrisi.

Posledično se entiteta vsem uporabnikom »skrije« in postane dostopna le uporabnikom z vlogo »Deletion«.

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

- »sys:Title«: naslov entitete.
- »sys:Description«: opis entitete (prej neobvezen postane v izbrisanih entitetah obvezen). V primeru, da se opis entitete posreduje v času izbrisa, pride pred izbrisom do odpiranja entitete v načinu za spreminjanje, popravek atributa in shranitev entitete, vse to je zabeleženo v revizijski sledi.

Dodatno strežnik samodejno doda naslednje sistemske attribute 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>DeleteReason«: razlog za izbris posreduje uporabnik kot obvezen podatek ob izbrisu.
- »sys>DeleteAgent«: uporabnik, ki je izbris izvedel.
- »sys>DeleteDateTime«: datum in čas izbrisa.
- »sys>DeleteClassificationCode«: klasifikacijska oznaka entitete pred izbrisom.

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.

3.3.8 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 katerimkoli dokumentom, zbirko ali načrtom 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 dogodkih 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
- tip 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 po principu WORM (angl. Write Once, Read Many) samodejno beležijo:

Nova entiteta [1]

Dogodek se zapiše v revizijsko sled vsakokrat, ko uporabnik ustvari novo entiteto ([glej poglavje 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 sporočilu 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 Odpiranje entitete za »branje in pisanje« ali »samo za branje«](#)).

V sporočilu 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 Odpiranje entitete za »branje in pisanje« ali »samo za branje«](#)).

V sporočilu 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 Shranitev entitete](#)).

V sporočilu 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 Premik](#)).

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

V sporočilu 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 *Brisanje*](#)). Obvezni podatek o razlogu za izbris se poleg zabeleženja v za to namenjeni atribut zabeleži tudi v revizijski sledi.

V sporočilu 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 sporočilu revizijske sledi se zabeleži:

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

Sprememba nastavitvev revizijske sledi [8]

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

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

V sporočilu revizijske sledi se zabeleži (primer):

- »Global audit log enabled setting changed to on*«.
- »Global required audit log parameters settings changed. UserName: on*
ComputerName: on, Reason: on«.
- »Global audit log events settings changed. Entity.Create: on, Entity.OpenRO: on,
Entity.OpenRW: on, Entity.Save: on, Entity.Move: on, Entity.Delete: on,
Entity.PropertyChange: on*, Entity.PhysicalRecordChange: on*, Entity.ACL: on*,
Entity.SecClass: on*, AuditLog.Query: on*, Directory.GroupMembership: on*,
Directory.EntityAuthData: on*, Directory.EntityIdentity: on*,
Directory.EntityState: 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 Spreminjanje vsebine entitete](#) in [poglavje Shranitev entitete](#)). Atributi upravljanja s fizičnim gradivom so zavedeni v svojem dogodku in ta dogodek ne beleži.

V sporočilu 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 Spreminjanje vsebine entitete](#) in [poglavje 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 kak 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 sporočilu 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 članstva v skupini [11]

Dogodek se zapiše v revizijsko sled ob spremembi članstva neke skupine iz imenika.

Zapis se doda v revizijsko sled vsakokrat, ko administrator imenika spremeni (briše ali doda) uporabnike v skupini.

V sporočilu revizijske se zabeleži:

- enolična oznaka entitete
- spisec dodanih uporabnikov
- spisec odstranjenih uporabnikov.

Sprememba avtentikacijskih podatkov [12]

Dogodek se zapiše v revizijsko sled ob spremembi avtentikacijskih podatkov uporabnika. Zapis se doda v revizijsko sled vsakokrat, ko se uporabniku spremenijo avtentikacijski podatki, ali s strani uporabnika samega, ali s strani administratorja imenika (primer: izguba gesla s strani uporabnika).

V sporočilu revizijske sledi se zabeleži:

- enolična oznaka entitete
- spremenjeni avtentikacijski podatki z naslednjimi atributi:
 - tip uporabnika – uporabnik, skupina (angl. Type)
 - varnostni nivo (angl. Security Level)
 - dovoljene ure (angl. Logon Hours)
 - geslo (angl. Authentication Data).

Spremenjeni osebni podatki uporabnika [13]

Dogodek se zapiše v revizijsko sled ob spremembi evidenčnih podatkov uporabnika (ime, priimek, opis, ...). Zapis se doda v revizijsko sled vsakokrat, ko se spremeni katerikoli od uporabnikovih evidenčnih podatkov, shranjenih v imeniku.

V sporočilu revizijske sledi se zabeleži:

- enolična oznaka entitete
- spremenjeni evidenčni podatki:
 - ime
 - priimek
 - opis.

Spremenjeno stanje uporabnika [14]

Dogodek se zapiše v revizijsko sled ob aktivaciji, deaktivaciji ali »izbrisu« uporabnika iz imenika. Zapis se doda v revizijsko sled vsakokrat, ko pride do spremembe stanja uporabniškega računa.

V sporočilu revizijske se zabeleži:

- enolična oznaka entitete
- tip spremembe statusa uporabnika:
 - nov, aktiviran
 - nov, deaktiviran
 - izbrisan
 - deaktiviran
 - aktiviran.

Sprememba atributov upravljanja s fizičnim gradivom [15]

Dogodek se zapiše v revizijsko sled ob shranjevanju entitete ([glej poglavje Spreminjanje vsebine entitete](#) in [poglavje Shranitev entitete](#)).

Zapiše se samo v primeru, ko je bil spremenjen vsaj en atribut upravljanja s fizičnim gradivom.

V sporočilu 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[16]

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

V sporočilu revizijske sledi se zabeleži:

- stara stopnja tajnosti
- nova stopnja tajnosti
- komentar.

3.3.8.3 Pravica do vpogledov 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 »AuditLogQuery«. Rezultat vpogleda je poročilo v XML datoteki, ki je [opisano v poglavju 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 uporabljeni za dostop do sistema
- seznam šifriranih enoličnih identifikatorjev entitete.

V primeru, da so iskalni parametri premalo selektivni in bi bilo prikazanih preveč rezultatov, strežnik vrne v odgovoru napako. V tem primeru je priporočljivo ponoviti poizvedbo z bolj natančno določenimi parametri iskanja. Najlažje omejimo razpon rezultatov s čim manjšim časovnim obdobjem dogodkov.

Iskalne parametre lahko med seboj tudi kombiniramo, vendar obstajajo omejitve.

Ne glede na ostale parametre lahko vedno iščemo po datumu poizvedbe in po seznamu šifriranih enoličnih identifikatorjev entitete.

Dodatno lahko izberemo 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 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 pooblaščenih 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

```
<?xml version="1.0" encoding="UTF-8"?>
<auditlog.query.resultset xsi: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.-->
    <session id="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"/>
    <session id="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"/>
```

```

    <session id="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-->
<event seq="0" sessionId="23" type="2" dateTime="2014-04-24T10:01:47Z"
classificationCode="C=01" context="0D=R}rSc^&lt;N3ZQRdCGJqtf&amp;z2iB`~aWgCI
+MeIYx"/>
<event seq="1" sessionId="3" type="1" dateTime="2014-04-23T14:51:43Z"
classificationCode="C=01" context="0D=R}rSc^&lt;N3ZQRdCGJqtf&amp;z2iB`~aWgCI
+MeIYx"/>
<event seq="2" sessionId="3" type="4" dateTime="2014-04-23T14:51:43Z"
classificationCode="C=01" context="0D=R}rSc^&lt;N3ZQRdCGJqtf&amp;z2iB`~aWgCI
+MeIYx"/>
<event seq="3" sessionId="3" type="10" dateTime="2014-04-23T14:51:43Z"
classificationCode="C=01" context="0D=R}rSc^&lt;N3ZQRdCGJqtf&amp;z2iB`~aWgCI
+MeIYx" message="Values for the following properties have changed: sys:Creator,
sys:Opened, sys:Owner, sys:Status, sys:Title"/>
<event seq="4" sessionId="4" type="2" dateTime="2014-04-23T14:54:26Z"
classificationCode="C=01" context="0D=R}rSc^&lt;N3ZQRdCGJqtf&amp;z2iB`~aWgCI
+MeIYx"/>
</events>
</auditlog.query.resultset>

```

XML datoteko sestavljata 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
- tip dogodka (type)
- čas dogodka (dateTime)
- klasifikacijska oznaka pripadajočega objekta (classificationCode)
- šifrirani enolični identifikator entitete (context)
- sporočilo/razlog za dejanje (message).

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 e-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 tipa 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 unikatni 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 klasifikacijskih oznakah. Tega določi administrator arhiva v nastavitvah strežnika (primer: »/«).

Strežnik IMiS®/ARChive Server samodejno določa klasifikacijske oznake ob dodajanju novih entitet vseh tipov ([glej poglavje 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 praviloma biti 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 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 podrazredov)
- 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 katerikoli nivo je v arhiv mogoče dodajati kadarkoli. Novim razredom bodo klasifikacijske oznake dodeljene samodejno ali ročno, odvisno od nastavitve 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 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 kateregakoli 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 spreminjanja entitete, ki se premika ([glej podpoglavje Dostopne pravice za entiteto](#)).
- Uporabnik mora imeti pravico premika entitete, ki se premika ([glej podpoglavje Dostopne pravice za entiteto](#)).
- Uporabnik mora imeti pravico ustvarjanja novih entitet na mestu, kamor premika izbrano entiteto premika ([glej podpoglavje 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.

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, 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 Sistemski atributi](#).

Za vse premaknjene entitete se v revizijsko sled zabeleži dogodek [opisan v poglavju 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, 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.

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 !=).

Primeri: enostavni pogoji za iskanje po metapodatkih:

```
[Znesek]>=231.5
[Avtor]="Janez Novak"
[ZacetekSeje]<=2014-05-07T11:05+02:00
[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 zavutih oklepajih: { "Janez Novak" }. Znotraj zavutih 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 zavutih oklepajev, če je to le mogoče.

Primer: Izraz: **[Povrsina] >= 12 AND {"Janez"} AND {"Novak"}**

bo na primer vrnil enake rezultate kot izraz:

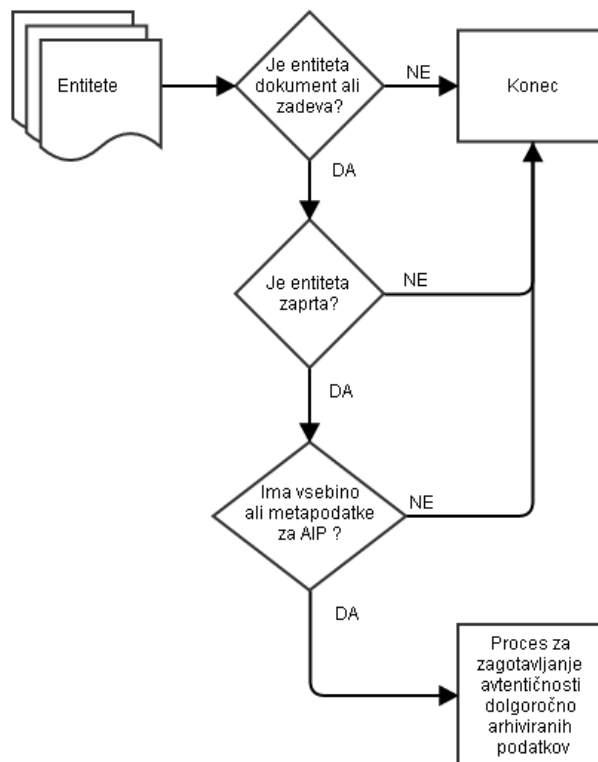
[Povrsina] >= 12 and {"Janez" AND "Novak"}

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 proces se zato vključijo samo entitete, ki so zaprte in so obenem zadeve ali dokumenti. 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-ja in posledično entitete dokazuje pripadajoča sintaksa evidenčnih podatkov (ERS), ki se ustvari v postopku zagotavljanja avtentičnosti dolgoročno arhiviranih podatkov. Spodnja slika prikazuje postopek umeščanja zaprtih zadev ali dokumentov, 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 v vhodnih podatkih 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.

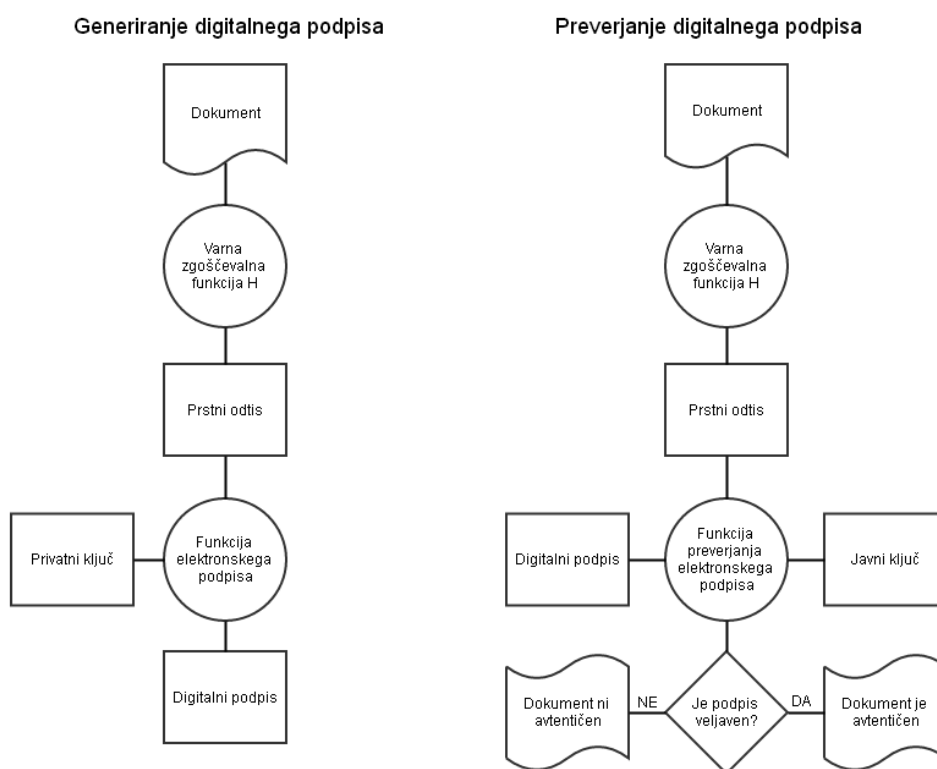


Slika 8: Delovanje zgoščevalne funkcije

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:

- Generiranje ključev (generirata se privatni in javni ključ, ki sta ključna za generiranje in preverjanje podpisa).
- Algoritem za podpisovanje (na podlagi privatnega ključa ter dokumenta se generira 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, ki je 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 e-podpisati vsebino. Na ta način časovni žig zagotavlja avtentičnost vsebine tudi, ko so digitalna potrdila podpisnikov že pretečena.

Vsi zgoraj naštetni 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:

- ETSI standardi
- 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
- odstranjevanje digitalnih potrdil
- iskanje digitalnih potrdil
- iskanje seznamov preklicanih digitalnih potrdil.

Za hitrejšo 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.

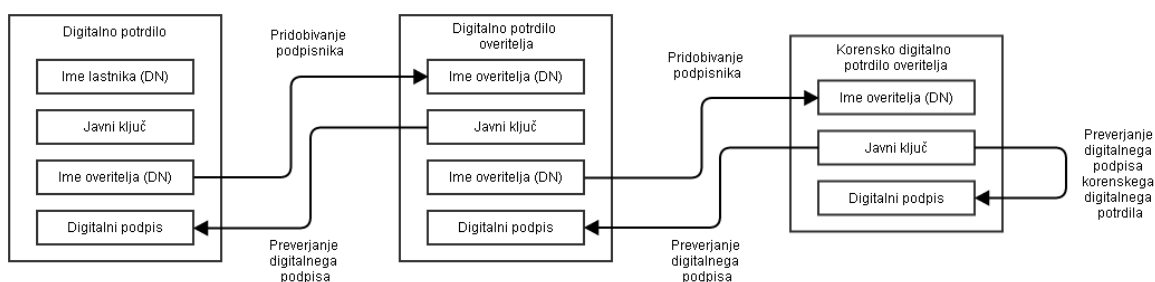
3.6.3.1 Veriga digitalnih potrdil

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

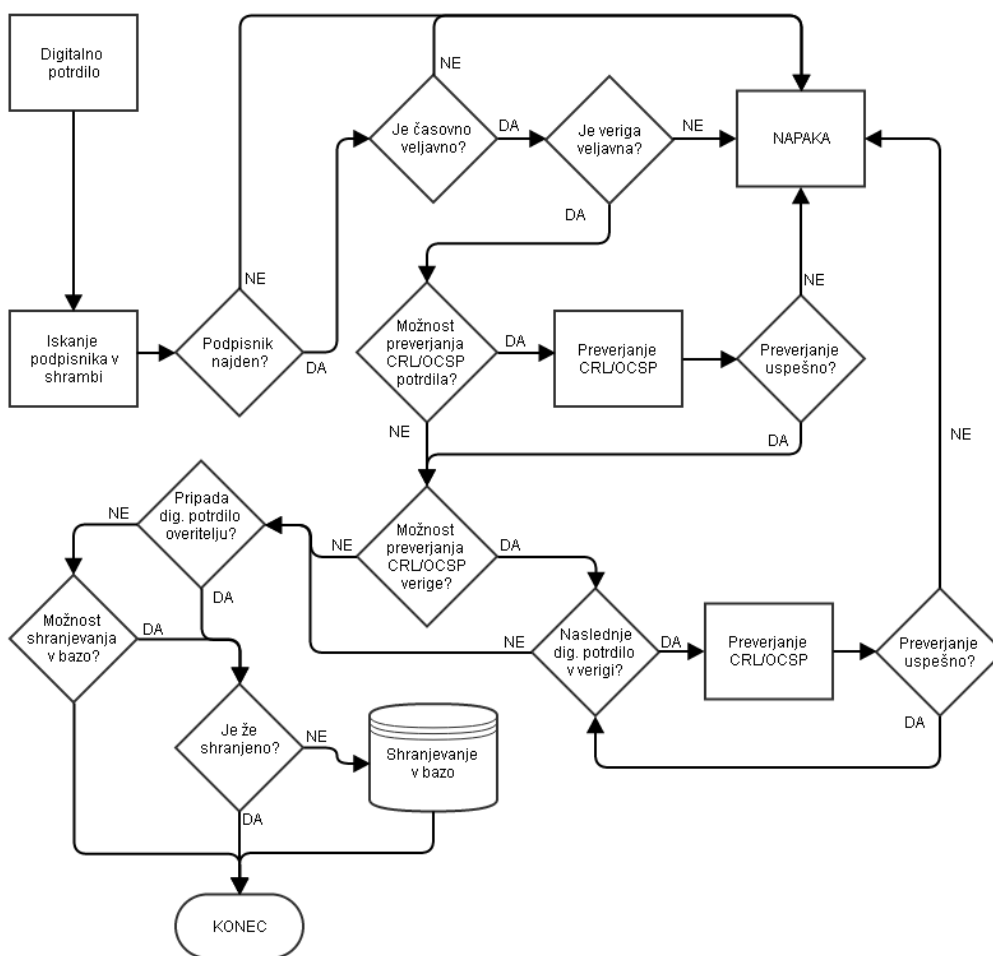
3.6.3.2 Dodajanje digitalnih potrdil

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.
- 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, ki tvorijo verigo zaupanja.



Slika 11: Postopek dodajanja digitalnega potrdila v shrambo

3.6.3.3 Odstranjevanje digitalnih potrdil

Odstranjevanje digitalnih potrdil iz shrambe je možno le, če digitalno potrdilo ni podpisnik nobenemu drugemu digitalnemu potrdilu. Obenem se mora digitalno potrdilo, ki ga hočemo odstraniti nahajati na koncu verige. Ob uspešni odstranitvi se digitalno potrdilo odstrani iz podatkovne baze in predpomnilnika.

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 ERS](#) in [poglavje 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
- drugo.

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>).

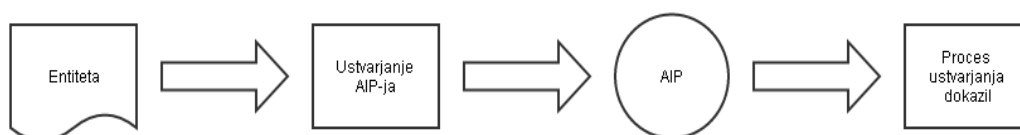
3.6.4 ERS

Sintaksa evidenčnih podatkov (angl. Evidence Record Syntax - ERS) je standard, ki opisuje sistem za zagotavljanje avtentičnosti dolgoročno arhiviranega gradiva. V strežniku IMiS®/ARChive Server je izveden ERS v XML formatu po specifikaciji RFC 6283 (<https://tools.ietf.org/html/rfc6283>).

Ključna procesa za dolgoročno zagotavljanje avtentičnosti arhiviranega gradiva sta:

- proces ustvarjanja dokazil
- proces podaljševanja dokazil.

Pred pričetkom procesa 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 za vsako entiteto, ki ustreza predpogojem za zagotavljanje avtentičnosti dolgoročno shranjenih podatkov ([glej poglavje Predpogoji](#)).



Slika 12: Ustvarjanje AIP-ja za proces 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 jih je potrebno podaljševati (generiranje novega elektronskega podpisa ali časovnega žiga). S časom se varnost algoritma za ustvarjanje prstnega odtisa zmanjša, saj se poveča verjetnost sovpadanja zgoščenih vrednosti.

Zato je potrebno algoritem nadomestiti z novejšim. To imenujemo tudi proces podaljševanja dokazil. S tem zagotovimo, da dokazila ne izgubijo zanesljivosti, kljub slabljenju matematičnih algoritmov, ki jih ustvarjajo. Ustvarjena dokazila je potrebno periodično preverjati ter jih pred izgubo zanesljivosti podaljšati, kot je prikazano na spodnji sliki.



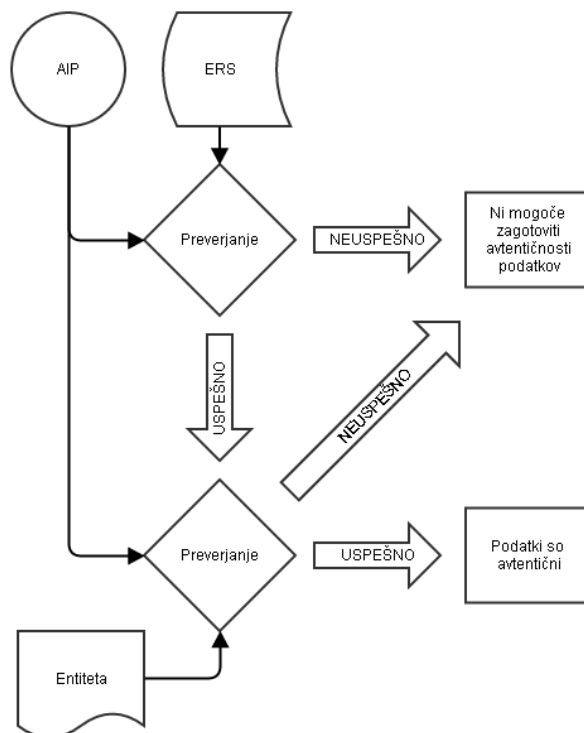
Slika 13: Podaljševanje zanesljivosti dokazil

Sintaksa evidenčnih podatkov predpisuje kako naj se ustvarjajo in podaljšujejo dokazila. Istočasno predpisuje, kako naj bodo dokazila strukturirana, da bodo nedvoumno dokazovala avtentičnost arhiviranega gradiva od trenutka, ko je to vstopilo v proces dolgoročnega zagotavljanja avtentičnosti.

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.



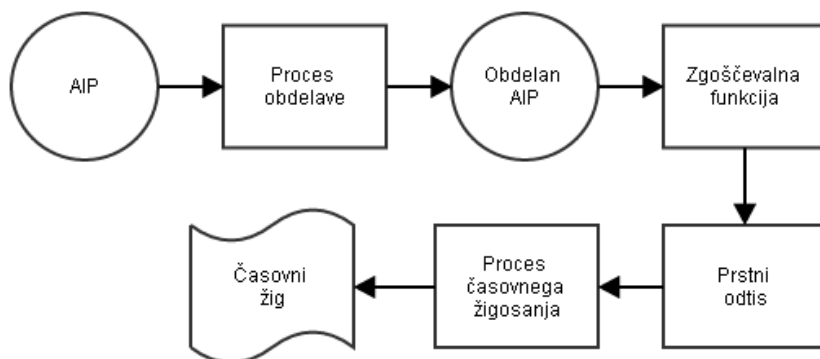
Slika 14: Primer preverjanja avtentičnosti podatkov

3.6.4.1 Proces ustvarjanja dokazil

Za vsako entiteto, ki pride v proces zagotavljanja avtentičnosti dolgoročno arhiviranih podatkov se ustvari arhivski informacijski paket (AIP). Tak paket se obdela glede na informacije, ki se nahajajo v glavi paketa ([glej poglavje 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.



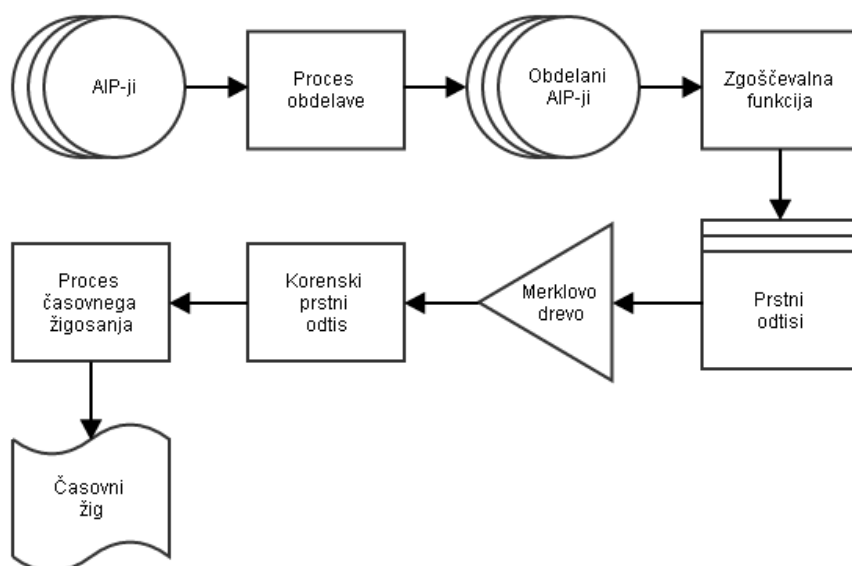
Slika 15: Postopek časovnega žigosanja posameznega AIP-ja

Ker je proces 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 vrši po naslednjih korakih:

- Za vse entitete, za katere je potrebno ustvariti dokazila o verodostojnosti se ustvarijo AIP.
- Vse ustvarjene AIP se obdelajo, 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 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 pa racionaliziramo proces 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 Proces 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:

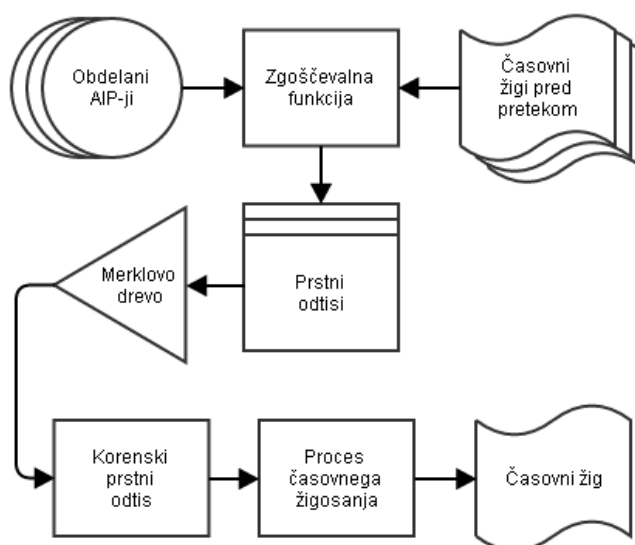
- enostavno podaljševanje
- kompleksno podaljševanje.

3.6.4.2.1 Enostavno podaljševanje

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

Zaradi neracionalnosti podaljševanja posameznih časovnih žigov, se večinoma za podaljševanje uporablja Merklovo drevo (enak postopek kot pri paketni obdelavi AIP). Lahko pa združimo proces enostavnega podaljševanja z procesom ustvarjanja dokazil. Tako lahko uporabimo Merklovo drevo, ki mu skozi proces zgoščevanja dodamo prstne odtise AIP in časovnih žigov pred potekom veljavnosti.

Tako lahko s časovnim žigosanjem korenkega prstnega odtisa Merklovega drevesa ustvarimo dokazilo za obstoj AIP ter hkrati podaljšamo veljavnost časovnih žigov pred pretekom.



Slika 17: Postopek enostavnega podaljševanja v kombinaciji z ustvarjanjem dokazil

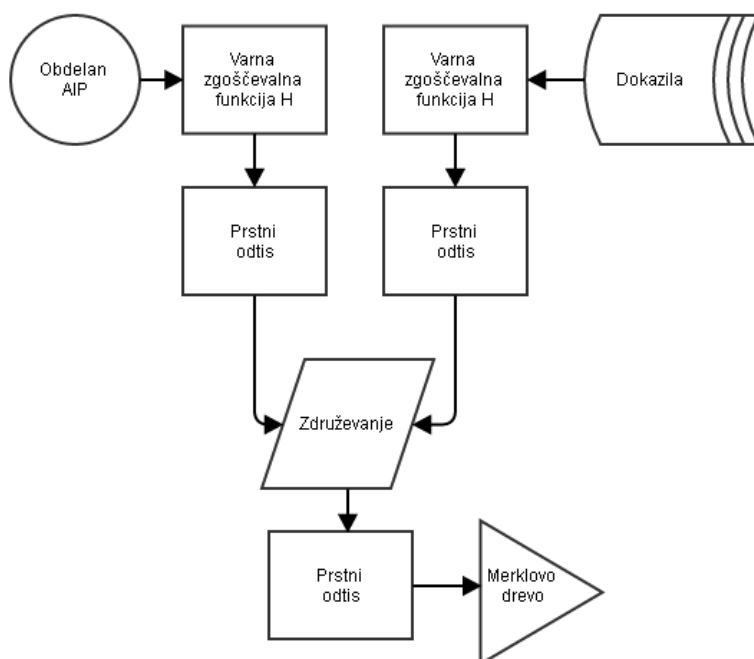
3.6.4.2.2 Kompleksno podaljševanje

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 Merklovo drevo](#)).
5. Za vsak arhivski informacijski paket (AIP), za katerega je bil v procesu ustvarjanja dokazil uporabljen šibek algoritem je potrebno izvesti operacije, našteje 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 nova zgoščevalna funkcija, uporabljena pri zadnjem kompleksnem podaljševanju za varno.



Slika 18: Del procesa 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 odzemanje 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 korenškega 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 proces gradnje vozlišč od končnih vozlišč do korenškega 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 ($H3 > H4$).

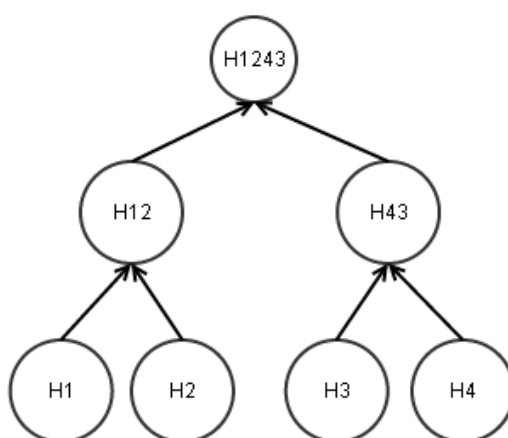
Na podlagi teh relacij izračunamo novi vozlišči H12 in H43 po naslednjih formulah:

- $H12 = \text{HASH}(H1 || H2)$ – ker velja $H1 < H2$, sortiranje ohrani vrstni red, zato oba prstna odtisa samo združimo (operacija $||$)
- $H43 = \text{HASH}(H4 || H3)$ – 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: $H1243 = \text{HASH}(H12 || H43)$.

Ker je H1243 zadnje vozlišče v drevesu pomeni, da je drevo zgrajeno, vrednost vozlišča pa je korenski prstni odtis.



Slika 19: Primer Merklevega drevesa

3.6.4.3.2 Reducirano Merklevo drevo

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

- Izberemo končno vozlišče, za katerega hočemo reducirano Merklevo 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 korenkega vozlišča). Vsa dobljena vozlišča razvrstimo naraščajoče.
- Neposredno nadrejenih vozlišč ne vključujemo v reducirano drevo, saj se jih da izračunati po pravilo gradnje Merklevega drevesa.

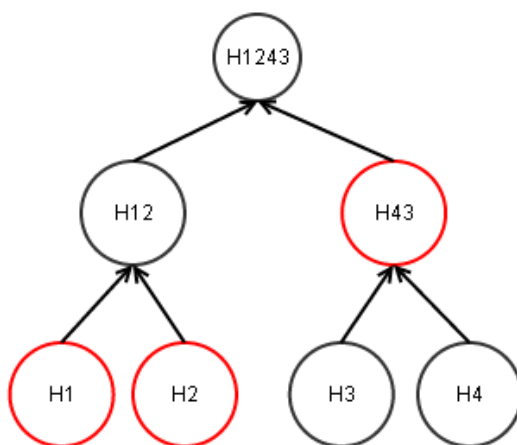
Primer: Reducirano Merklovo drevo, ki smo ga uporabili pri gradnji drevesa.

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 vozlišča 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 korenkega prstnega odtisa

Korenski prstni odtis se iz reduciranega drevesa izračuna na naslednji način:

$$H1243 = \text{HASH}(\text{HASH}(H1 \parallel H2) \parallel H43).$$

Postopek izračuna je naslednji:

- $H12 = \text{HASH}(H1 \parallel H2)$ – izračun vmesnega vozlišča H12
- $\text{SORT}(H12, H43) = [H12, H43]$ – rezultat sortiranja H12 in H43 ($H12 < H43$)
- $H1243 = \text{HASH}(H12 \parallel H43)$ – izračun korenkega 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:

$H1243 = \text{HASH}(H12 || \text{HASH}(H4 || H3))$.

Postopek je naslednji:

- $H43 = \text{HASH}(H4 || H3)$ – izračun vmesnega vozlišča H43
- $\text{SORT}(H43, H12) = [H12, H43]$ – rezultat sortiranja H43 in H12 ($H12 < H43$)
- $H1243 = \text{HASH}(H12 || H43)$ – izračun korenskega prstnega odtisa.

3.6.4.4 Sintaksa

Kot je bilo že omenjeno ima strežnik IMiS®/ARChive Server izveden ERS v XML obliki. Korenski element XML-a je etiketa »EvidenceRecord«, ki vsebuje elemente iz spodnje tabele.

Ime XML elementa	Tip XML elementa	Obveznost
Version	atribut	DA
ArchiveTimeStampSequence	etiketa	DA

Tabela 6: 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 Proces ustvarjanja dokazil](#) in [poglavje Proces podaljševanja dokazil](#)).

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

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

```
<EvidenceRecord xmlns="http://www.setcce.org/schemas/ers"
Version="1.0">
  <ArchiveTimeStampSequence>
    <ArchiveTimeStampChain Order="1">...</ArchiveTimeStampChain>
    <ArchiveTimeStampChain Order="2">...</ArchiveTimeStampChain>
  </ArchiveTimeStampSequence>
</EvidenceRecord>
```

3.6.4.4.1 Etiketa »ArchiveTimeStampChain«

Etiketa vsebuje verigo arhivskih časovnih žigov, ki si bili ustvarjeni v procesih ustvarjanja in procesu 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 7: 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 proces kompleksnega podaljševanja se ustvari nova veriga arhivskih časovnih žigov, z vrednostjo atributa »2«, »3« itn. ([glej poglavje Enostavno in 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.

3.6.4.4.1.1 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 8: 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 procesu ustvarjanja dokazil, vsi ostali arhivski časovni žigi pa v procesu enostavnega podaljševanja.

3.6.4.4.1.1.1 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 korenškega vozlišča lahko izračuna. Elementi etikete »Sequence« so opisani v tabeli 9.

Ime XML elementa	Tip XML elementa	Obveznost
Order	atribut	DA
DigestValue	etiketa	DA

Tabela 9: XML elementi etikete »Sequence«

Atribut »Order«

Vrednost atributa predstavlja nivo reduciranega Merklovega drevesa v obratnem vrstnem redu, kot je določena globina drevesa ([glej poglavje 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 korenškega vozlišča.

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

3.6.4.4.1.2 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 10: 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-a.
»RFC3161«	Časovni žig je narejen po standardu RFC3161. Vsebina TimeStampToken-a je Base64 kodirana zaradi kompatibilnosti z XML-jem.

Tabela 11: 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 12: 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+/oVEs6CjRHl3UNL1vk4WcsEkA=</DigestValue>
      <DigestValue>EBEhfDzZh9+rfb1Kaqe65o7TTok=</DigestValue>
    </Sequence>
    <Sequence Order="2">
      <DigestValue>fBuhe8txb0OmNt27uvYupJTTrgBQ=</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">MIISKTCERC...
  </CryptographicInformation>
  </CryptographicInformationList>
</TimeStamp>
</ArchiveTimeStamp>

```

3.6.4.5 Enostavno podaljševanje

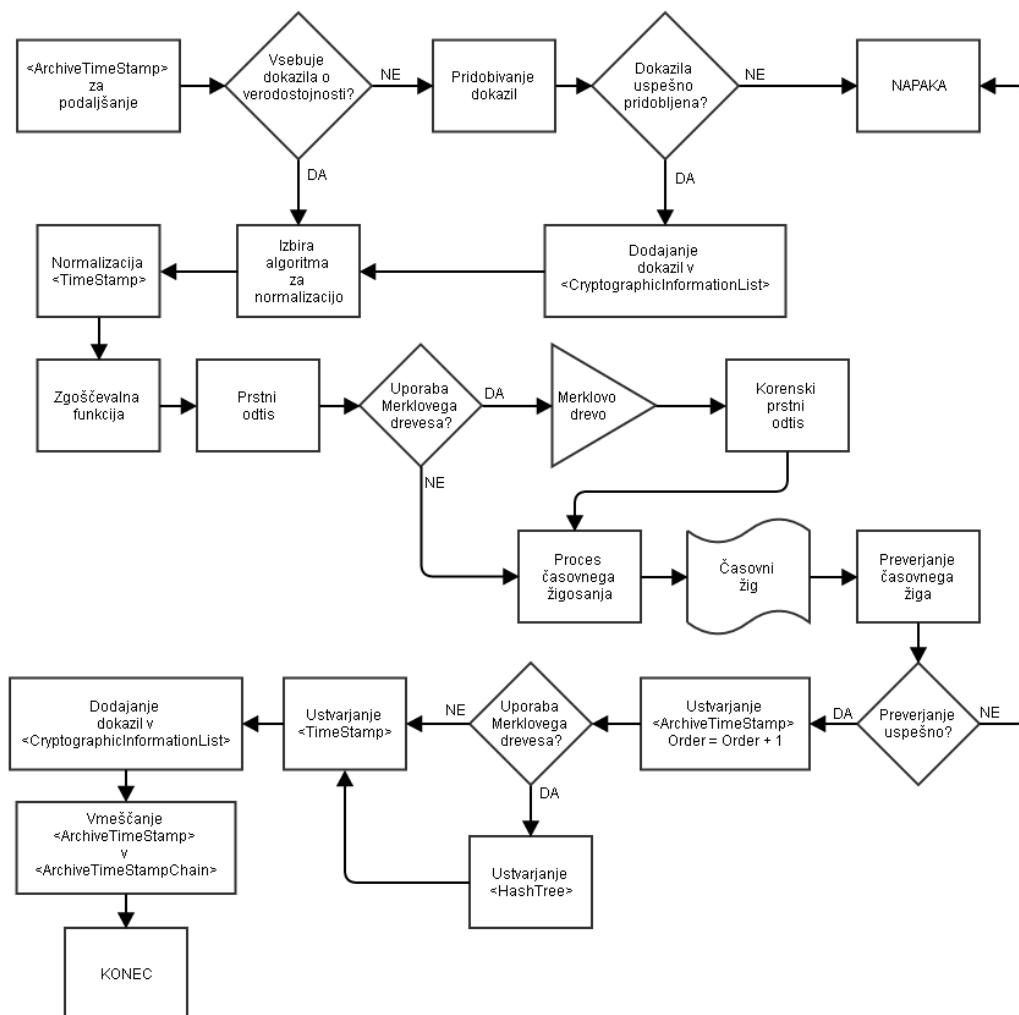
Pri enostavnem podaljševanju se zmeraj podaljša zadnji arhivski časovni žig v zadnji verigi arhivskih časovnih žigov.

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 procesa 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 Proces podaljševanja dokazil](#)).
5. Preverimo časovni žig ([glej poglavje 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 časovni 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 procesu 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

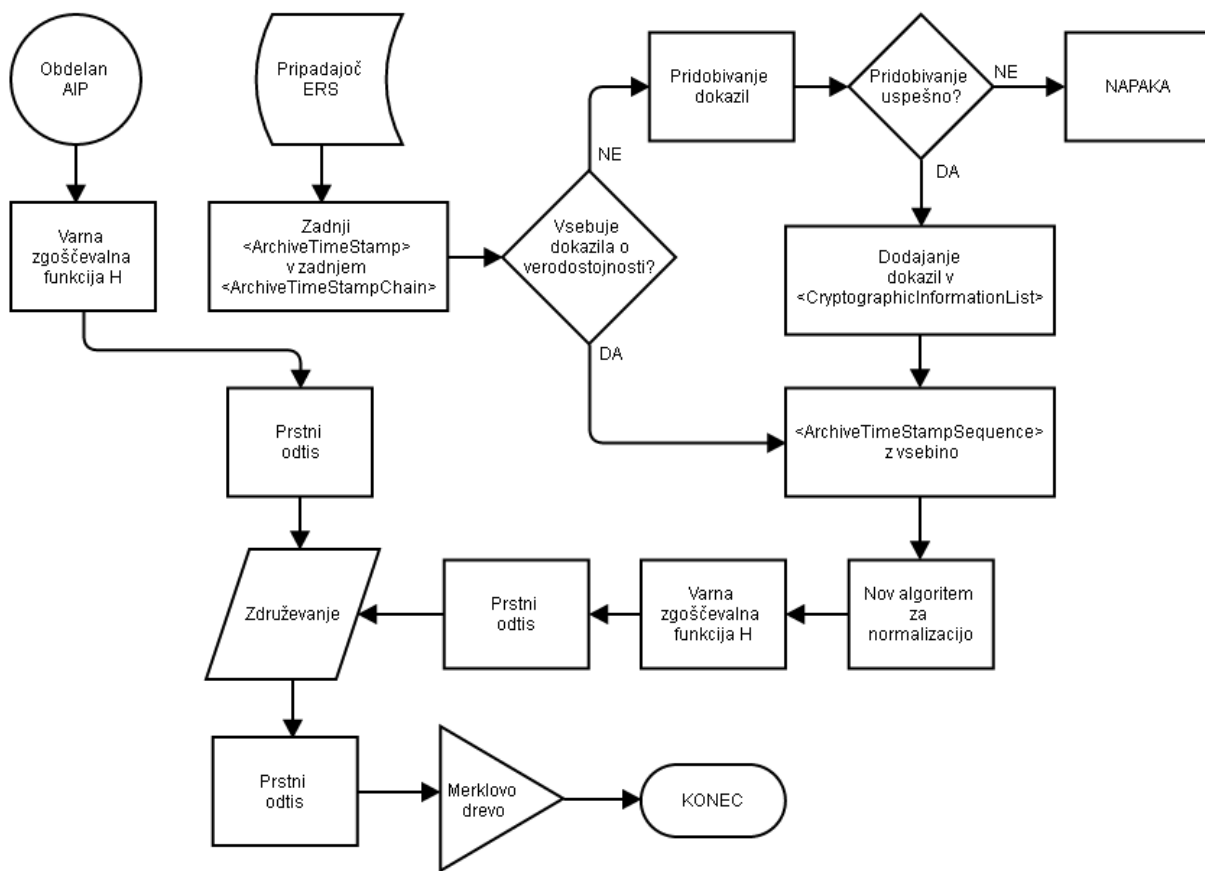
Kompleksno podaljševanje se zgodi ob menjavi šibkega zgoščevalnega algoritma za računanje prstnih odtisov z močnejšim. V tem primeru je potrebno kompleksno podaljšati vsa dokazila, ki so bila narejena z šibkejšim algoritmom za vse AIP, ustvarjene v procesu dolgoročne hrambe.

Postopek podaljševanja je naslednji:

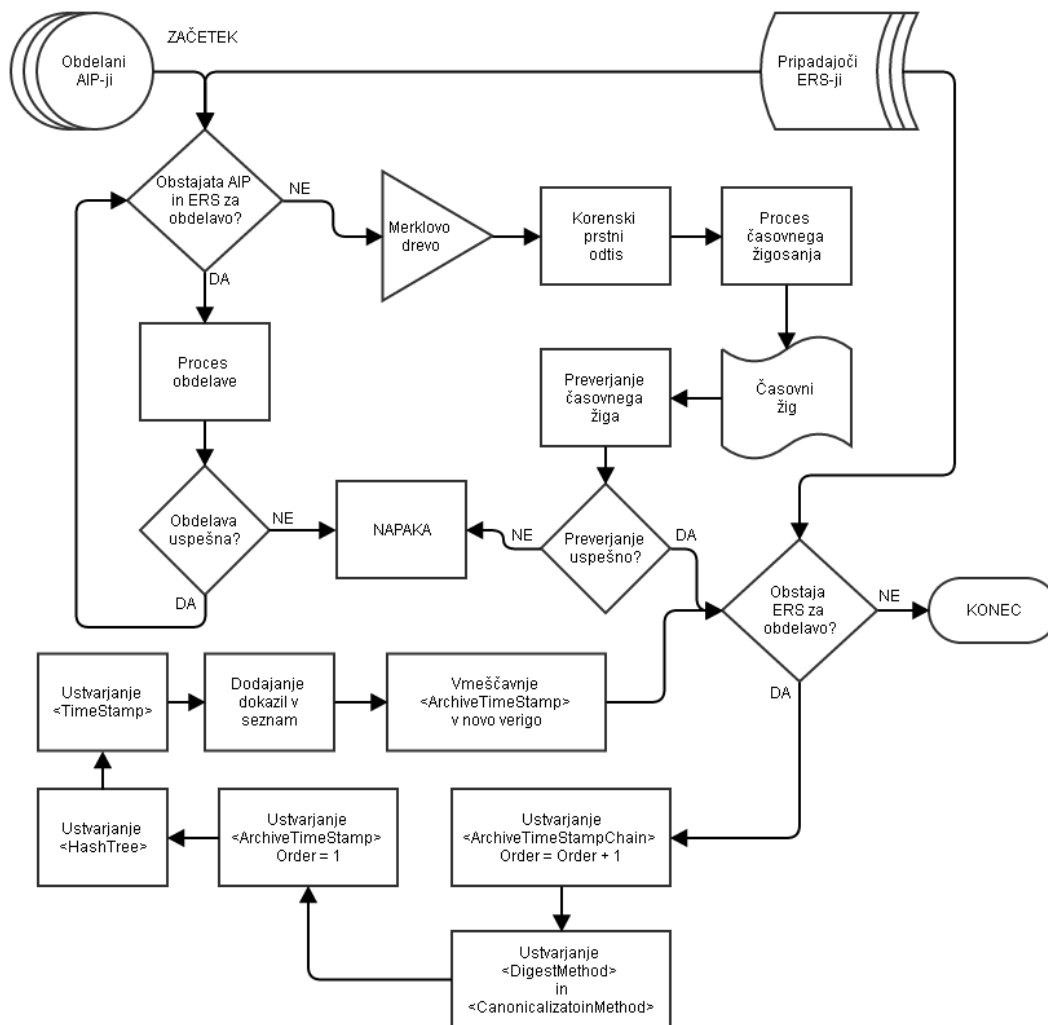
1. Izberemo nov močan zgoščevalni algoritem.
2. Izberemo nov algoritem za pretvorbo v normalizirano obliko.
3. Z novim zgoščevalnim algoritmom izračunamo prstni odtis obdelanega AIP.
4. Za pripadajoči ERS preverimo, ali zadnji arhivski časovni žig v trenutni verigi časovnih žigov vsebuje vse potrebne 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 procesa 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 procesa 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 procesu preverjanja.
14. Postopek iz točk 10. – 12. ponavljamo za vse ERS, ki so bili vključeni v proces kompleksnega podaljševanja.

Kompleksno podaljševanje z Merklovim drevesom prikazujeta spodnji sliki.



Slika 22: Postopek obdelave AIP in pripadajočega ERS



Slika 23: Postopek kompleksnega podaljševanja z Merkvim drevesom

3.6.4.7 Verifikacija

Verifikacija ERS je proces, v katerem preverimo avtentičnost vseh verig arhivskih časovnih žigov. S tem dokažemo avtentičnost AIP, ki ga ščiti ERS.

V osnovi lahko verifikacijo razdelimo 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 so 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 Predpogoji](#)).

AIP se generira v naslednjih primerih:

- Ob zaprtju entitete (če je entiteta tipa dokument ali zadeva);
- Ko ima zaprta entiteta vsaj en metapodatek ali vsebino, označeno za vključitev v arhivski informacijski paket. Spreminjanje entitete po tem, ko je bila enkrat zaprta, ni več možno.

Z več informacij [glej poglavje Predpogoji](#) in [poglavje 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
- vsebina
- 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 13: 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 14: 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 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 15: XML elementi etikete »Attribute«

Atribut »Id«

Vrednost atributa »Id« je naziv metapodatka.

Atribut »Type«

Vrednost atributa »Type« predstavlja tip metapodatka ([glej poglavje Vrste atributov](#)).

Etiketa »Value«

Vrednost etikete »Value« predstavlja vrednost atributa oziroma vrednosti, v primeru, da gre za atribut z več vrednostmi.

Primer: predstavlja metapodatke 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="ClientAddress1" Type="23">
  <aip:Value>Brnciceva 41g</aip:Value>
  <aip:Value>1231 Ljubljana</aip:Value>
</aip:Attribute>
<aip:Attribute Id="ClientAddress2" Type="23"/>

```

3.6.5.3 Vsebina

V sekcijo »Vsebina« so uvrščeni metapodatki tipa »File«, ki predstavljajo vsebovalnike (angl: Container) vsebin dokumenta ([glej poglavje Vrste atributov](#)).

Sekcija je sestavljena iz etiket »Content«, ki vsebujejo elemente navedene v spodnji tabeli.

Ime XML elementa	Tip XML elementa	Obveznost
Id	Atribut	DA
ContentValue	Etiketa	DA

Tabela 16: XML elementi etikete »Content«

Atribut »Id«

Vrednost atributa »Id« je ime metapodatka.

Etiketa »ContentValue«

Etiketa »ContentValue« vsebuje podatke o:

- tipu prstnega odtisa vsebine
- vrednosti prstnega odtisa
- morebitne elektronske podpise vsebine v XMLDSIG obliki.

Predpisuje jo W3C konzorcij (<http://www.w3.org/TR/xmlsig-core>).

XML elementi vsebovani v etiketi »ContentValue« so opisani v spodnji tabeli.

Ime XML elementa	Tip XML elementa	Obveznost
DigestMethod	etiketa	DA
DigestValue	etiketa	DA
Signature	etiketa	NE

Tabela 17: XML elementi etikete »ContentValue«

Etiketa »DigestMethod«

Etiketa »DigestMethod« vsebuje atribut »Algorithm«, ki je obvezen in katerega vrednost je URI. Ta predpisuje algoritem za izračun prstnega odtisa vsebine.

Spodnja tabla predstavlja podprte algoritme za izračun prstnih odtisov in pripadajoče URI, definirane 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 18: Algoritmi in pripadajoči URI-ji

Etiketa »DigestValue«

Vrednost etikete »DigestValue« je vrednost prstnega odtisa vsebine.

Etiketa »Signature«

»Signature« etiketa vsebuje:

- elektronski podpis v XMLDSIG obliki
- povezave do pripadajočih digitalnih potrdil
- povezave do pripadajočih informacij o preklicanih digitalnih potrdilih.

Primer: XML zapis prikazuje prstni odtis vsebine, narejen z algoritmom SHA256.

Vsebina ni podpisana, saj ne vsebuje etikete »Signature«.

```
<aip:Content Id="Content">
  <aip:ContentValue>
    <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
    <ds:DigestValue>Nrv1DtLoQdEEQ7y2...2G+Z+ycSorSwMRMpCw=</ds:DigestValue>
  </aip:ContentValue>
</aip:Content>
```

Ime XML elementa	Tip XML elementa	Obveznost
CertificateId	atribut	DA
RevocationDataId	etiketa	NE
Signature	etiketa	DA

Tabela 19: XML elementi etikete »Signature«

Atribut »CertificateId«

Vrednost atributa »CertificateId« je identifikator digitalnega potrdila, ki je generalal elektronski podpis.

Etiketa »RevocationDataId«

Vrednost etikete »RevocationDataId« je identifikator CRL ali OCSP podatka, ki dokazuje, da digitalno potrdilo, ki je generiralo podpis (ali katerokoli digitalno potrdilo v njegovi veji) ni preklicano.

Etiketa »Signature«

Vrednost etikete »Signature« je vrednost elektronskega podpisa v XML obliki (XMLDSIG).

Primer: XML zapis prikazuje elektronsko podpisane vsebine s pripadajočimi povezavami do digitalnega potrdila, ki je generiralo elektronski podpis in pripadajoči CRL.

```
<aip:Content Id="Content">
  <aip:ContentValue>
    <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmenc#sha256"/>
    <ds:DigestValue>Nrv1DtloQdEEQ7y...wo0t2G+Z+ycSorSwMRMpCw=</ds:DigestValue>
    <aip:Signature CertificateId="1">
      <dsig:Signature Id="TimeStampToken" xmlns:dsig="http://www.w3.org/2000/09/xmlsig#">
        ... digitalni podpis v XMLDSIG obliki ...
      </dsig:Signature>
    <aip:RevocationDataId>1</aip:RevocationDataId>
    </aip:Signature>
  </aip:ContentValue>
</aip:Content>
<aip:Certificate Id="1" ParentId="2" Type="X509DER">MIIFYDCCBEiBAg...</aip:Certificate>
<aip:Certificate Id="2" Type="X509DER">MIIEHDCCAwSgAwI...</aip:Certificate>
<aip:RevocationData Id="1" Type="CRLDER">MIISKTCERECaqEwDQ...</aip:RevocationData>
```

3.6.5.4 Digitalna potrdila

V to sekcijo so uvrščene celotne veje digitalnih potrdil, ki pripadajo elektronskim podpisom in so Base64 kodirana.

Sekcija je sestavljena iz etikete »Certificate«, ki vsebujejo elemente navedene v spodnji tabeli.

Ime XML elementa	Tip XML elementa	Obveznost
Id	Atribut	DA
ParentId	Atribut	NE
Type	Atribut	DA

Tabela 20: XML elementi etikete »Certificate«

Atribut »Id«

Vrednost atributa »Id« je unikatni identifikator.

Atribut »ParentId«

Vrednost atributa »ParentId« je identifikator digitalnega potrdila, ki je podpisnik trenutnega digitalnega potrdila. Če je digitalno potrdilo korensko (angl. Root certificate), potem atribut ni prisoten.

Atribut »Type«

Vrednost atributa »Type« predstavlja tip digitalnega potrdila. Trenutno podprta vrednost je »X509DER«, ki predstavlja digitalno potrdilo tipa X509 v binarni obliki.

Primer: XML zapis prikazuje dve digitalni potrdili tipa X509, ki tvorita vejo. Digitalno potrdilo z »Id = 2« je podpisnik digitalnega potrdila z »Id = 1«.

```
<aip:Certificate Id="1" ParentId="2" Type="X509DER">
MIIFYDCCBEigAwIBAg...
</aip:Certificate>
<aip:Certificate Id="2" Type="X509DER">
MIIHDCCAwSgAwI...
</aip:Certificate>
```

3.6.5.5 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 21: 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 22: Podprti tipi informacij o preklicu digitalnih potrdil

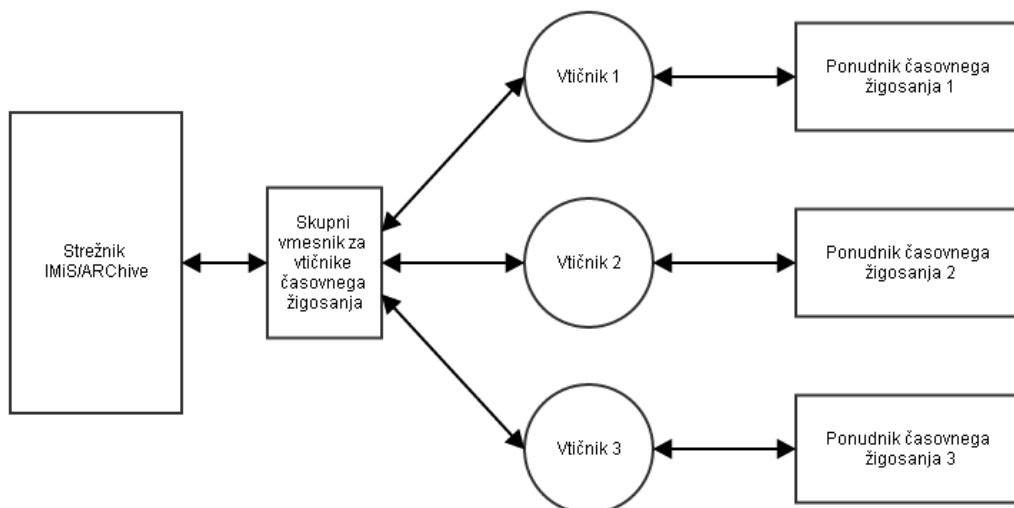
Primer: XML zapis za oba podprta tipa informacij.

```
<aip:RevocationData Id="1" Type="CRLDER">
MIISKTCCERCAQEwDQ...
</aip:RevocationData>
<aip:RevocationData Id="2" Type="OCSP">
MIIDBjCB7wI...
</aip:RevocationData>
```

3.6.6 Časovno žigovanje

Časovni žigi so dokazila, ki dokazujejo obstoj vsebine pred časom navedenim v časovnem žigu. Storitve časovnega žigovanja 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 vtičnik, ki zna komunicirati s ponudnikom časovnih žigov in jih na strežniku znan način vključevati v procese zagotavljanja avtentičnosti hranjenega gradiva.



Slika 24: Koncept vtičnikov

3.6.6.1 Pridobivanje časovnega žiga

Proces 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 z 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 Razširjena uporaba ključa](#)).
- Preveri se veljavnost elektronskega podpisa časovnega žiga.

Če je katerikoli 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-13ED106F54C2C33ED42000000000007B81">
      <dsig:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
      <dsig:DigestValue>LaeChaxwIaM8e9WZ1RD0QjxzFrw=</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
qDC01zRogaPojVC06Ee545/VrdP1JmnzcwFXAW3UU+q6VrDDHLLyD
z4uW9hXxEy31YNkQmJ7BOicHNY9m2T0lk8tjCSec6s5UJxhJP49tY
u8wE7gMSgWpLlnMeAZCE/DVOP1qesVTUYzaLSbEqELpL5qFkvCmNC
TBjnaNuyKe/YhQWbvZ0cIvHePqyADNwX+IPOsA0S8NezpZHYriBO8
B+cAwgIep/gZb1h8zDIqeJHS8ibnFmVblk3Z01bG/Y1SK36yk+Fu5
ya12KH1FOACzx/im3GE8v1WQ==
  </dsig:SignatureValue>
  <dsig:KeyInfo Id="TimeStampAuthority">
    <dsig:X509Data>
      <dsig:X509Certificate>
MIIFYDCCBEigAwIBAgIEQlMGwDANBgkqhkiG9w0BAQUFADA+MQswC
QYDVQQGEwJzaTEbMBkGA1UEChMSc3RhdGUtaW5zdG10dXRpb25zMR
IwEAYDVQQLEw1zaXRlc3QtY2EwHhcNMTIwNTIwMDc0ODM4WbcNMTC
wNTIwMjExMDU2WjBvMQswCQYDVQQGEwJzaTEbMBkGA1UEChMSc3R
hdGUtaW5zdG10dXRpb25zMRIwEAYDVQQLEw1TSVRFU1QtQ0ExLzAUB
gNVBAMTDXRzYS10ZXN0LTlWMTIwFwYDQ0FEwAxMTEwMTEwMTEwMT
...
MzFOVSwteivgwgVmtUZjSbeClpELUblGWzwyxW15SHufOdJdtgcnn
r2hCLtkYJ9ky4T0m5gvvy0xmQi+o3rLvPa5yLeHYa//KzPo+H8CPf
UGqRFTPVSYkalm6evTg6CDQvG9jfiilPbaeQRm0xLQPpVNYFA66e
2DwVv+9UdKzcoBfujvZRtEuPlnphHBlzxEkFU3
      </dsig:X509Certificate>
    </dsig:X509Data>
  </dsig:KeyInfo>
  <dsig:Object Id="TimeStampInfo-13ED106F54C2C33ED42000000000007B81">
    <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>
```

```
108487637460984003369566416624147310345089
</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-13ED106F54C2C33ED42000000000007B81": 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 procesu ustvarjanja zahtevka za časovno žigosanje.

3.6.7 Pravila

Pravila določajo pogoje, kdaj naj se izvajata procesa ustvarjanja in podaljševanja dokazil in katere vsebine so predmet teh procesov.

Strežnik IMiS®/ARChive Server uporablja pravilo z naslednjimi lastnostmi:

- Ustvarjanje in podaljševanje dokazil poteka paketno.
Uporablja Merklovo drevo razen v primeru, če se v paketu nahaja samo en AIP ali arhivski časovni žig.
- Pri paketnem podaljševanju ne ločimo med ustvarjanjem in podaljševanjem dokazil. V isto Merklovo drevo dodamo prstne odtise AIP, za katere je potrebno ustvariti začetno dokazilo ter prstne odtise arhivskih časovnih žigov, ki jih je potrebno podaljšati.
- Arhivski časovni žigi postanejo predmet podaljšanja, če se veljavnost digitalnega potrdila, ki je ustvarilo časovni žig izteče v 180 dneh od trenutka preverjanja.

3.7 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 procese in operacije:

- preverjanje istovetnosti (angl. Authentication)
- pridobivanje informacij o registrirani entiteti v strežniškem imeniku preko atributov
- upravljanje z seznamom dostopnih pravic.

Preverjanje istovetnosti

Preverjanje istovetnosti je proces, 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 laično povedano 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 izdelava 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 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 podpoglavje Efektivne pravice v ACL](#)).

3.7.1 Tipi

Strežniški imenik podpira dva tipa entitet:

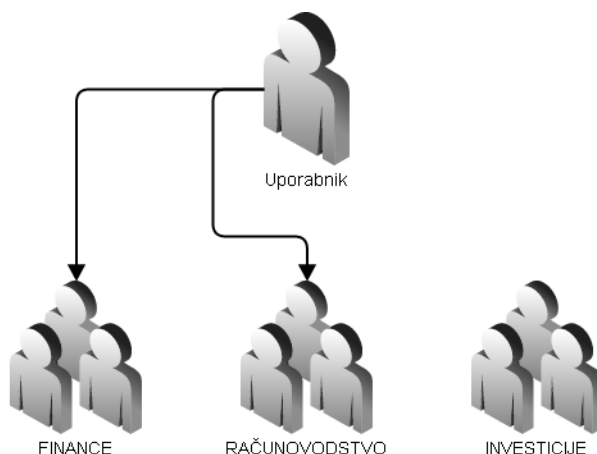
- uporabnike
- uporabniške skupine.

3.7.1.1 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.

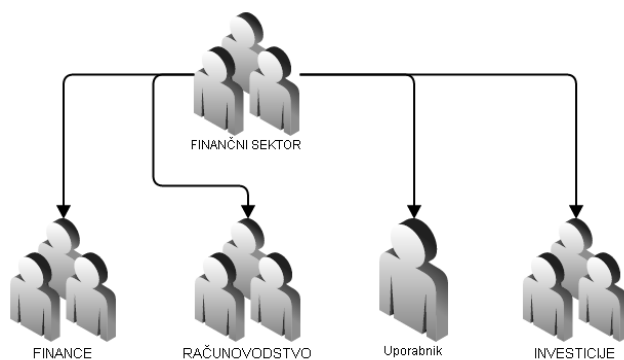


Slika 25: Pripadnost uporabnika v uporabniških skupinah

3.7.1.2 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 26: Primer gnezdenja uporabniških skupin

3.7.2 Preddefinirane (sistemske) določene entitete

Strežnik IMiS®/ARChive Server ima v svojem imeniku vnaprej določene naslednje sistemske ali navadne entitete:

- sys:Administrator
- sys:Administrators
- sys:CurrentUser
- 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 procesu ustvarjanja nove entitete (razreda, dokumenta ali zadeve) zamenja z identifikatorjem uporabnika, ki ustvarja entiteto.

S tem uporabnik prevzame dovoljenja in prepovedi, ki postanejo eksplicitne za uporabnika na tej entiteti.

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 procesu 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.

3.7.3 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 procesu 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 zlogi 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.
- AdvancedAuthenticationEnabled: Določa, ali uporabniški račun omogoča avtentikacijo preko napredne avtentikacijske metode.
- PreSharedKeyAuthenticationEnabled: Določa, ali uporabniški račun omogoča avtentikacijo preko napredne avtentikacijske metode s kriptiranjem preko deljenega ključa.
- SRP6aAuthenticationEnabled: Določa, ali uporabniški račun omogoča avtentikacijo preko SRP-6a avtentikacijske metode.

»AuthenticationType«

Številčna vrednost predstavlja tip avtentikacije, ki jo uporabnik lahko izbere za avtentikacijo na strežniku. Trenutno podprta vrednost je TYPE_SRP6A, ki predstavlja avtentikacijo na strežniku po SRP protokolu (angl. Secure Remote Password protocol).

»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 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 Dostopi](#)).

3.7.4 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".

3.7.5 Avtentikacija

Avtentikacija uporabnika je del procesa, ki se izvede pri vzpostavljanju uporabniške seje z arhivskim strežnikom.

V tem procesu se preveri, ali se poverilnice, ki jih je poslal uporabnik ujemajo s poverilnicami uporabnika, shranjenimi v podatkovni bazi strežnika. Če se ujemajo, je uporabnik dokazal verodostojnost in lahko opravlja operacije na strežniku v skladu z njegovimi pravicami. Če se poverilnice ne ujemajo, strežnik zavrne uporabniško sejo, saj uporabnik ni dokazal svoje verodostojnosti.

Za proces 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 resistance).
- Odpornost na aktiven napad s slovarjem (angl. On-line dictionary attack resistance).
- 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), potem 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 ugame natanko eno geslo). Take vrste napadov se da preprosto zaznati na strežniku, ter jih preprečiti (recimo z zavrnitvijo vzpostavljanja seje z določene IP številke po določenem številu neuspešnih 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

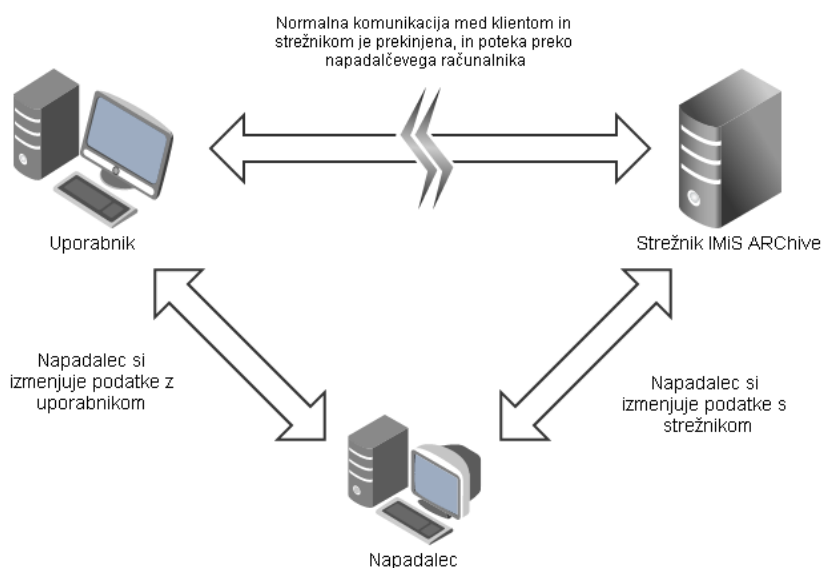
Če napadalcu uspe napad in vzpostavi sejo s strežnikom v imenu nekoga drugega (angl. Impersonation attack), se predvideva, da napadalec pozna celoten postopek vzpostavljanja seje (vključno z vsemi vmesnimi koraki, ki jih izvajata strežnik in odjemalec in niso javno znani, oziroma se ne pošiljajo po protokolu).

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>.

3.7.5.1 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 27: 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 zaupanju vrednih tretjih oseb (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 geslo, ne da bi strežniku poslal kakršenkoli dokaz (prstni odtis gesla, ...) o vrednosti le-tega.

SRP (verzija 6) se uporablja:

- pri zagotavljanju varnosti transportnega sloja (angl. Transport layer security – SRP verzijo opisuje povezava <http://en.wikipedia.org/wiki/TLS-SRP>)
- v EAP protokolu (angl. Extensible authentication protocol – RFC 3748)
- v SAML protokolu.

SRP je poleg tega še del standardov IETF (RFC 2944, RFC 2945, RFC 5054), IEEE (P1363.2) in ISO (IEC 11770-4).

3.7.5.1.1 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 - http://en.wikipedia.org/wiki/Salt_%28cryptology%29), ki se uporablja pri ustvarjanju prstnega odtisa gesla.
- 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; ju 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 proces 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 »I« 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:

$$A \neq 0 \pmod{N} - \text{preverjanje na strežniku}$$

$$B \neq 0 \pmod{N} - \text{preverjanje na odjemalcu}$$

Če katerikoli izmed parametrov ne ustreza kriterijem, potem se proces avtentikacije prekine.

Če parametra ustrezata kriterijem, se proces 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 proces avtentikacije prekine.

V nadaljevanju avtentikacije odjemalec in strežnik izračunata vsak svoj dokaz o poznavanju gesla in varen sejni ključ »K«.

Računanje varnega sejnega ključa na odjemalcu

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:

$$M_1 = H(H(N) \text{ xor } H(g), H(I), s, A, B, K)$$

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:

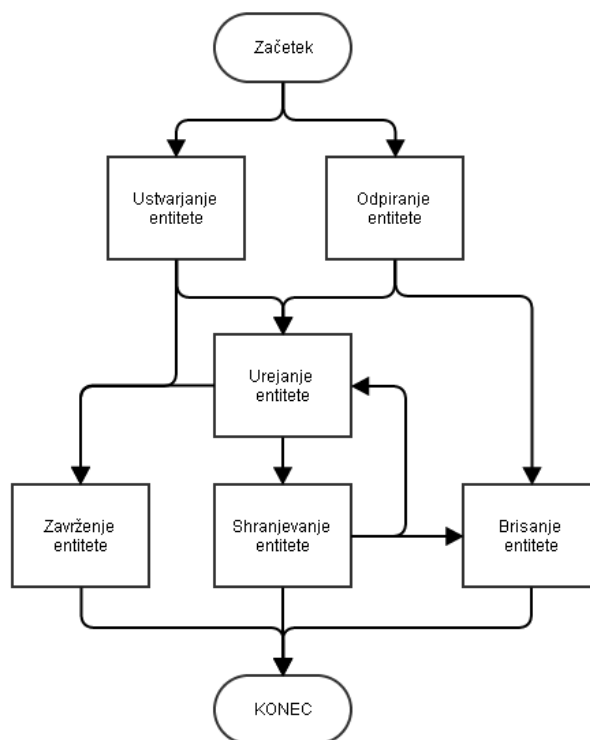
$$M_2 = H(A, M_1, K)$$

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.

3.7.6 Življenjski cikel entitet imenika

Življenjski cikel entitete v imeniških storitvah lahko razdelimo na šest vrst procesov:

- odpiranje entitete
- ustvarjanje entitete
- urejanje entitete
- shranjevanje entitete
- zavrženje entitete (angl. Discard)
- brisanje entitete.



Slika 28: Povezava med procesi v življenjskem ciklu entitete v imeniku

3.7.6.1 Odpiranje entitete

Odpiranje entitete je postopek pridobivanja primerka (angl. Instance) entitete iz imenika. Strežnik IMiS®/ARChive Server v procesu uvodne nastavitve (angl. Initialization) naloži vse entitete, ki se v imeniku nahajajo. Istočasno pri odpiranju entitete preveri, če je entiteta predhodno naložena. V primeru da je naložena, vrne njen primerek, v nasprotnem primeru pa javi napako.

3.7.6.2 Ustvarjanje entitete

Proces ustvarjanja entitete je prvi v življenjskem ciklu entitete. Na začetku se določi tip entitete in uporabniški račun ali ime uporabniške skupine, če gre za skupino. V procesu urejanja lahko nastavimo še dodatne komponente glede na to, za kateri tip entitete gre. V primeru, da se novo ustvarjena entiteta zavrže preden je le-ta shranjena, se vsebina imenika ne spremeni.

3.7.6.3 Urejanje entitete

Urejanje entitete je mogoče na novo ustvarjenih entitetah ter entitetah, ki so že v imeniku. Pri urejanju entitet se lahko nastavljajo in spreminjajo komponente, ki so opisane v [poglavju Komponente entitete](#).

3.7.6.4 Shranjevanje entitete

Shranjevanje entitete je proces, kjer se vse vrednosti, ki smo jih ustvarili ali spremenili v procesu urejanja, shranijo v imenik.

3.7.6.5 Zavrženje entitete

Do zavrženja entitete 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 pa pride v imenik le v primeru shranjevanja.

3.7.6.6 Brisanje entitete

Brisanje entitete poteka tako, da se entiteto v imeniku označi kot izbrisano, fizično pa se imenik ne izbriše. Tak uporabnik ali uporabniška skupina tako 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 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
ACLENTY	Vnosi za listo dostopnih pravic.
ACLENTYVALIDITY	Časovne omejitve dostopnih pravic.
ATTRIBUTE	Atributi, ki si 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

Tabela	Opis
	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, dolge 10 bytov.
ATTRIBUTEVALUESTRING20	Tabela vsebuje nize UTF-8 znakov, dolge 20 bytov.
ATTRIBUTEVALUESTRING30	Tabela vsebuje nize UTF-8 znakov, dolge 30 bytov.
ATTRIBUTEVALUESTRING40	Tabela vsebuje nize UTF-8 znakov, dolge 40 bytov.
ATTRIBUTEVALUESTRING50	Tabela vsebuje nize UTF-8 znakov, dolge 50 bytov.
ATTRIBUTEVALUESTRING100	Tabela vsebuje nize UTF-8 znakov, dolge 100 bytov.
ATTRIBUTEVALUESTRING200	Tabela vsebuje nize UTF-8 znakov, dolge 200 bytov.
ATTRIBUTEVALUESTRING20IDX	Tabela vsebuje indeksirane nize UTF-8 znakov, dolge 20 bytov.
ATTRIBUTEVALUESTRING30IDX	Tabela vsebuje indeksirane nize UTF-8 znakov, dolge 30 bytov.
ATTRIBUTEVALUESTRING40IDX	Tabela vsebuje indeksirane nize UTF-8 znakov, dolge 40 bytov.
ATTRIBUTEVALUESTRING50IDX	Tabela vsebuje indeksirane nize UTF-8 znakov, dolge 50 bytov.
ATTRIBUTEVALUESTRING100IDX	Tabela vsebuje indeksirane nize UTF-8 znakov, dolge 100 bytov.
ATTRIBUTEVALUESTRING200IDX	Tabela vsebuje indeksirane nize UTF-8 znakov, dolge 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.

Tabela	Opis
ATTRIBUTEVALUEUINT128IDX	Tabela vsebuje indeksirana ne-predznačena 128 bitna cela števila.
COMPRESSION	Tabela vsebuje informacije o vtičnikih za kompresiranje podatkov.
CONTENT_TYPE	Tabela vsebuje informacije o podprtih MIME tipih na strežniku IMiS®/ARChive Server.
DIGITALCERTIFICATE	Tabela vsebuje digitalna potrdila, ki se nahajajo v strežniški shrambi digitalnih potrdil.
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.
COMPRESSIONLIB	Tabela vsebuje informacije o knjižnicah za kompresiranje podatkov.
PROFILE	Tabela vsebuje podatke o HSM profilih.
PROPERTY	Generična tabela nastavitvev.
STORAGE_DRIVER	Tabela vsebuje informacije o gonilniku za HSM volumne.
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 predlogami.
ARCHIVALINFORMATIONPACKAGE	Tabela vsebuje arhivske informacijske pakete.
ARCHIVALINFORMATIONPACKAGEQUEUE	Tabela predstavlja čakalno vrsto za AIP.
CREATEAIPQUEUE	Tabela predstavlja čakalno vrsto za

Tabela	Opis
	ustvarjanje AIP.
AIPJOBTIMESTAMP	Tabela vsebuje podatke o procesu časovnega žigosanja.
REVOCATIONDATA	Tabela vsebuje informacije o preklicih digitalnih potrdil.
FULLTEXTINDEXINGQUEUERECORD	Tabela vsebuje informacije za indeksiranje vsebin.
TIMESTAMP	Tabela vsebuje časovne žige.
TIMESTAMPRULE	Tabela vsebuje pravila za časovno žigosanje.
VOLUME	Tabela vsebuje podatke o HSM volumnih.
LOOKUPTABLE	Tabela vsebuje statuse entitet.
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.
TIMESTAMPARCHIVALINFORMATIONPACKAGE	Tabela vsebuje arhivske informacijske pakete za časovne žige.
CERTIFICATEBODY	Tabela vsebuje binarne ali Base64 kodirane podatke, ki predstavljajo digitalno potrdilo.

Tabela 23: 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, ...).

3.8.1 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.

3.8.1.1 Nastavitve varnostnega kopiranja

Nastavitve varnostnega kopiranja so odvisne od tega, katero orodje za varnostno kopiranje uporabimo. V večini primerov se opis nastavitve 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.

3.8.2 Obnovitev podatkov

Obnovitev arhiviranih dokumentov na strežniku IMiS®/ARChive Server je odvisna od stanja strežnika in željenega 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 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 željenega 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.

3.8.2.1 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 Konfiguriranje](#)).

3.8.3 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.

3.8.4 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 pravih 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 procesu uvoza podatkov zgoraj opisano napako najprej izbrišemo (uporabimo ukaz `init` v `dbtool-u`), ter ponovimo postopek uvoza podatkov za to tabelo (to ne velja za uvoz revizijske sledi, saj se le-te ne da izbrisati).
- pri uvozu podatkov z orodjem `dbtool` uporabimo možnost `-o`, ki morebitne obstoječe podatke v bazi prepíše 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"
```

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 24: Povprečne velikosti skeniranega dokumenta pri uporabi različnih metod 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 hrbtnico 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 zahtevke 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 namestitvev, nadgradnjo in administriranje strežnika. Za svoje delovanje ne potrebuje privilegiranih 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 kakršenkoli 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 GSM 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).
- Katerikoli 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 priklon 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
libz2.so.1
libz.so.1
rpmllib

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 Strojna oprema](#) in [poglavje 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 pripravljene in dostopne 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 ~]# sudo rpm -ivh imisarc.9.1.1406-600.0001.e14.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:

<code>/iarc/db</code>	<code>in</code>	
<code>/iarc/db/iarc</code>		imenika, ki vsebujeta datoteke podatkovne baze strežnika;
<code>/iarc/fti</code>		imenik, ki vsebuje podatkovno bazo sistema za iskanje po polnem besedilu dokumentov;
<code>/iarc/vol</code>		imenik, ki vsebuje volumne, kamor strežnik IMiS®/ARChive Server shranjuje datoteke dokumentov;
<code>/iarc/wcache</code>		imenik, ki ga strežnik uporablja za predpomnenje datotek dokumentov pri nastajanju ali urejanju;
<code>/iarc/rcache</code>		imenik za predpomnenje datotek dokumentov, za katere uporabniki podajo zahtevo za pregledovanje;
<code>/iarc/work</code>		imenik, v katerem strežnik shranjuje datoteke začasne narave pri posameznih procesih;
<code>/opt/IS/imisarc</code>		vsebuje izvajalne datoteke in knjižnice, ki jih strežnik potrebuje za svoje delovanje;
<code>/etc/iarc.conf</code>		konfiguracijska datoteka strežnika;
<code>/etc/init.d/iarc</code>	<code>d</code>	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
```

ali nastavimo pravice kar 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 systemske administracije strežnika, kamor se strežnik IMiS®/ARChive Server namešča
- osebni preferenci glavnega systemskega 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 iarc on`
- Za SLES in OpenSuSE distribucije z ukazom: `chkconfig iarc 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 iarc -list`
(izpiše na katerih nivojih se servis samodejno zažene).
- Za SLES in OpenSuSE distribucije z ukazom: `chkconfig iarc -list` ali `yast`.

Pomembno je, da se servis zažene na nivoju 3 in 5 kar potrjuje naslednji izpis:

```
[user1@iarc ~]# sudo chkconfig iarc -list
iarc          0:off  1:off  2:on   3:on   4:on   5:on   6:off
```

5.3 Preizkušanje namestitve in nastavitvev

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 iarc start
Starting IMiS/ARChive HSM Storage Server: [ OK ]
```

Korak 2

Z ukazom `ps tree -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 ps tree -G
init └─ ...
...
  └─ iarc └─ iarc ── 7*[{iarc}]
            └─ 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, definiranih v `iarc.conf` pričakujeta zahteve.

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 (opsijsko)
- 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.e14.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 succesfull! 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/vol100" 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/vol100 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/vol100) 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/vol100 -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 izvajajo 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 overridden 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, ...).

8.1 Postopek zagona in zaustavitve

Za zagon in zaustavitev strežnika IMiS®/ARChive Server uporabimo zagonsko skripto.

Skripta `iarc.d` 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 iarc.d <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 *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 nastavitve 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 nastavitve ustvarjajo po potrebi z naslovno shemo `/var/log/iarc/iarc.XX.log` (XX = sekvenca arhivske datoteke, večja številka predstavlja dogodka, 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. "nultege" 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 po navadi 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(quiet)] [-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

```

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 lokaciji – /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 ACLENTY (3 records) to acleentry.txt ... OK.
Exporting ACLENTYVALIDITY (0 records) to acleentryvalidity.txt ... OK.
Exporting ATTRIBUTE (52 records) to attribute.txt ... OK.
Exporting ATTRIBUTEGROUP (0 records) to attributegroup.txt ... OK.
Exporting ATTRIBUTEVALUE (0 records) to attributevalue.txt ... OK.
Exporting ATTRIBUTEVALUEBINARY (0 records) to attributevaluebinary.txt ...
OK.
Exporting ATTRIBUTEVALUEDATETIME (0 records) to attributevaluedt.txt ...
OK.
Exporting ATTRIBUTEVALUEDOUBLE (0 records) to attributevaluedouble.txt ...
OK.
Exporting ATTRIBUTEVALUEFILE (0 records) to attributevaluefile.txt ... OK.
Exporting ATTRIBUTEVALUEINT32 (0 records) to attributevalueint32.txt ... OK.
Exporting ATTRIBUTEVALUEINT64 (0 records) to attributevalueint64.txt ... OK.
Exporting ATTRIBUTEVALUEINT128 (0 records) to attributevalueint128.txt ...
OK.
Exporting ATTRIBUTEVALUEINT128IDX (0 records) to attributevalueint128idx.txt
... OK.

```

Exporting ATTRIBUTEVALUESTRING (0 records) to attributevaluemaxstring.txt ... OK.
Exporting ATTRIBUTEVALUESTRING10 (0 records) to attvalstr10.txt ... OK.
Exporting ATTRIBUTEVALUESTRING20 (0 records) to attvalstr20.txt ... OK.
Exporting ATTRIBUTEVALUESTRING30 (0 records) to attvalstr30.txt ... OK.
Exporting ATTRIBUTEVALUESTRING40 (0 records) to attvalstr40.txt ... OK.
Exporting ATTRIBUTEVALUESTRING50 (0 records) to attvalstr50.txt ... OK.
Exporting ATTRIBUTEVALUESTRING100 (0 records) to attvalstr100.txt ... OK.
Exporting ATTRIBUTEVALUESTRING200 (0 records) to attvalstr200.txt ... OK.
Exporting ATTRIBUTEVALUESTRING20IDX (0 records) to attvalstr20idx.txt ... OK.
Exporting ATTRIBUTEVALUESTRING30IDX (0 records) to attvalstr30idx.txt ... OK.
Exporting ATTRIBUTEVALUESTRING40IDX (0 records) to attvalstr40idx.txt ... OK.
Exporting ATTRIBUTEVALUESTRING50IDX (0 records) to attvalstr50idx.txt ... OK.
Exporting ATTRIBUTEVALUESTRING100IDX (0 records) to attvalstr100idx.txt ... OK.
Exporting ATTRIBUTEVALUESTRING200IDX (0 records) to attvalstr200idx.txt ... OK.
Exporting ATTRIBUTEVALUEUINT32 (0 records) to attvaluint32.txt ... OK.
Exporting ATTRIBUTEVALUEUINT64 (0 records) to attvaluint64.txt ... OK.
Exporting ATTRIBUTEVALUEUINT128 (0 records) to attvaluint128.txt ... OK.
Exporting ATTRIBUTEVALUEUINT128IDX (0 records) to attvaluint128idx.txt ... OK.
Exporting audit log data (11 records) to auditlog.bin ... OK.
Exporting COMPRESSION (1 records) to compression.txt ... OK.
Exporting CONTENT_TYPE (1334 records) to objtype.txt ... OK.
Exporting DIGITALCERTIFICATE (2 records) to digitalcert.txt ... OK.
Exporting COUNTER (0 records) to counter.txt ... OK.
Exporting DIGITALSIGNATURE (0 records) to digitalsig.txt ... OK.
Exporting DIRECTORYENTRY (10 records) to direntry.txt ... OK.
Exporting DIRECTORYENTRYALIAS (3 records) to direntryalias.txt ... OK.
Exporting DIRECTORYENTRYGROUP (6 records) to direntrygroup.txt ... OK.
Exporting ENTITY (26 records) to entity.txt ... OK.
Exporting ENTITYCLASS (6 records) to entityclass.txt ... OK.
Exporting ENTITYDOCUMENT (0 records) to entitydocument.txt ... OK.
Exporting ENTITYFOLDER (0 records) to entityfolder.txt ... OK.
Exporting COMPRESSIONLIB (1 records) to compresslib.txt ... OK.
Exporting PROFILE (2 records) to profile.txt ... OK.
Exporting PROPERTY (2 records) to property.txt ... OK.
Exporting STORAGE_DRIVER (1 records) to store.txt ... OK.
Exporting TEMPLATE (20 records) to template.txt ... OK.
Exporting TEMPLATEATTRIBUTE (74 records) to templateatt.txt ... OK.
Exporting TEMPLATEBIND (17 records) to templatebind.txt ... OK.
Exporting ARCHIVALINFORMATIONPACKAGE (0 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 (0 records) to revodata.txt ... OK.

```
Exporting FULLTEXTINDEXINGQUEUEURECORD (0 records) to ftiqueuerecord.txt ...
OK.
Exporting TIMESTAMP (0 records) to timestamp.txt ... OK.
Exporting TIMESTAMPRULE (0 records) to timestamprule.txt ... OK.
Exporting VOLUME (2 records) to volume.txt ... OK.
Exporting LOOKUPTABLE (3 records) to lookup.txt ... OK.
Exporting HASH128 (0 records) to hash128.txt ... OK.
Exporting HASH160 (0 records) to hash160.txt ... OK.
Exporting HASH224 (0 records) to hash224.txt ... OK.
Exporting HASH256 (0 records) to hash256.txt ... OK.
Exporting HASH384 (0 records) to hash384.txt ... OK.
Exporting HASH512 (0 records) to hash512.txt ... OK.
Exporting REVOCATIONDATATIMESTAMP (0 records) to revodatats.txt ... OK.
Exporting REVOCATIONDATADIGITALSIGNATURE (0 records) to revodatadsig.txt ...
OK.
Exporting TIMESTAMPARCHIVALINFORMATIONPACKAGE (0 records) to tsaip.txt ...
OK.
Exporting CERTIFICATEBODY (2 records) to certbody.txt ... OK.
Export command completed with no errors.
```

Opomba: Številke (xx records) pri vsaki vrstici se lahko spreminjajo, odvisno od števila zapisov.

Rezultat je 70 tekstovnih datotek, ki predstavljajo posamezne tabele v bazi, binarna datoteka, ki predstavlja revizijsko sled ter dodatni podatki (shranjeni v datotekah s končnicami `.blob` in `.txt`), ki vsebujejo del interne baze strežnika IMiS®/ARChive Server.

8.4 Administracija

Administracijska opravila lahko bistveno vplivajo na delovanje produkta. Pravilna namestitvev 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 načrt razvrščanja gradiva, predloge entitet, metapodatkovne sheme atributov, nastavitve vtičnikov za različne podporne storitve sistema, itd. V trenutni verziji produkta so ti posegi omogočeni s pomočjo konzolnih orodij in poznavanja podatkovnega modela sistema, prihodnje verzije pa bodo tovrstno funkcionalnost omogočile skozi administracijski vmesnik. 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 nastavitvev zaradi nepravilne prilagoditve nastavitvev in/ali baze.

8.4.1 Konfiguracijska datoteka iarc.conf

Nastavitveni parametri strežnika IMiS®/ARChive Server so nastavljeni 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 (opsijsko 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 zahteve. 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 na 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 zahteve odjemalcev. Privzeta vrednost je 16807.
Listen:	Označuje omrežni naslov na katerega strežnik veže TCP vrata na katerih pričakuje zahteve 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:efaf:1244:32] fd00:192:168:92:2340:efaf:1244:32 [fd00:192:168:92:2340:efaf: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 <code>Path</code> opisanega zgoraj.
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 <code>/var/run/iarc</code> .
CountryLanguage:	Označuje regionalne nastavitve v POSIX formatu: <code>xx_YY[.CHARSET[@variant]]</code> ; prvi dve črki <code>xx</code> predstavljata kodo jezika po standardu ISO-639, drugi dve črki predstavljata kodo države po standardu ISO-3166, <code>CHARSET</code> podatek (opcijski) določa kodno tabelo oziroma kodni razpored (seznam dostopen v imeniku <code>/usr/share/i18n/charsets</code>), podatek <code>variant</code> (opcijski) pa določa nacionalne posebnosti jezika. Seznam možnih nastavitvev je možno pridobiti preko izpisa ukaza <code>"locale -a"</code> . 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 <code>"</code> , kar učinkovito določa sistemsko nastavljenih pravila (glej sistemski ukaz <code>"locale"</code> 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 <code>/iarc/db</code> .
MaxRequestSize:	Določa največji možen enkratni zahtevek odjemalca. Privzeta vrednost je 3276800 byte-ov. Najmanjša možna vrednost je 32768 in največja 32768000.

Sekcija [Database]

`ProviderBootstrapFile`: Označuje relativno pot to 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 lahko bere in zapisuje datoteke.
ReadSize:	Označuje najmanjšo velikost medpomnilnika. Strežnik to omejitve prilagaja dinamično, glede na potrebe in 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 omejitve prilagaja dinamično, glede na potrebe in 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 lahko 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 noter – zadnji ven. Privzeta vrednost je 1, priporočena 9.
LogLevel:	Nivo dogodkov, ki jih strežnik zapisuje v dnevniško datoteko (glej poglavje Beleženje dogodkov delovanja). Najmanjša vrednost je 1, največja 7, priporočena 6.

Sekcija [AuditLog]

Enabled:	Z vrednostjo 1 dnevnik beleženja operacij nad objekti omogočimo, z vrednostjo 0 dnevnik beleženja operacij nad objekti onemogočimo. Strežnik IMiS®/ARChive Server dogodke beleži v šifrirani obliki v interni bazi podatkov.
----------	--

Events:	<p>Označuje nabor operacij, ki jih strežnik kot dogodke beleži v dnevnik.</p> <p>Veljaven nabor operacij:</p> <p><code>ent.create</code> (kreiranje entitete)</p> <p><code>ent.update</code> (shranjevanje entitete)</p> <p><code>ent.delete</code> (brisanje entitete)</p> <p><code>ent.move</code> (premikanje entitete v načrtu razvrščanja gradiva)</p> <p><code>ent.openro</code> (odpiranje entitete z namenom branja)</p> <p><code>ent.openrw</code> (odpiranje entitete z namenom branja in spreminjanja)</p> <p><code>ent.acl</code> (sprememba dostopnih pravic na entiteti z naštetimi spremembami)</p> <p><code>ent.prop</code> (sprememba vrednosti metapodatkov entitete z naštetimi nazivi atributov, ki se spreminjajo)</p> <p><code>ent.prm</code> (sprememba vrednosti metapodatkov hrambe fizičnega gradiva entitete z naštetimi nazivi in vrednostmi vseh atributov hrambe fizičnega gradiva)</p> <p><code>ent.secclass</code> (sprememba stopnje tajnosti na entiteti s prejšnjo in novo vrednostjo stopnje tajnosti ter razlogom za spremembo)</p> <p><code>auditlog.query</code> (vpogled v revizijsko sled z navedeno poizvedbo)</p> <p><code>dir.groupmem</code> (sprememba članstva navedene skupine uporabnikov z naštetimi spremembami (dodani, odstranjeni)</p> <p><code>dir.entauth</code> (sprememba prijavnih podatkov navedene entitete imenika - uporabnika)</p> <p><code>dir.entident</code> (sprememba evidenčnih podatkov navedene entitete imenika – uporabnika (ime, priimek, opis, ipd))</p> <p><code>dir.entstate</code> (sprememba stanja navedene entitete imenika – uporabnika (omogočen, onemogočen, izbrisan, ipd))</p>
RequiredParams:	<p>Nabor zahtevanih podatkov, ki jih mora posredovati odjemalec pri odpiranju seje ali dogodka. V kolikor manjka en sam zahtevan podatek, IMiS®/ARChive Server zavrne vzpostavitev seje ali dogodka. Veljaven nabor zahtevanih podatkov lahko vsebuje vsaj enega izmed:</p> <p><code>username</code> (uporabniško ime uporabnika, ki izvaja operacijo)</p> <p><code>computername</code> (ime računalnika, iz katerega operacija izvira)</p> <p><code>message</code> (razlog/sporočilo, ki ga uporabnik vpiše pri izvedbi operacije)</p> <p>Privzeta nastavitvev je <code>username</code>, <code>computername</code>.</p>

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: Predpomnjeni strežniški ključ, ki se uporabi za šifriranje avtentikacijskih sporočil in kasnejše komunikacije z odjemalcem, ki predstavlja pooblaščen 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*) pa 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).

<code>CryptoModes:</code>	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: <pre> 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 </pre>
<code>PreSharedKey:</code>	Predpomnjeni strežniški ključ, ki se uporabi za šifriranje avtentikacijskih sporočil (prijavna metoda <code>psk</code>) in kasnejše komunikacije z odjemalcem, ki predstavlja navadnega, nepriviligiranega IMiS® odjemalca.
<code>RequiredParams:</code>	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: <pre> username (uporabniško ime uporabnika, ki izvaja operacijo) computername (ime računalnika, iz katerega operacija izvira) message (razlog/sporočilo, ki ga uporabnik vpiše pri izvedbi operacije) </pre> Privzeta nastavitvev je prazno – ne vsebuje nobenega iz nabora.

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 pravočasnem 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 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 iarc status
Status of IMiS/ARChive HSM Storage Server: iarc (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 na 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 iarc status
Status of IMiS/ARChive HSM Storage Server: iarc (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 iarc status
Status of IMiS/ARChive HSM Storage Server: iarc (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 iarc 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 nezdržljivost strežnika z operacijskim sistemom.

Opis pogoste težave 5

Ob zagonu strežnika IMiS®/ARChive Server se na konzoli pojavi izpis:

```
[user1@iarc ~]# sudo service iarc 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 sosedju 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 iarc 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 <Timeout> 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 storitev pa javlja naslednji status:

```
[user1@iarc ~]# sudo service iarc status
Status of IMiS/ARChive HSM Storage Server: iarc 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 iarc on
[user1@iarc ~]#
```

Preverimo uspešnost izvedbe ukaza:

```
[user1@iarc ~]# sudo chkconfig iarc --list
iarc          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 iarc 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 iarc 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 receive 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 nastavitve 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 iarc status
Status of IMiS/ARChive HSM Storage Server: iarc (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,

in vse skupaj 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 iarc status
Status of IMiS/ARChive HSM Storage Server: iarc (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,

in vse skupaj poslati po e-pošti na naslov: `podpora@imis.si`.

Naše tehnično osebje nato 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 Server v 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 (nastavitve 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 obstoječem 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 po navadi 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 occurred. 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 po navadi 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 nastavitvev 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 zahteve.

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 formatted according to POSIX standard and cannot be used. Use format ll CC.CHARSET[@variant]].«**CRIT: »Locale parameter '<vrednost parametra>' for LC CTYPE in not formatted 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 pravih POSIX formata.

Glej nastavitve `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: »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 nastavitvev 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« nastavitvev revizijske sledi, čeprav do njih ni prišlo.

Trenutne nastavitve bodo označene kot spremenjene, saj storitev nima na voljo stanja nastavitvev 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 po navadi 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 po navadi 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 Server je 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, objekta IMiS®/ARChive Server 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 namestitvev, napačno nastavitvev 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 dešifriranja 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(<systemski identifikator>, IPC RMID) error <koda napake>: <opis systemske napake>«

Prišlo je do systemske 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 systemske 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 systemski 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 systemske 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 systemske 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 systemske 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 systemskih 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 kreiranju 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 takim 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 nastavitve 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 opsijskim dodatkom omrežnih vrat. Ločilo med omrežnim naslovom in omrežnimi vrati je znak »:«. Napaka pri nastavitvi produkta.

ERR: »Attribute 'destination' contains 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 števecv 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 smel 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 definiciji 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 zahtev 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 opis napake [glej Entity handle is invalid](#). zgoraj) 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 identifier (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 entity.«

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 kode.

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.«

Zahtevk 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 exceeds 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 exceeds 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 exceeds 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 exceeds 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 assign 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 '['<naslov>]:<port>' (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 podentitet. 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.«

Za spreminjanje je najprej potrebno entiteto odpreti.

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.«

Za dodajanje novih podrejenih entitet je potrebno odpreti entiteto.

ERR: »Destination status is 'Closed'. Adding new entities under it is not allowed.«

Kakršnokoli spreminjanje entitete s statusom »zaprto« 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 MiS®/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 nastavitvev šifrirnega podsistema odjemalca, oziroma neustrezna nastavitvev 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 nastavitvev 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 sistema 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 nastavitve 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 nastavitve 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.