

UNIVERZITET U NOVOM SADU FAKULTET TEHNIČKIH NAUKA U NOVOM SADU



Milan Kostić

Stefan Karać

Milan Mutić

Nemanja Rogić

Sigurnost i bezbednost u smart grid sistemima

Public key infrastructure system

-Predmetni projekat-

Profesor: Docent dr. Imre Lendak

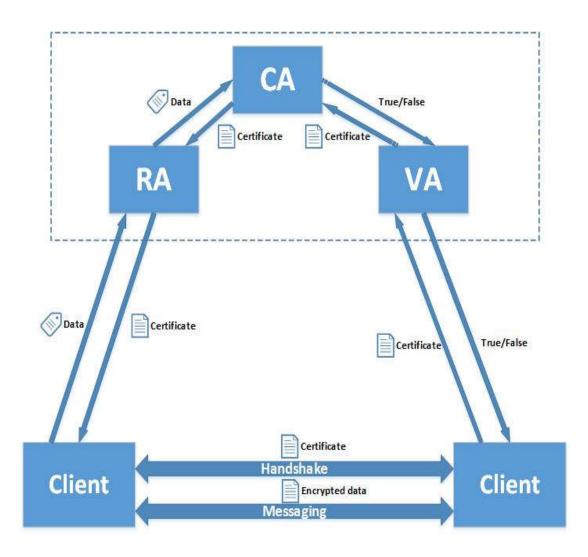
Asistent: Stefan Dejanović

Novi Sad, 2016

1. Arhitektura sistema

Tema rada je implementacija sistema (eng. Public key infrastructure system) za upravljanje sertifikatima. Sistem treba da omogući kreiranje sertifikata, izdavanje sertifikata klijentima (registracija klijenata) kao i validaciju sertifikata.

Sistem se sastoji iz tri komponente (Slika 1): Certification authority (u daljem tekstu CA), Registration authority (u daljem tekstu RA) i Validation authority (u daljem tekstu VA).



Slika 1. Public key infrastructure system

CA ima sledeće zadatke:

- Generisanje i izdavanje sertifikata
- Provera validnosti sertifikata
- Povlačenje sertifikata

CA sve izgenerisane sertifikate čuva trajno na masovnoj i u operativnoj memoriji. U slučaju da CA sistem postane nedostupan, postoji rezervni sistem sa istom funkcionalnošću koji se u najkraćem roku aktivira i stavlja na korišćenje. Svi podaci se zbog toga repliciraju sa aktivnog sistema na rezervni. CA određene akcije evidentira u Windows event log-u.

RA komponenta ima posrednu ulogu prilikom registracije klijenata. Vrši početne provere klijenata kao i delegiranje zahteva ka CA komponenti, koja treba da izvrši registraciju klijenata.

VA podsistem ima ulogu validacije klijenata, koji žele da uspostave međusobnu komunikaciju. Kao i RA komponenta VA prosleđuje klijentske zahteve ka CA podsistemu koji vrši proveru i validiranje klijenata.

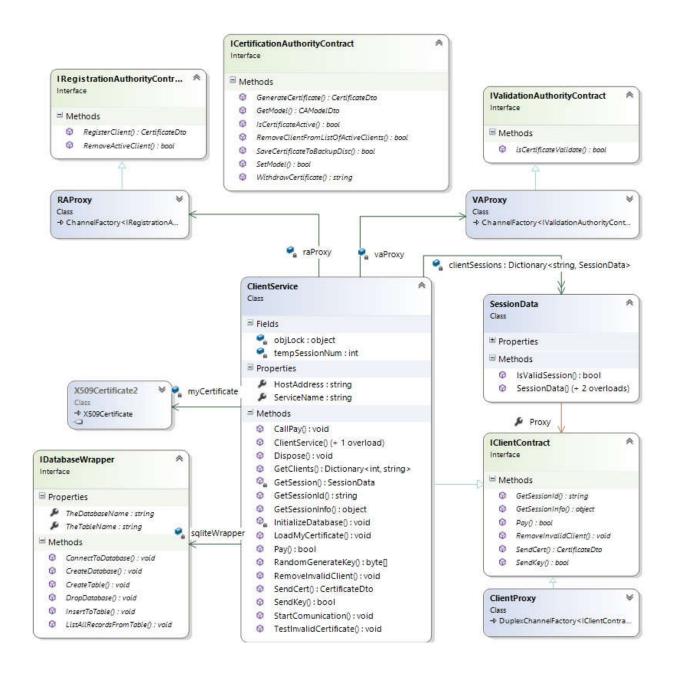
Da bi klijenti uspešno komunicirali potrebno je da prvo budu registrovani. Za registraciju se obraćaju RA komponenti. Prilikom uspostave komunikacije vrše obostranu validaciju sertifikata, tako što se obraćaju VA komponenti sa zahtevom validiranja sertifikata druge strane. Nakon uspšne validacije, klijenti razmenjuju ključ sesije koji će se koristiti za kriptovanje i dekriptovanje poruka u toj sesiji. Ključ sesije se razmenjuje asimetričnim algoritmom, dok se dalja komunikacija obavlja kriptovanjem poruka simetričnim algoritmom. Za kriptovanje poruka nakon bezbednog otvaranja komunikacionog kanala se koristi AES (Advanced encryption standard) algoritam u ECB obliku. Za razmenu ključa sesije se koristi asimetrični RSA algoritam.

Svaku uspešnu uspostavu komunikacije klijenti beleže u SQLite bazi podataka. Prilikom toga se upisuju podaci o vremenu i identifikaciji klijenta sa kojim je uspostavljena komunikacija.

1. Dizajn sistema

Slika 2 prikazuje ključne klase koje implementiraju opisane funkcionalnosti u prethodnom poglavlju. Svaka od pomenutih komponenti su nezavisne (CA, RA, VA, klijentska komponenta) aplikacije, odnosno WCF servisi.

.



Slika 2. Class diagram

CA komponenta nudi ICertificationAuthorityContract interfejs. Najznačajnije metode ovog servisa su metode za generisanje sertifikata i proveru važenja sertifikata (Listing 1). Kao što se sa dijagrama zaključuje ove metode pozivaju RA servis i VA servis na zahtev klijenata za registraciju, odnosno validaciju sertifikata.

RA klijentima stavlja na raspologanje IRegistrationAuthorityContract interfejs. Na listingu 2 je prikazana jedina značajna metoda RA servisa koja služi za registrovanje klijenata.

Za proveru validnosti sertifikata koristi se VA servis, kroz interfejs IValidationAuthorityContract (Listing 3). Za validaciju sertifikata klijenti će pozvati IsCertificateValid() metodu.

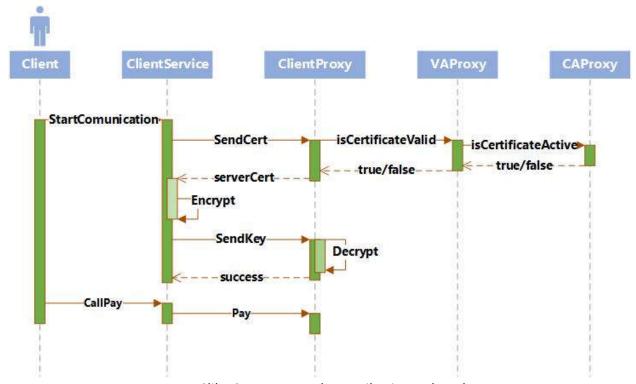
```
[ServiceContract]
public interface IValidationAuthorityContract
{
          [OperationContract]
          bool IsCertificateValid(X509Certificate2 certificate);
}
    Listing 3. Interfejs VA komponente
```

Svaka klijentska aplikacija može da se ponaša i kao klijent i kao servis u komunikaciji sa drugim klijentima. Za potrebe komunikacije klijent nudi IClientContract interfejs (Listing 4).

```
[ServiceContract(CallbackContract = typeof(IClientContract), SessionMode =
    SessionMode.Required)]
public interface IClientContract
{
    [OperationContract]
    bool Pay(byte[] message);
    [OperationContract]
    CertificateDto SendCert(CertificateDto cert);
```

```
[OperationContract]
bool SendKey(byte[] key);
.....
}
Listing 4. Interfejs klijentskog servisa
```

Tok uspostavljanja komunikacije između klijenata prikazuje dijagram na slici 3.



Slika 3. Uspostava komunikacionog kanala

Klijent inicira komunikaciju ka drugom klijentu tako što mu pošalje svoj sertifikat sa javnim ključem. Drugi klijent potom izvrši validaciju sertifikata tako što se obrati VA servisu. Ukoliko je sertifikat validan, klijent će vratiti svoj javni ključ, koji će takođe biti validiran kod VA servisa. Tim javnim ključem će prvi klijent kriptovati ključ sesije, a drugi klijent će isti dekriptovati svojim privatnim ključem. Nakon toga je uspostavljena komunikacija, pa će se sve ostale poruke kriptovati AES algoritmom.

AES algoritam je simetrični algoritam, koji kriptuje podatke podelom na blokove koji mogu biti različite dužine. U ovom ovom radu je implementiran u ECB obliku, što znači da pored ključa ne koristi dodatne podatke za kriptovanje. Ključ je dužine 128 bita, a i svaki blok je isto dužine 128 bita. Zbog boljih performansi, algoritam je paralelizovan tako da paralelno vrši kriptovanje/dekriptovanje blokova.

Veličina poruke (B)	Broj iteracija	Vreme - paralelno	Vreme - sekvencijalno
1000 B	100	2837 ms	1727 ms
5000 B	100	5800 ms	7331 ms
10000 B	100	9926 ms	14421 ms
30000 B	100	26795 ms	46863 ms

Tabela 1. Poređenje paralelne i sekvencijalne varijante AES algoritma

Rezultati testiranja su prikazani u tabeli 1. Testovi su vršeni na procesoru Intel Pentium 2 x 2.20 GHz. Iz ovih nekoliko testova se zaključuje da veličina poruke direktno utiče na brzinu izvršavanja algoritma, kada je paralelna obrada u pitanju. Kada je poruka prilično mala, u ovom slučaju samo 1 KB, paralelna obrada biva čak sporija u odnosu na sekvencijalnu. Kako veličina poruke raste, tako paralelna obrada postaje sve brža.

Za slučaj da se paralelizuju još neke operacije kao što je na primer dodavanje ključa runde, tada performanse značajno padaju. Na primer kod primene paralelizacije kod dodavanja ključa runde i poruka veličine 1 KB i 100 iteracija, paralelna obrada traje 15385 ms, a sekvencijalna slično tome 15308. To je velika razlika u odnosu na prvi test primer gde je paralelna obrada trajala 2837 ms a sekvencijalna 1727 ms.

2. Testiranje

2.1 Analiza enkriptovanja uz pomoć Wireshark programa

U okviru aplikacije enkripcija podataka se radila prilikom međusobne komunikacije klijenata. Kako bi utvrdili da su poruke koje se između njih razmenjuju zaista zaštićene upotrebili smo Wireshark program. U prvom delu testiranja Binding. Security. Mode je podešen na Transport, i na slici 4 se može videti kako izgleda razmenjena poruka (sa i bez enkripcije). U svim test slučajevima je sa jednog na drugi klijent slata poruka "pozdrav". Primećuje se da je u oba slučaja za Transport mod gotovo nemoguće izvući neku smisleniju poruku.

```
Bez enkripcije (Security = Transport)
      93 00 00 00 01 00 00 00 4d 8c 03 3a e4 4e 0a 53 ......M..:.N.S
0000
                                                      .....^-.+c<....
0010
      05 00 00 00 da ef 5e 2d d9 2b 63 3c f1 86 b3 1e
0020
      27 ad 88 3d 98 bl 1f cc 36 ed 33 36 89 fb a3 a6 '..=...6.36....
      15 d3 6a 87 8e e4 05 26 ee 6e 2e 8a 0f c4 9b 35
                                                      ..j....&.n.....5
0040
      49 93 b2 b3 66 65 f0 f5 53 12 a3 b8 87 9c 7a 11 I...fe..S....z.
      2c 95 4a f2 b3 17 a8 c4 e6 e4 75 1f 65 87 a0 28 ,.J.....u.e..(
0050
0060
      66 85 c0 58 ae 48 18 9b b5 63 8b 48 08 84 4c 0d f..X.H...c.H..L.
0070
      bf ac a5 e0 52 68 ad ba 93 97 06 99 21 0c 29 e5 ....Rh.....!.).
0800
      66 8b 17 01 85 6f 63 b7 f6 b5 00 d3 1f 92 ca 8b f....oc......
0090 53 e7 98 b6 45 ca 67
                                                      S...E.g
Sa enkripcijom(Security = Transport)
      93 00 00 00 01 00 00 00 39 2e 5c 38 37 d1 3b 47 ......9.\87.;G
      05 00 00 00 b6 31 45 44 f3 9c 9c ad f6 cb 8d 58 .....1ED......X
0010
      d2 9c 35 d7 23 5d 77 3e d8 0f cc 73 82 48 69 aa ..5.#]w>...s.Hi.
0020
0030
      a9 92 17 6e bc 08 f1 54 d1 54 76 10 a7 0f 63 6e
                                                      ...n...T.Tv...cn
0040
      43 50 da 93 b6 9b d8 e4 09 d9 65 54 ad 20 23 16 CP.....eT. #.
0050
      71 ee ff 55 6c a5 01 0f 2e 54 79 45 a5 18 64 12 q..Ul....TyE..d.
      59 5c a1 bf ae bc e1 20 06 32 4c d5 a2 a4 12 d0 Y\..... .2L.....
0060
0070
      c5 ea 7e 12 fb 06 2e 6b 49 fa 66 e9 d0 79 b5 8f ..~...kI.f..y..
      c6 1b 2f 8b 88 23 74 ff c8 73 d1 49 56 ac cc 27 ../..#t..s.IV...
0080
0090 ed 63 3d 1b 8c c1 59
                                                      .c=...Y
```

Slika 4. Razmenjena poruka za Transport mod

Na slici 5. Binding.Security.Mode je podešen na None. Razlika u odnosu na Transport je uočljivija jer se određeni podaci ipak mogu primetiti. Ukoliko nemamo enkripciju i poruka se razmenjuje bez ikakve zaštite možemo doći do otvorenog teksta i saznati šta su to klijenti razmenjivali. Ukoliko se poruka koja se razmenjuje kriptuje njen sadržaj se ne može pročitati, bez obzira i na None mod koji je postavljen.

```
Bez enkripcije(Security = None)
       06 99 02 33 26 68 74 74 70 3a 2f 2f 74 65 6d 70 ...3&http://temp
       75 72 69 2e 6f 72 67 2f 49 43 6c 69 65 6e 74 43 uri.org/IClientC
       6f 6e 74 72 61 63 74 2f 50 61 79 03 50 61 79 07 ontract/Pay.Pay.
0020
       6d 65 73 73 61 67 65 56 02 0b 01 73 04 0b 01 61 messageV...s...a
       06 56 08 44 0a 1e 00 82 ab 31 44 1a ad 71 3d 43 .V.D.....1D..q=C
                                                        .-..L..n.L]!.D,D
       8a 2d 93 b3 4c ab 15 6e eb 4c 5d 21 8f 44 2c 44
      2a ab 14 01 40 17 56 73 44 65 62 75 67 67 65 72 *...@.VsDebugger
       43 61 75 73 61 6c 69 74 79 44 61 74 61 08 41 68 CausalityData.Ah
             70 3a 2f 2f 73 63 68 65 6d 61 73 2e 6d 69 ttp://schemas.mi
       63 72 6f 73 6f 66 74 2e 63 6f 6d 2f 76 73 74 75 crosoft.com/vstu
00a0
       64 69 6f 2f 64 69 61 67 6e 6f 73 74 69 63 73 2f dio/diagnostics/
00ъ0
       73 65 72 76 69 63 65 6d 6f 64 65 6c 73 69 6e 6b servicemodelsink
00c0
       9f 3c b8 80 cf a3 d8 71 44 aa 17 8f d6 44 98 c8 .<....qD....D..
       6d 2d ce 24 fd 36 00 00 00 00 08 b4 ef 41 53 48 m-.$.6......ASH
00d0
       1e 4a bd 2d 2e 01 80 c8 c6 72 ef 68 ac 6a b9 47 .J.-...r.h.j.G
00e0
       bc 4f 82 4c 5a f4 39 33 95 21 00 09 00 00 44 0c .O.LZ.93.!...D.
00f0
0100
       1e 00 82 ab 03 01 56 0e 42 33 0a 07 42 35 9e 06 .....V.B3..B5..
       70 6f 7a 64 72 61 9f 01 76 01 01 01
0110
                                                        pozdra..v...
Sa enkripcijom(Security = None)
       06 a2 02 33 26 68 74 74 70 3a 2f 2f 74 65 6d 70 ...3&http://temp
0000
0010
       75 72 69 2e 6f 72 67 2f 49 43 6c 69 65 6e 74 43 uri.org/IClientC
       6f 6e 74 72 61 63 74 2f 50 61 79 03 50 61 79 07 ontract/Pay.Pay.
0020
0030
       6d 65 73 73 61 67 65 56 02 0b 01 73 04 0b 01 61 messageV...s...a
0040
       06 56 08 44 0a 1e 00 82 ab 31 44 1a ad 33 71 a5 .V.D.....1D..3q.
0050
       cb ac 54 17 4a aa 91 d5 fc 5e 01 44 9d 44 2c 44 ...T.J....^.D.D,D
2a ab 14 01 40 17 56 73 44 65 62 75 67 67 65 72 *...@.VsDebugger
0060
0070
       43 61 75 73 61 6c 69 74 79 44 61 74 61 08 41 68 CausalityData.Ah
       74 74 70 3a 2f 2f 73 63 68 65 6d 61 73 2e 6d 69 ttp://schemas.mi
0080
0090
       63 72 6f 73 6f 66 74 2e 63 6f 6d 2f 76 73 74 75 crosoft.com/vstu
00a0
       64 69 6f 2f 64 69 61 67 6e 6f 73 74 69 63 73 2f dio/diagnostics/
00b0
       73 65 72 76 69 63 65 6d 6f 64 65 6c 73 69 6e 6b servicemodelsink
00c0
       9f 3c b8 80 cf a3 ae f8 4e a4 93 63 57 41 b2 ee
                                                        .<....N..cWA..
00d0
       a5 49 c7 08 a9 0a 00 00 00 00 1b 0e c5 05 1a 30 .I................0
00e0
       5e 46 98 21 69 e6 2c 3a f5 25 b7 94 06 bd 31 80 ^F.!i.,:.%....1.
00f0
       a7 45 b2 7b 37 e5 69 4e 33 f9 00 09 00 00 44 0c .E.{7.iN3.....D.
0100
       1e 00 82 ab 03 01 56 0e 42 33 0a 07 42 35 9e 0f
0110
       15 dc 4f 3d 9f 9a cd 8b 79 b6 1e 80 92 ba c3 9f ..O=....y.....
0120
       01 be 01 01 01
```

Slika 5. Razmenjena poruka za None mod

3.2. Analiza performansi uz pomoć WinDbg-a i Performance Monitor-a

Uz pomoć alata WinDbg analiziran je scenario pokretanja CA servera i registrovanja novog klijenta korišćenjem RA komponente. Na slikama 6, 7 i 8 prikazano je stanje heap-a u ključnim trenucima.

```
Command - Pid 5128 - WinDbg:10.0.14321.1024 X86
                               132 System.Signature
132 System.Globalization.NumberFormatInfo
739c6984
                   1
739c652c
                                144 System Version
                   2
739c62bc
                               144 System. Globalization. CultureInfo
68708c94
                               144 System .Xml .XmlDictionaryReaderQuotas
                               144 System.ServiceModel.Channels.AddressingVersion 156 System.ServiceModel.WsrmFeb2005Dictionary
61d693d8
                   3
61d6ffd0
                               156 System.ServiceModel.EnvelopeVersion
61d691c8
739c4180
                               168 System. Threading. ThreadAbortException
739c10a4
                               180 System.Reflection.CustomAttributeRecord[]
                   1
739c0ef8
                               188 System.Collections.Generic.Dictionary 2+Entry[[Syste
<u>6a7652d0</u>
72d7726c
                  14
                               280 System.Xml.XmlQualifiedName
                               448 System UriParser+BuiltInUriParser
                  16
739c7664
                   3
                               468 System.Collections.Generic.Dictionary 2+Entry[[Syste
739c6d34
                   3
                               549 System.Byte[]
739c64f8
72d77224
                               616 System.Globalization.CultureData
                               952 System.Collections.Generic.Dictionary`2+Entry[[Syste
                 17
739c55d4
                              1076 System. Int32[]
739c5000
                              1680 System.RuntimeType
                              1800 System.Xml.XmlDictionaryString
2188 System.Char[]
68708c58
                 75
7<u>39047d8</u>
                 21
739c3e18
                343
                             14924 System String
                 59
                             35428 System.Object[]
 73980cb
Total 799 objects
                                                                                               >
0:003>
```

Slika 6. Stanje heap-a neposredno pre hostovanja CA servisa

```
Σ
                       Command - Pid 5128 - WinDbg:10.0.14321.1024 X86
                                   Jystem.Collections.JpcclaffZed.MameObjectCollectionE
739d0594
                155
                              6820 System Signature
72d63250
                 32
                              8320 System.Net.Sockets.SocketAsyncEventArgs
                424
                              8480 System.Configuration.ConfigurationValue
6a1f7b68
739cfcb0
                104
                              8736 System.RuntimeType+RuntimeTypeCache
                              8848 System Collections.ArrayList+ArrayListEnumeratorSimp
<u>739d33d8</u>
                316
                203
                              9744 System.Configuration.ConfigurationValues
6a1f7a00
739aa21c
                226
                              9944 System.Reflection.Emit.OpCode
                            10608 System ServiceModel_Configuration.ExtensionElement
                156
6<u>1d5d8a4</u>
<u>739d63c0</u>
                204
                            10808 System.Collections.DictionaryEntry[]
739c5000
                486
                            13608 System.RuntimeType
                            13632 System Collections ArrayList
                568
739c13bc
739c10a4
                            16692 System.Reflection.CustomAttributeRecord[]
                732
739c60f0
                350
                            18200 System Collections Hashtable
                118
                             22184 System.Collections.Generic.Dictionary 2+Entry[[System.collections.generic.Dictionary 2+Entry]
739b5c58
                            22344 System.Configuration.ConfigurationProperty 24120 System.Reflection.RuntimeMethodInfo
6a1f7304
                399
739cf 0d4
                402
739c6d34
                181
                            34252 System.Byte[]
                             91126 System Char[]
739c47d8
                372
739c77fc
                419
                           124824 System.Collections.Hashtable+bucket[]
739c55d4
                           127548 System Int32[]
                618
7<u>3980cbc</u>
               3086
                           162944 System.Object[]
739c3e18
               5814
                           443088 System String
Total 24253 objects
0:007>
```

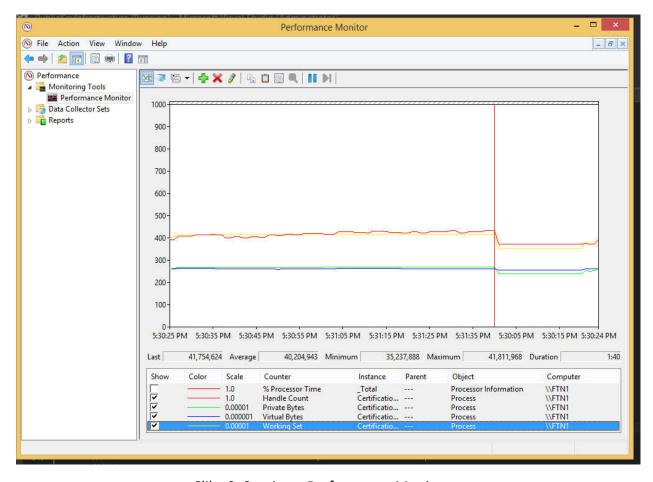
Slika 7. Stanje heap-a neposredno nakon hostovanja CA servisa

```
2
                        Command - Pid 5040 - WinDbg:10.0.14321.1024 X86
                              4040 Jystem.Runtimenethodiniostub
028dca30
                              5232 Org.BouncyCastle.Asn1.DerObjectIdentifier
5940 System.Configuration.ConfigurationValue
                327
6a1f7b68
                297
6a1f7820
                114
                              6384 System.Configuration.FactoryRecord
61d5d8a4
                 94
                              6392 System.ServiceModel.Configuration.ExtensionElement
6a1f7a00
                138
                              6624 System.Configuration.ConfigurationValues
68708c58
                277
                              6648 System.Xml.XmlDictionaryString
739c13bc
                310
                              7440 System.Collections.ArrayList
                              8580 System Signature
739d0594
                195
72d63250
                 35
                              9100 System. Net. Sockets. SocketAsyncEventArgs
<u>739aa21c</u>
                226
                              9944 System.Reflection.Emit.OpCode
                196
                             10192 System.Collections.Hashtable
739c60f0
                             12172 System Reflection RuntimeParameterInfo
739d0878
                179
                295
<u>6a1f7304</u>
                             16520 System.Configuration.ConfigurationProperty
739c5000
                629
                             17612 System.RuntimeType
                             18300 System.Reflection.RuntimeMethodInfo
739cf 0d4
                305
739<u>c55d4</u>
                             21440 System.Int32[]
                413
                 55
<u>00f61a20</u>
                             43110
                                         Free
739c77fc
                197
                             74448 System.Collections.Hashtable+bucket[]
               1535
                            115976 System.Object[]
73980cbc
                            176126 System.Char[]
295677 System.Byte[]
739c47d8
                141
<u>739c6d34</u>
                244
739c3e18
               4462
                            538258 System String
Total 16357 objects
0:010>
```

Slika 8. Stanje heap-a nakon uspešnog registrovanja novog klijenta

Na osnovu prikazanih stanja heap-a može se uočiti porast broja zauzetih objekata prilikom inicijalnog hostovanja CA servera. Porast broja zauzetih objekata se javlja kao posledica instanciranja klasa CAProxy i CertificationAuthorization. Međutim, nakon uspešnog registrovanja novog klijenta (slika 8) dolazi do primetnog smanjenja broja zauzetih objekata na heap-u iz razloga što su u ovom trenutku dealocirani objekti koji su prethodno služili za inicijalizaciju CA servera.

Sledeći test u okviru kojeg su analizirane performanse CA servera podrazumeva nekoliko uzastopnih registracija i validacija klijenata u određenom vremenskom intervalu (pogledati sliku 9). Pomoću alata Performance Monitor analizirane su promene sledećih vrednosti: handle count, private bytes, virtual bytes i working set. Kao što se može primetiti vrednosti private bytes i virtual bytes uz manje oscilacije ostaje manje-više nepromenjene dok se handle count i working set vraćaju na početne vrednosti prilikom gašenja CA servera.



Slika 9. Stanje na Performance Monitor-u