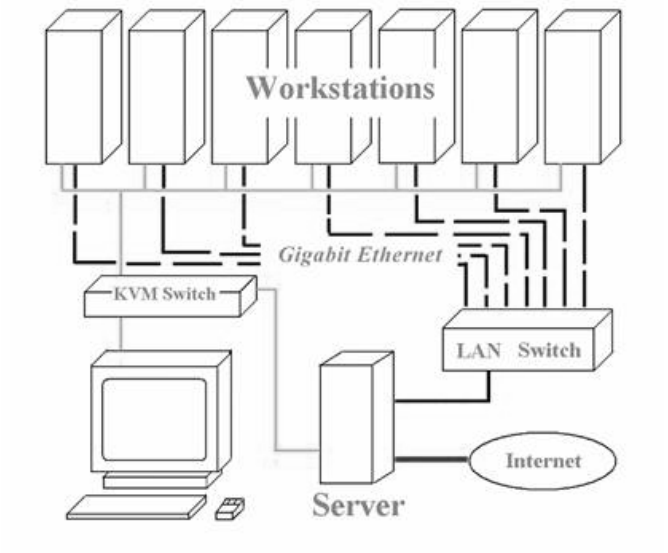
BEZBEDNOST RAČUNARSKOG GRIDA NA NIVOU MREŽE

Marko Dimitrijević
Elektronski fakultet, Niš

Sadržaj – *U ovom radu će biti predstavljeni mehanizmi kojima se obezbeđuje zaštita integriteta podataka i resursa računarskog grida. Računarski grid predstavlja distribuirani sistem geografski udaljenih klastera (sajtova) udruženih u jedinstven sistem sa deljenim resursima. Ovakav sistem mora imati mehanizme za prijavljivanje korisnika i identifikaciju njegovih prava, kao i odgovarajuće metode zaštite od neovlašćenog pristupa preko računarske mreže.*

1. UVOD

Razvoj Interneta kao globalne svetske računarske mreže i WAN linkova velike propusne moći omogućio je nastanak računarskog grida. Ime računarski grid je nastalo po analogiji sa električnim mrežom (*electrical power grid*) – kao što koristimo električnu energiju iz mreže bez znanja o poreklu, tako se računarski resursi u računarskom gridu mogu dobiti na zahtev, pri čemu krajnji korisnik ne zna ko je te resurse obezbedio i gde se oni fizički nalaze [1]. Slično kao i kod razvoja Interneta, grid je u početku bio u akademskim okvirima, ali vremenom postaje široko prihvaćen koncept. Jedna od najčešće korišćenih definicija predstavlja grid kao hardversku i softversku infrastrukturu koja pruža pouzdan, konzistentan i jeftin pristup računarskim resursima visokih performansi, omogućivši pristup na zahtev procesorskim resursima, podacima i servisima [2].



Slika 1. *Prikaz jednog klastera - sajta*

U osnovi, grid omogućava pristup resursima putem WAN linkova i često se definiše kao „klaster klastera“, tj. saradnja geografski distribuiranih klastera (sajtova) na kojima se izvršavaju korisnički procesi ili skladište korisnički podaci. Ovakva definicija opisuje hardversku komponentu i bazičnu hijerarhiju grida: na najnižem nivou se nalaze pojedinačni PC računari – kompjuterski nodovi (*computer nodes*), više nodova čine lokalni računarski klaster – sajt (slika 1.), a više geografski odvojenih sajtova računarski grid.

Očigledno je da hardverska struktura grida može biti izuzetno heterogena: veliki broj različitih sajtova sadrži različit broj nodova drugačijih procesorskih performansi, veličine operativne memorije, LAN ili WAN kapaciteta i prostora za skladištenje podataka.

Funkcija softverske komponente je da obezbedi distribuirane servise za pokretanje i manipulaciju korisničkih procesa, transfer podataka, pristup bazama podataka i monitoring. Softverski deo grida se sastoji od dva sloja: operativnog sistema i *middleware*-a. Osim osnovnih funkcija, softverska komponenta mora da obezbedi integritet podataka i raspodelu u višekorisničkom okruženju.

Pitanje bezbednosti računarskog grida je od posebnog značaja, imajući u vidu činjenicu da je sistem distribuiran, da se za prenos podataka koriste nesigurni linkovi i da resursima pristupaju korisnici koji pripadaju različitim institucijama. Bezbednost se može posmatrati na nivou noda (računara), na nivou mreže ili arhitekturnom nivou, kao i na nivou prava korišćenja resursa.

2. BEZBEDNOST NA NIVOU NODA

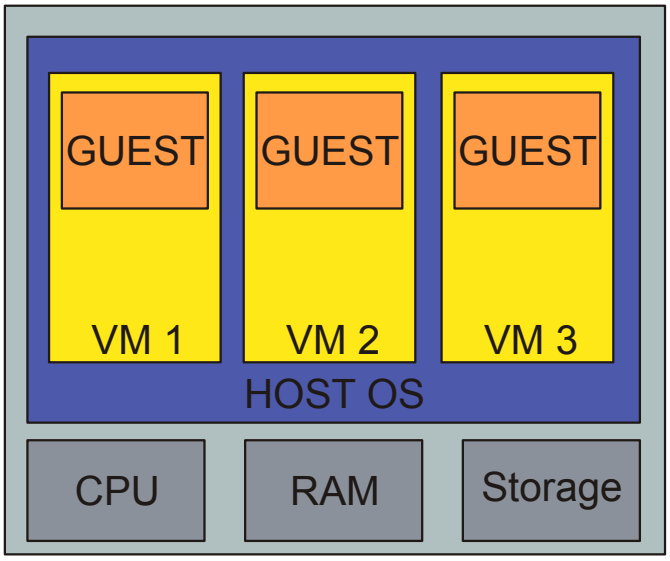
Bezbednost na nivou noda (računara) predstavlja zaštitu integriteta podataka pojedinačnog elementa u okviru računarskog grida. Osnovni metodi bezbednosti su slični kao kod pojedinačnih računara.

Korisnički zadaci (*jobs*) se predaju računarskom elementu (*computing element – CE*), koji prosleđuje zadatak na izvršavanje kompjuterskom nodu (*worker node – WN*) u okviru sajta. Zadaci – korisničke aplikacije – pristupaju podacima i resursima, i mogu predstavljati štetan kod (virus), tako da je potrebno obezbediti mehanizme za zaštitu integriteta podataka i zaštitu od nepravilne upotrebe resursa (*job starvation*).

Postoji nekoliko pristupa u rešavanju postavljenih bezbedonosnih problema. Najčešće primenjivani metod je *user-level sandboxing*, koji omogućava izolovanje korisničkih procesa u prostoru u kome se lako mogu zabraniti ili pratiti određeni sistemski pozivi koji bi mogli kompromitovati bezbednost sistema. Ovakav mehanizam se oslanja na standardni sistem zaštite kod UNIX/LINUX sistema i postojeće *batch* servere. Udaljeni korisnici koji pripadaju određenim virtuelnim organizacijama (VO) su mapirani u lokalne korisnike sa određenim pravima pristupa nad podacima i resursima sistema. Prednost ovakvog načina je jednostavnost i kompatibilnost sa postojećim sistemima bezbednosti.

Drugi način je *aplication-level sandboxing*, koji zahteva postojanje kôda za potvrdu autentičnosti korisničke aplikacije – zadatka koji se izvršava na gridu (*proof-carryng code* – PCC). Ovakav kôd se proverava pre izvršavanja, i u slučaju pozitivne identifikacije prosleđuje na izvršavanje. Prednosti ovakvog načina zaštite su što efikasno sprečava izvršavanje malicioznih aplikacija, a za implementaciju mehanizma je odgovoran sam autor aplikacije koja se izvršava na gridu [3].

Treći način je primena virtuelizacije, koja omogućava kreiranje *virtuelne mašine* (VM) (slika 3.), koja ostavlja utisak računara potpuno izolovanog od ostatka sistema. Na jednom fizičkom računaru – nodu – se mogu postaviti više virtuelnih mašina različitog ili istog tipa, čime se postiže efikasnija raspodela resursa. Kod virtuelizacije razlikujemo *host* operativni sistem koji je iznad hardverskog sloja i *guest* operativni sistem koji se izvršava u virtuelnoj mašini – aplikaciji koja se izvršava na host sistemu [4]. Model ne zahteva promenu *host* operativnog sistema. Glavna prednost je u smislu bezbednosti i fleksibilnosti, jer se krajnjem korisniku stavlja na raspolaganje celokupan *guest* sistem, koji je ipak izolovan i ne može kompromitovati *host* sistem.



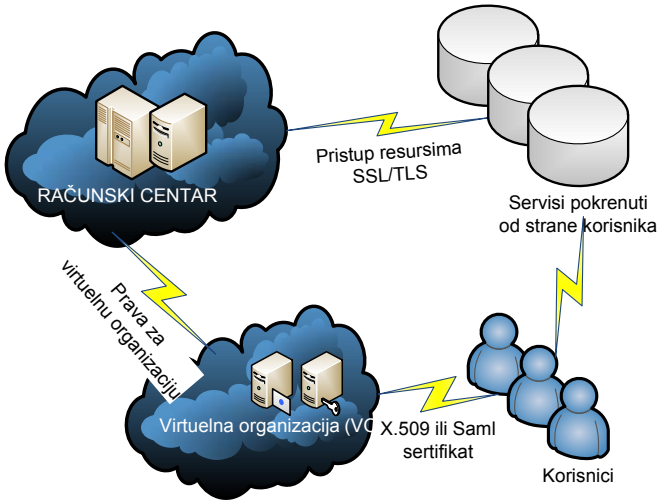
Sl. 2. Prikaz virtuelne mašine

Veliki nedostatak virtuelizacije predstavlja veliki *overhead* sistema i degradaciju performansi. Prilikom projektovanja softvera za virtuelizaciju se uvode brojne optimizacije kako bi se performanse poboljšale. Sličan je i model *paravirtuelizacije*, koji je baziran na modifikaciji host operativnog sistema, čime je postignuta kompatibilnost i poboljšanje performansi [5]. Koncept virtuelizacije prelazi sa softverskog na hardverski nivo; savremeni mikroprocesori imaju ugrađenu podršku za virtuelizaciju koja obezbeđuje bolju upravljivost bolje i performanse.

Četvrti koncept je bezbednost na nivou jezgra (kernela) operativnog sistema – pristup fizičkim resursima na nivou aplikacije kontroliše deo jezgra operativnog sistema (*exokernel*) [6].

3. BEZBEDNOST NA NIVOU ARHITEKTURE

Bezbednost na nivou mreže ili arhitekturnom nivou podrazumeva bezbednost sistema u celini. Računarski grid je geografski distribuiran sistem koji koristi internet kao osnovnu infrastrukturu za razmenu podataka i podložan je svim rizicima otvorenog sistema. U okviru ovog rada biće reči o bezbednosti podataka, politici pristupa resursima grida i zaštiti od DoS napada.



Sl. 3. GSI implementacija

Integritet i zaštita podataka su obezbeđeni implementacijom *Grid Infrastucture Secuty* – GSI (slika 3) [7]. GSI je model bezbedne komunikacije između nodova u okviru računarskog grida koji koristi sistem javnih ključeva za kriptografiju (*Public Key Infrastructure* – PKI).

Osnovni zahtevi prilikom implementacije GSI koncepta su potreba za sigurnom komunikacijom, podrška za bezbednu komunikaciju van granica odgovornosti organizacije, centralizovana kontrola pristupa i mogućnost jednostavnog „logovanja“ korisnika sa odgovarajućim pravima na resurse i podatke.

Osnovni koncept u GSI autentifikaciji predtsavlja sertifikat. Svaki korisnik i resurs grida ima svoj digitalni sertifikat na osnovu kojeg se identifikuje i on sadrži relevantne podatke za pozitivnu identifikaciju i autentifikaciju u sistemu. GSI sertifikat ima četiri osnovne informacije:

- Ime subjekta, koja identifikuje osobu ili objekat koji sertifikat predstavlja,
- Javni ključ koji subjektu pripada,
- Sertifikaciono telo (*Certificate Authority* – CA) koje je potpisalo sertifikat i garantuje identitet subjekta i validnost odgovarajućeg javnog ključa,
- Digitalni potpis sertifikacionog tela.

Sertifikaciono telo u ovom slučaju predstavlja treću stranu u komunikaciji koje sertifikuje vezu između javnog ključa i subjekta u sertifikatu. Da bi se verovalo određenom sertifikatu, neophodno je da CA sertifikat takođe bude nesporan, pa veza između CA subjekta i CA sertifikata mora biti ostvarena drugim, nekriptografskim, sredstvima.

GSI sertifikati su kodirani u X.509 formatu, koji predstavlja standard postavljen od strane *Internet Engineering Task Force* (IETF), i kompatibilni su sa ostalim softverom koji podržava PKI [8]. Primer X.509 sertifikata je dat na slici 4.

Ukoliko dve strane u komunikaciji imaju sertifikate i veruju sertifikacionim telima koja su ih potpisala, onda obe strane mogu potvrditi međusobni identitet.

```
Certificate:
   Data:
       Version: 1 (0x0)
       Serial Number: 7829 (0x1e95)
       Signature Algorithm: md5WithRSAEncryption
       Issuer: C=ZA, ST=Western Cape, L=Cape Town, O=Thawte Consulting cc,
               OU=Certification Services Division,
               CN=Thawte Server CA/emailAddress=server-certs@thawte.com
       Validity
           Not Before: Jul  9 16:04:02 1998 GMT
           Not After : Jul  9 16:04:02 1999 GMT
       Subject: C=US, ST=Maryland, L=Pasadena, O=Brent Baccala,
                OU=FreeSoft, CN=www.freesoft.org/emailAddress=baccala@freesoft.org
       Subject Public Key Info:
           Public Key Algorithm: rsaEncryption
           RSA Public Key: (1024 bit)
               Modulus (1024 bit):
                   00:b4:31:98:0a:c4:bc:62:c1:88:aa:dc:b0:c8:bb:
                   33:35:19:d5:0c:64:b9:3d:41:b2:96:fc:f3:31:e1:
                   66:36:d0:8e:56:12:44:ba:75:eb:e8:1c:9c:5b:66:
                   70:33:52:14:c9:ec:4f:91:51:70:39:de:53:85:17:
                   16:94:6e:ee:f4:d5:6f:d5:ca:b3:47:5e:1b:0c:7b:
                   c5:cc:2b:6b:c1:90:c3:16:31:0d:bf:7a:c7:47:77:
                   8f:a0:21:c7:4c:d0:16:65:00:c1:0f:d7:b8:80:e3:
                   d2:75:6b:c1:ea:9e:5c:5c:ea:7d:c1:a1:10:bc:b8:
                   e8:35:1c:9e:27:52:7e:41:8f
               Exponent: 65537 (0x10001)
   Signature Algorithm: md5WithRSAEncryption
       93:5f:8f:5f:c5:af:bf:0a:ab:a5:6d:fb:24:5f:b6:59:5d:9d:
       92:2e:4a:1b:8b:ac:7d:99:17:5d:cd:19:f6:ad:ef:63:2f:92:
       ab:2f:4b:cf:0a:13:90:ee:2c:0e:43:03:be:f6:ea:8e:9c:67:
       d0:a2:40:03:f7:ef:6a:15:09:79:a9:46:ed:b7:16:1b:41:72:
       0d:19:aa:ad:dd:9a:df:ab:97:50:65:f5:5e:85:a6:ef:19:d1:
       5a:de:9d:ea:63:cd:cb:cc:6d:5d:01:85:b5:6d:c8:f3:d9:f7:
       8f:0e:fc:ba:1f:34:e9:96:6e:6c:cf:f2:ef:9b:bf:de:b5:22:
       68:9f
```

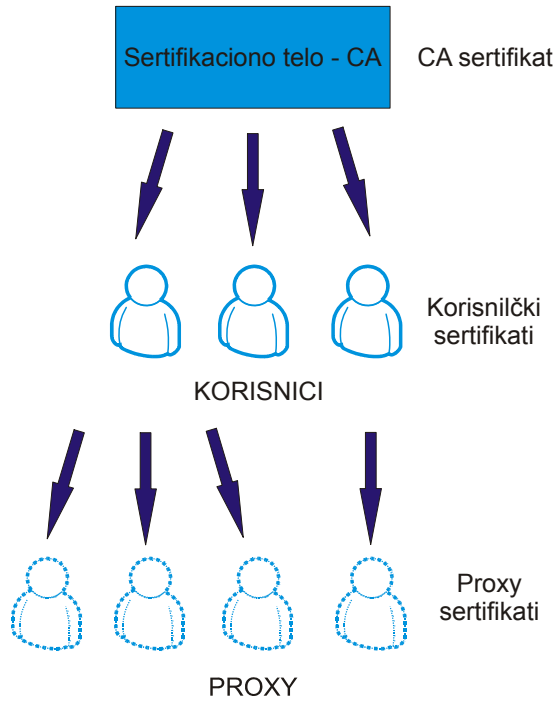
Sl. 4. Primer X.509 sertifikata

Obe strane u komunikaciji moraju imati odgovarajuće validne CA sertifikate, kako bi mogle potvrditi autentičnost druge strane, odnosno da li je odgovarajući sertifikat potpisan od strane CA. GSI koristi *Secure sockets layer* (SSL) protokol za međusobnu autentifikaciju. SSL je implementiran na transpornom nivou OSI sloja, tako da ga mogu koristiti različite aplikacije. SSL omogućava sigurnu komunikaciju i sprečava prisluškivanje komunikacije (*eavesdropping*) i *man-in-the-middle* napade. Primena SSL protokola unosi *overhead* u komunikaciji, tako da GSI neće podrazumevano uspostaviti siguran kanal između učesnika, već samo u slučaju potrebe.

Privatni ključ neophodan za enkripciju/potpisivanje je smešten u lokalnom fajl-sistemu i zaštićen lozinkom koju korisnik mora saopštiti GSI softveru (*Globus Toolkit*) kako bi pristupio resursima. Privatni ključ mora biti zaštićen i pravima pristupa na nivou fajl sistema koji onemogućuju njegovo čitanje od strane drugih korisnika. Određene verzije GSI softvera ne funkcionišu ukoliko prava pristupa nisu korektno postavljena (najčešće r--------). Bolja zaštita privatnog ključa se može ostvariti upotrebom kriptografskih *smart* kartica, na kojima se nalazi privatni ključ koji je na taj način fizički izolovan od sistema na kome bi mogao biti zloupotrebljen.

GSI omogućava delegaciju prava, čime se smanjuje potreba za stalnom proverom identiteta korisnika, odnosno broj unosa lozinke kako bi se prava korisnika ostvarila. Ukoliko se zahteva korišćenje većeg broja različitih resursa ili servisa koji se izvršavaju u korisnikovo ime (i sa njegovim privilegijama), potreba za višestrukom identifikacijom putem lozinke se može izbeći kreiranjem *proxy*-a (zastupnika).

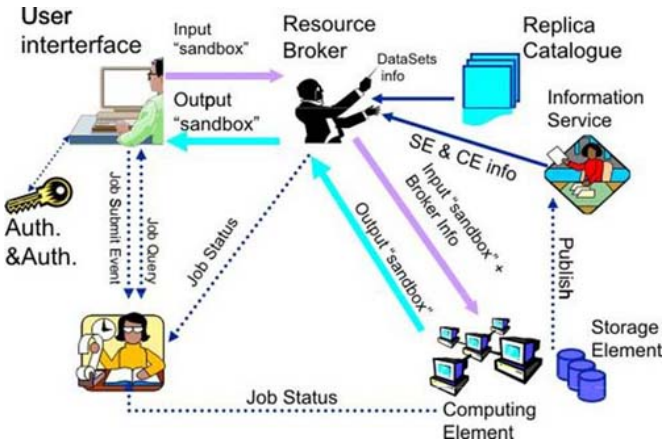
Proxy se sastoji od novog sertifikata koji sadrži korisnikov identitet sa nezavisnim parom ključeva. Novi sertifikat je potpisan od strane samog korisnika koji ga kreira, i vremenski je limitiran. Koncept potpisivanja u okviru jednog sertifikacionog tela je prikazan na slici 5.



Sl. 5. Potpisivanje sertifikata. Provera potpisa ide u obrnutom smeru

Privatni ključ proxy-a takođe mora biti čuvan na sigurnom mestu, ali imajući u vidu njegovo ograničeno vreme

trajanja, nije neophodno postupati kao sa korisničkim ključevima. Moguće je čuvati ključ u lokalnom fajl sistemu u neenkriptovanoj formi, pošto privilegije definisane u fajl sistemu onemogućavaju njegovo čitanje. Korisnik može koristiti sertifikat kreiranog proxy-a za sigurnu komunikaciju u njegovom vremenu trajanja. Prilikom autentifikacije u ovom slučaju, druga strana u komunikaciji prima proxy i korisnički sertifikat, čijim se javnim ključem proverava validnost potpisa proxy sertifikata. Javni ključ CA koji je smešten u fajl sistemu druge strane služi za proveru potpisa korisničkog sertifikata. Privatni ključ korisnika se koristi samo za potpisivanje proxy sertifikata, tako da je potrebno samo jedno prosleđivanje lozinke GSI sistemu. Delegacija prava nije standardno implementirana u SSL protokol, i koriste je samo GSI aplikacije (*Globus Toolkit*, GSI-SSH i GridFTP). Shematski prikaz toka podataka u računarskom gridu je prikazan na slici 6.



Sl. 6. Shematski prikaz toka podataka u računarskom gridu

Politika pristupa resursima i servisima računarskog grida je koncipirana oko *virtuelnih organizacija* (VO). Virtuelne organizacije predstavljaju grupe korisnika koji pripadaju različitim institucijama, geografskim područjima i administrativnim domenima, sa različitim pravima pristupa i korišćenja resursa. Korisnici u okviru jedne VO mogu imati različite uloge (*roles*) ili pripadati različitim aplikacionim grupama. Prava pristupa na nivou virtuelne organizacije se mogu implementirati na različite načine i ovde će biti opisana *Virtual Organization Membership Service* – VOMS implementacija [9]. VOMS server predstavlja bazu podataka u kojoj su korisnici asocirani sa njihovim virtuelnim organizacijama.

Scenario pristupa resursima je sledeći:

- korisnik kontaktira VOMS server koji vrši njegovu autentifikaciju na osnovu korisničkog sertifikata;
- korisnik podnosi zahtev VOMS serveru za konkretnu aplikacionu grupu ili ulogu;
- VOMS server generiše atributni sertifikat, pomoću koga korisnik kreira proxy kojim pristupa resusrima grida.

Zaštita od različitih DoS (*Denail of Service*) napada se bitno ne razlikuje od ostalih otvorenih sistema (servera). Zaštita se može ostvariti na nivou jednog noda primenom odgovarajućih preventivnih mera: redovno ažuriranje softverskih komponenti, implementacija lokalnog *firewall*-a, korišćenjem restriktivnog pristupa, itd. Na nivou mreže je takođe moguće primeniti *firewall* koji će štititi odgovarajući mrežni segment, imajući u vidu sve servise koji su neophodni za funkcionisanje grida u celini i odgovarajuće portove koji moraju biti prohodni.

4. ZAKLJUČAK

Računarski grid predstavlja kompleksnu distribuiranu strukturu sa mnoštvom korisnika koji poseduju različita prava korišćenja resursa i uloge. Bezbednost celog sistema se implementira na nekoliko nivoa. Implementacija bezbedonosnih mehanizama u ovakav sistem se bazira na već postojećim rešenjima (SSL, X.509) kojima su pridodate ekstenzije koje omogućuju funkcionisanje u distribuiranom sistemu (GSI, VOMS).

5. LITERATURA

[1] I. Foster, C. Kesselmann, S. Tuecke, "The Anatomy of the Grid, Enabling Scalable Virtual Organizations", International J. Supercomputer Applications, 2001.

[2] I. Foster, C. Kesselmann, "The Grid: Blueprint for a New Computing Infrastructure", 1998.

[3] A. Chakrabarti, A. Damodaran, S. Sengupta, "Computer Grid Security, A Taxonomy", IEEE Security & Privacy, Vol. 6, No 1, January 2008.

[4] VMware virtualization softvare, www.vmware.com.

[5] P. Barham, "Xen and the Art of Virtualization", ACM Symposium of Operating System Principles, ACM Press, pp. 164-177, 2003.

[6] http://en.wikipedia.org/wiki/Exokernel.

[7] I. Foster, C. Kesselman, G. Tsudik, S. Tuecke, "A Security Architecture for Computational Grids", ftp://ftp.globus.org/pub/globus/papers/security.pdf.

[8] C. Adams, S. Ferrell, "Internet X.509 Public Key Infrastructure", IETF RFC 2510, http://tools.ietf.org/html/rfc2510, 1999.

[9] R. Alfieri at al. "From Gridmap-File to VOMS: Managing Authorization in a Grid Environment", Future Generation Computer Systems. Vol. 21, no. 4, pp. 549-558. Apr. 2005.

Abstract – In this paper we will present methods that provide data and resource security of computer grid. Computer grid represents geographically distributed system of clusters (sites) joined into unique system with shared resources. It must have appropriate methods for logging and identification, as well as appropriate methods for unauthorized computer network access.

SECURITY OF COMPUTER GRID AT NETWORK LEVEL

Marko Dimitrijević