ROLE PENGGUNA DALAM PENDATAAN DATA PENDUDUK MISKIN

Asrianda
Teknik Informatika Universitas Malikussaleh
Kampus Bukit Indah, Lhokseumawe
e-mail : 4srianda@gmail.com

Abstrak

Pembatasan akses dapat dilakukan bagi pengguna yang berwenang, mekanisme dibuat untuk memastikan bahwa informasi tersedia bagi pengguna yang diizinkan untuk mengaksesnya. Kebijakan hak akses yang dilakukan oleh pengguna untuk menjaga keamanan dalam sebuah sumber daya kadangkala membutuhkan biaya mahal. Pelaksanaan untuk menyederhanakan kebijakan akses ke sumber daya akan berdampak positif dalam penghematan biaya yang akan dikeluarkan. Perhitungan rangkaian ranah otoritas yang tersedia dalam hak akses dalam mengimplementasikan DSD. Hak akses dipisahkan. Hak akses negatif memiliki tindakan negatif ditetapkan pada suatu objek yang menggantikan hak akses positif dan memiliki kegiatan positif sesuai dengan ketetapan pada objek yang sama sebagai hak akses negatif.

Kata Kunci : hak akses, pengguna, informasi, informasi

1. PENDAHULUAN

Membatasi akses informasi dapat dilakukan pengguna dengan memanfaatkan mekanisme hambatan hak akses pengguna, berguna mencegah pencurian informasi. Pengguna dapat mendeteksi terjadinya kesalahan dalam melakukan akses informasi. Pencegahan lebih bermanfaat dari pada melakukan pendeteksian, jika terjadi pencurian informasi. Kontrol akses menyediakan sarana untuk mengontrol sistem informasi yang memiliki akses ke sumber daya. Pembatasan akses dapat

dilakukan bagi pengguna yang berwenang, mekanisme dibuat untuk memastikan bahwa informasi tersedia bagi pengguna yang diizinkan untuk mengaksesnya.

Keamanan sumber daya komputasi membutuhkan biaya yang mahal juga memerlukan pengawasan yang rumit. Administrator harus melakukan pengontrolan yang sangat ketat dan dapat menentukan setiap pengguna yang mana saja berhak mendapatkan hak akses untuk mengakses suatu sumber daya. Pengguna akan diberikan *password* dan adminstrator dapat memperbaharui hak perizinan pengguna untuk mengakses ke sumber daya. Dalam menerapkan suatu kebijakan hak akses yang dilakukan oleh pengguna untuk menjaga keamanan dalam sebuah sumber daya kadangkala membutuhkan biaya mahal. Pelaksanaan untuk menyederhanakan kebijakan akses ke sumber daya akan berdampak positif dalam penghematan biaya yang akan dikeluarkan.

Sebagian besar pihak melakukan manipulasi data dalam sebuah sistem dilakukan pihak internal di organisasi tersebut, mencegah terjadinya manipulasi data dapat dilakukan dengan melakukan pengaturan hak akses bagi pengguna. Kontrol akses merupakan cara mencegah ancaman keamanan internal. *Role Based Access Control* (RBAC) merupakan mekanisme pengelolaan sejumlah besar hak akses pada basis data berukuran besar yang fleksibel. Dibandingkan model kontrol akses tradisional yaitu *Mandatory Access Control* (MAC) dan *Discretionary Access Control* (DAC) (Habib, 2011).

Dalam menjaga keamanan sumber daya komputasi akan membutuhkan biaya yang sangat mahal juga memerlukan pengawasan yang sangat rumit. Administrator harus melakukan pengontrolan yang sangat ketat dan dapat menentukan setiap pengguna yang mana saja berhak mendapatkan hak akses untuk mengakses suatu sumber daya.

Pengguna akan diberikan password dan adminstrator dapat memperbaharui hak perizinan pengguna untuk mengakses ke sumber daya. Dalam menerapkan suatu kebijakan hak akses yang dilakukan oleh pengguna untuk menjaga keamanan dalam sebuah sumber daya kadangkala membutuhkan biaya mahal. Pelaksanaan untuk menyederhanakan kebijakan akses ke sumber daya akan berdampak positif dalam penghematan biaya yang akan dikeluarkan.

Kontrol akses memegang peranan yang sangat penting dalam menjaga keamanan sumber daya maupun sistem, supaya tidak dipakai maupun digunakan oleh pihak yang tidak diinginkan. kontrol akses dapat melakukan suatu kebijakan dalam komponen perangkat lunak, maupun komponen perangkat keras yang berguna untuk membatasi hak akses ke sumber daya. Dalam menjaga keamanan sistem tidak cukup hanya dengan menggunakan password, sidik jari atau yang lainnya tetapi dengan cara membatasi hak apa saja yang dapat diakses oleh pengguna tersebut sehingga sistem dapat memberikan izin kepada *resource*, dan apabila diterapkan dapat dilakukan dengan cara melakukan beberapa tingkat keamanan yang harus dilalui.

Hubungan antara kontrol akses dengan pemilik dari objek tersebut akan sangat menentukan sehingga akan sangat sulit mempertahankan konsistensinya. Kontrol akses dapat dibentuk dalam satu unit subjek-objek, dan pengguna yang telah memiliki izin dapat mengizinkan pengguna lainnya untuk mengakses sumber daya tersebut. Kebijakan dari kontrol akses dapat diubah oleh pemiliknya sendiri, dan pemilik tersebut secara optional dapat melimpahkan kewenangan yang mereka miliki sehingga akan mengalami kesulitan untuk mengontrol informasi secara efisien (Na et al, 2000).

Penerapkan SSD dalam pendataan masyarakat miskin mengalami kendala, sewaktu menetapkan *role* bagi pengguna yang melakukan pendataan atau pengguna tersebut seorang RT di

gampong dan pengguna itu termasuk dalam kategori miskin, oleh sistem pengguna tersebut ditolak karena SSD hanya memperbolehkan satu *role* dimiliki oleh satu pengguna. Dengan menggunakan DSD dapat menyelesaikan permasalahan di atas, tetapi hal tersebut akan mengalami kendala yang diakibatkan rawan terjadinya manipulasi, misalnya pegawai negeri sipil (PNS) tidak termasuk dalam kategori miskin tetapi jika ada bantuan PNS tersebut akan dimasukkan sebagai penerima bantuan.

Hal ini disebabkan DSD dapat mengaktifkan banyak *role* walaupun dalam waktu yang berbeda sehingga rawan terjadinya manipulasi data yang disimpan ke dalam database. Dalam penelitian ini akan mengimplementasikan hambatan RBAC yaitu dengan DSD sesuai fungsi dan mekanismenya, sehingga pengguna RBAC tidak akan kehilangan wewenang yang dimilikinya.

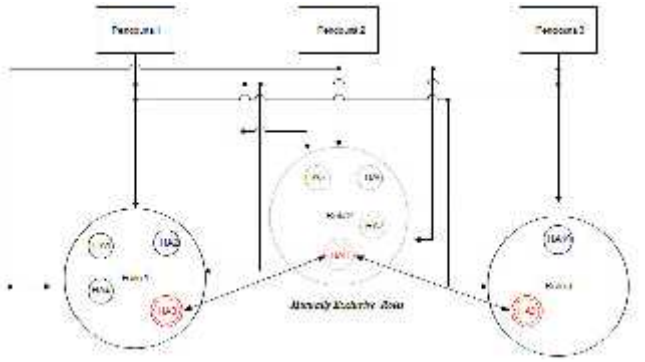
2. METODE PENELITIAN

DSD (*Dynamic Separation of Duty*) dengan menggunakan hambatan pada hak akses bukan hambatan pada *role*, serta mengimplementasikan studi kasus pendataan masyarakat miskin bukan suatu hal yang sangat mudah, pendataan membutuhkan banyak pengguna yang akan menanggani permasalahan tersebut. Penerapan DSD pada tingkat *role* di satu sisi menjadikan sistem lebih aman dari ancaman keamanan internal tetapi di sisi lain menciptakan berbagai permasalahan bagi pengguna RBAC termasuk pengguna akhir maupun administrator keamanan.

Model yang diajukan dan dijelaskan secara terperinci dengan mengikuti skenario yang digambarkan di bawah ini. Dalam ilustrasi gambar di bawah ini diproses secara lengkap sesuai yang dilakukan oleh *role*. Seperti dalam contoh melibatkan tiga pengguna dan tiga *role* dengan hak akses yang ganda. Misalkan kita memiliki tiga pengguna yaitu pengguna1, pengguna2 dan

pengguna3 serta mempunyai otoritas untuk mengaktifkan tiga *mutually exclusive roles* yaitu Role1, Role2 dan Role3. Setiap pengguna akan mengaktifkan salah satu dari tiga mutually *exclusive roles*, maka pengguna tersebut tidak dapat mengaktifkannya baik itu sebagian maupun keseluruhan role yang ada dalam *session* yang sama sesuai dengan definisi DSD standar.

Pada gambar di bawah ini pengguna1 ingin mengaktifkan Role1 dan pengguna2 mengaktifkan Role2 begitu juga pengguna3 mengaktifkan Role3. Tanda panah putus-putus memperlihatkan pengguna tidak dapat mengaktifkan *role* diakibatkan pelaksanaan DSD dari segi *mutually exclusive roles*. Akibat penerapan DSD pada tingkat *role*, pengguna RBAC ke hilangan ranah otoritas mereka.



Gambar 1 Tugas Pengguna dalam *Mutually Exclusive Roles*

3. HASIL DAN PEMBAHASAN

Identifikasi adalah pada tahap ini pengguna akan memberitahukan siapa dirinya. Otentikasi adalah tindakan untuk memverifikasi klaim identitas pengguna tersebut yaitu sesuatu yang mereka ketahui misalnya *password*, nomor induk karyawan atau sidik jari dan lain-lain. Proses otentikasi berfungsi sebagai

kesempatan pengguna untuk menerima dan melakukan proses pengaksesan *resource*. Pihak pengguna harus mampu untuk memberikan informasi yang dibutuhkan oleh pemberi layanan dan berhak untuk mendapatkan resourcenya. Sedangkan pihak yang memberi layanan harus mampu menjamin bahwa pihak yang tidak berhak tidak dapat mengakses *resource* ini.

Pengaksesan data yang dilindungi harus dibatasi untuk orang yang berwenang dalam mengakses data tersebut. Otorisasi akan menentukan sumber daya informasi yang diizinkan untuk diakses dan tindakan apa saja yang akan diizinkan dalam melakukan proses ke sistem tersebut yang dapat dilakukan oleh pengguna.

3.1 PENUGASAN HAK AKSES

Kegiatan hak akses dapat diperbaharui setiap saat setelah kegiatan membuat serta mengaktifkan perubahan dalam mengakses sumber daya yang diminta oleh pengguna. Sistem akan menyimpan riwayat hak akses yang telah diaktifkan oleh pengguna. Sistem melakukan verifikasi pengaktifan hak akses yang telah dilakukan oleh pengguna. Riwayat hak akses akan disimpan dalam jangka waktu yang telah ditentukan atau pada saat kegiatan tersebut dilakukan. Setelah kegiatan selesai dilakukan, riwayat terhapus dalam database sehingga sistem tidak menyimpan riwayat kegiatan tersebut selamanya.

Perhitungan rangkaian ranah otoritas yang tersedia dalam hak akses dalam mengimplementasikan DSD. Kedua hak akses dipisahkan seperti yang terlihat di bawah ini. Hak akses negatif memiliki tindakan negatif ditetapkan pada suatu objek yang menggantikan hak akses positif dan memiliki kegiatan positif sesuai dengan ketetapan pada objek yang sama sebagai hak akses negatif. Hak akses positif mempunyai hak akses negatif kemudian akan dibatalkan serta dihapus dari ranah pengguna RBAC.

Hak akses dalam kegiatan “baca”

HA1 = Baca (obj1)
HA2 = Tulis (obj2)
HA3 = Cetak (obj7)
HA4 = Baca (obj2)
HA5 = Baca (obj7)

Roles
R1 = {HA1, HA2, HA3}
R2 = {HA4}
R3 = {HA5}

Pengguna
P1 = {R1, R2, R3}

Hambatan dalam Hak akses
MEP = {HA4, HA5

MEP MER
HA4 ⟷ HA5 ⟶ Normalisasi R2 ⟷ Normalisasi R3

Setelah mengimplementasikan DSD dengan cara menentukan ranah kewenangan untuk pengguna P1= {S1, S2}

S1 = {R1, R2}
S2 = {R1, R3}

3.2 PEMBAHASAN

Role berorientasi pada *group*, sekumpulan transaksi dibuat berbentuk objek berupa program dan berhubungan dengan data. Seorang administrator dapat menambah dan menghapus transaksi ke dalam sebuah role atau menolak pengguna pada suatu *role*. Di bawah ini adalah *source code* untuk melakukan menyimpan *role*.

```
Private Sub Simpan()
    roles1 = TampilKodeCombo(" select * from roles where
    rolenama='"  & CboRole & " '" , Db)
    Db.ExecQuery (" insert into userrole(userID,roleID,status)
    values ('"  & CboUserID & " ','" & roles1 & " ','" & CboStatus & "
    ')" )
    CmdBatal_Click
End Sub

Private Sub CmdSimpan_Click()
     Simpan
    Db.ExecQuery (" insert into kunjungan(userid,awal,akhir,
    tgl_kunjung,keterangan) values ('" & UserID & " ','" & Waktu &
    " '," & _
    " '" & Time & " ','" & Format(Date, " yyyy/mm/dd" ) & "
    ','Menyimpan Data Role User'" & " )" )
End Sub
```

Hak dan izin diberikan pada role bukan pada pengguna. Pengguna memerlukan hak dan izin secara *virtual* dengan jalan memasukkan pengguna tersebut me njadi anggota dari role yang bersangkutan. *Role* berorientasi pada *group*, atau sekumpulan transaksi yang dibuat. Transakasi disini dapat merupakan obyek yang berupa program yang berhubungan dengan data. Seorang administrator dapat menambah dan menghapus transaksi ke dalam sebuah *role* atau bahkan menolak pengguna pada suatu *role*. Dengan mengelompokkan pengguna kedalam *role* maka ada memudahkan pada proses otorisasi dan kemampuan dalam melakukan pengamanan. Hal ini bertolak belakangan dengan *access list* model pada umumnya yang dilakukan dengan jalan mencari seluruh otorisasi yang ada kemudian mengalokasi hak dan izin untuk pengguna tersebut.

Metode yang cocok untuk digunakan dan dapat mengurangi komplesitas administrasi dan mengurangi biaya yang dibutuhkan dalam menjaga keamanan sebuah sistem. Pada dasarnya RBAC memberikan pengguna keanggotaan pada *role* berdasarkan

kompetensi dan tanggung jawan masing-masing pengguna. Pada RBAC pengguna tidak dapat melakukan operasi atas inisiatif sendiri melainkan berdasarkan role yang telah diperoleh dari administrator.

Pengaksesan data yang dilindungi harus dibatasi untuk orang yang berwenang dalam mengakses data tersebut. Otorisasi akan menentukan sumber daya informasi yang diizinkan untuk diakses dan tindakan apa saja yang akan diizinkan dalam melakukan proses ke sistem tersebut yang dapat dilakukan oleh pengguna. Otorisasi untuk mengakses informasi di dalam sistem di mulai dengan kebijakan administrasi yang dibuat oleh organisasi tersebut. Kebijakan akan menentukan data apa yang dapat diakses, oleh siapa dan dalam kondisi apa.

4. KESIMPULAN

Mutually exclusive permissions pada *prototype* yang telah peneliti rancang adalah sebagai berikut:

1. Penggunaan DSD dengan *mutually exclusive roles* rawan terjadinya manipulasi diakibatkan tidak membatasi hak akses pengguna berbeda dengan *mutually exclusive permissions* dapat membatasi hak akses pengguna
2. Pada DSD standar dua buah *role* tidak dapat diaktifkan oleh pengguna yang sama dalam *session* yang sama tetapi dapat diaktifkan oleh pengguna yang sama pada *session* berbeda

DAFTAR PUSTAKA

Habib, M. A. 2011, *Role inheritance with object-based DSD*. Int. J. Internet Technology and Secured Transactions, Vol. 3, No. 2, pp.149–160.

Na. S. Y & Cheon, S. H . 2000, *Role delegation in role-based access control*. *In Proceedings of the Fifth ACM Workshop on Role-Based Access Control* (RBAC'00), pages 39–44