مدل مدیریتی کنترل دسترسی نقش مبنا

Masood Niazi Torshiz
www.mniazi.ir

طرح یک مشکل و ارائه یک راه‌حل

- اگر سیستمی با هزاران کاربر و صدها نقش و مجوز در نظر بگیریم، مدیریت نقش‌ها و انتساب کاربران به نقش‌ها و مجوزها به نقش‌ها و همچنین ساخت سلسله مراتب از نقش‌ها بسیار پیچیده و مشکل می‌گردد و نمی‌توان آن را توسط یک مدیر در سیستم انجام داد .
- راهکار: مدیریت غیر متمرکز.
  - در نظر گرفتن نقش‌های مدیریتی و حوزه‌های مدیریتی برای هر نقش
  - همچنین ایجاد سلسله مراتبی از آنها
  - واگذاری مدیریت به افراد مختلف در حوزه‌های گوناگون سیستم
  - هر مدیر مسؤولیت مدیریت در حوزه خود را بر عهده داشته باشد

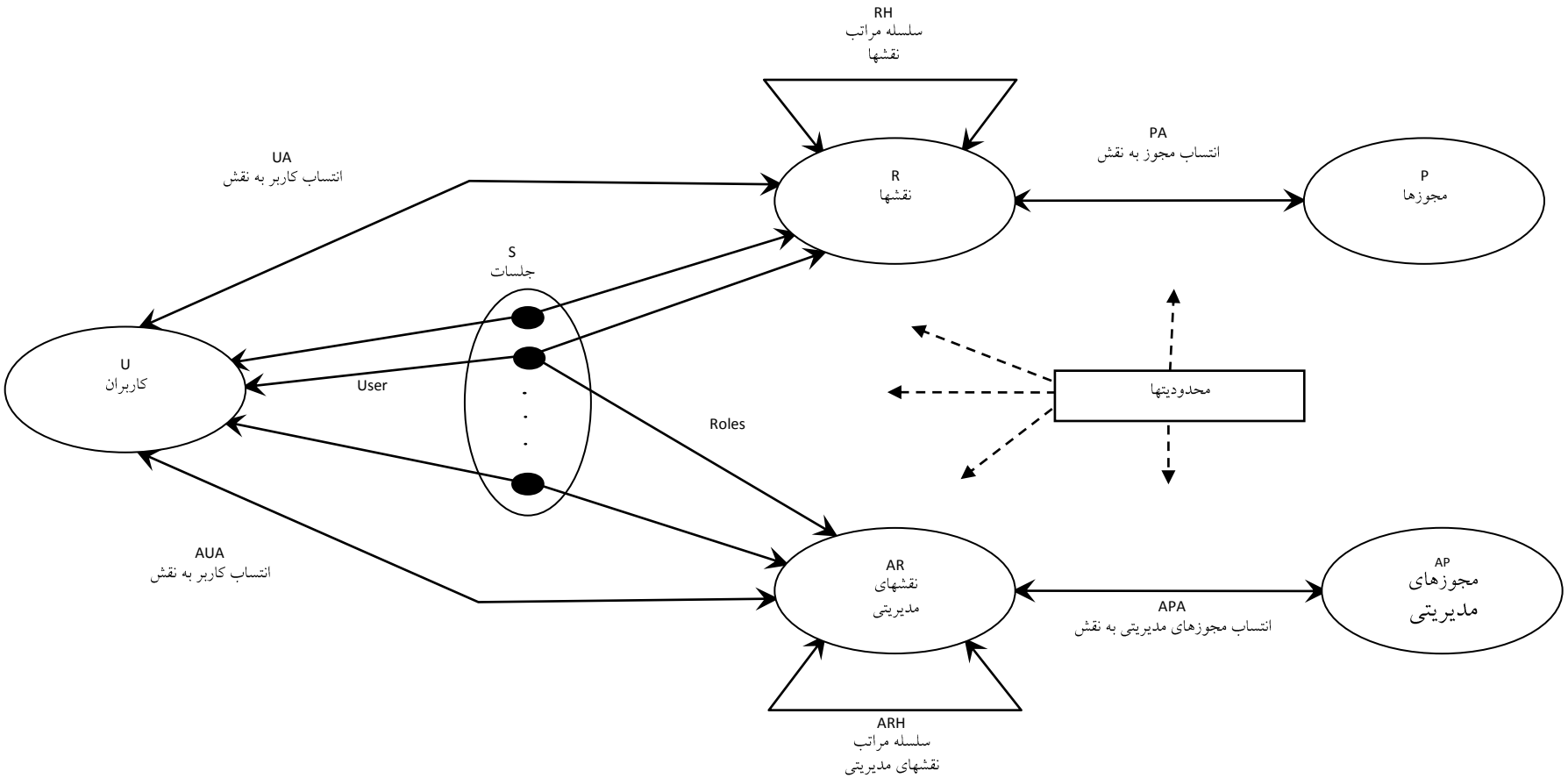
مدیریت در کنترل دسترسی نقش- مبنا

- در سیستم بزرگ که تعداد نقش‌ها به صدها و هزاران افزایش می‌یابد، مدیریت این نقش ها و روابط میان آنها یک کار سخت که به صورت مرکزی انجام می‌شود و به گروه کوچکی از مدیران امنیتی محول می‌گردد.
- نکته اصلی RBAC این که مدیریت را ساده می‌کند >> می‌توان از خود آن در مدیریت خودش استفاده نمود.
- نقش‌های مدیر یا AR و اختیارات مدیر یا AP را از نقش های معمولی یا R و اختیارات معمولی یا P جدا می‌کنیم .
- اختیارات تنها به نقش ها نسبت داده می‌شوند و اختیارات مدیریتی تنها به نقش‌های مدیریتی نسبت داده می‌شوند .

مدل مدیریتیRBAC

- انواع مدل های بحث شده، برای مدیر نیز مطرح است. البته معمولا مدل مدیر ساده تر از خود مدل RBAC است. بنابراین می توان از RBAC0 به جای RBAC3 استفاده نمود.
- چگونه مدل سلسله مراتبی مدل مدیر مدیریت می شود؟
  - به طور تئوریک، سطح دوم از سلسله مراتب می تواند برای مدیریت سطح اول مورد استفاده قرار گیرد. ولی برای مدل ضروری نمی باشد.
  - مدیریت سلسله مراتب مدیر می‌تواند توسط یک نفر رئیس سیستم مدیریت انجام شود. مجوزهای مدیر در RBAC توانایی تغییر نسبت نقش به کاربران و نیز تغییر دادن نسبت اختیارات به نقش ها و روابط موجود در سلسله مراتب نقش ها را به وجود آورد.

اجزاء مدل مدیریتیRBAC



اجزاء مدل مدیریتی RBAC

- این مدل در سال ۱۹۹۷ توسط Sandhu ارائه گردید. ایده اصلی آن استفاده از خود مدل RBAC برای مدیریت آن بود. این مدل شامل سه مدل اصلی به شرح زیر می باشد :

  - مدل URA یا مدل انتساب کاربران به نقش
  - مدل PRA یا مدل انتساب مجوزها به نقش
  - مدل RRA یا مدل انتساب نقش به نقش

مدل URA یا مدل انتساب کاربران به نقش

- این مدل دارای ۲ مؤلفه اصلی است:
  - انتساب کاربران به نقش ها یا Grant ← Can-assign
  - باز پس گیری عضویت آنها در نقش ها یا Revoke ← Can-revoke

- رابطه Can-assign بیان می‌کند که چه افرادی با چه پیش‌شرط‌هایی می‌توانند در چه حوزه‌ای کار اعطاء را انجام دهند. افراد را با نقش‌های مدیریتی که دارا هستند معین می کند.
  - نقشی را بیان می کند که افراد برای اعمال کارهای مدیریتی در یک حوزه خاص باید دارا باشند.
- رابطه Can-revoke بیان می‌کند که چه افرادی در چه حوزه‌هایی می‌توانند عمل بازپس‌گیری را انجام دهند.

این توابع برای عمل انتساب و بازپس‌گیری نقش ها به کاربران به کار می‌رود و بایستی در هر عمل، امکان انجام آنرا توسط آنها چک کرد.

مدل URA97

can-assign(x, y, z) // x: administrative role,
y: prerequisite condition, z: role range

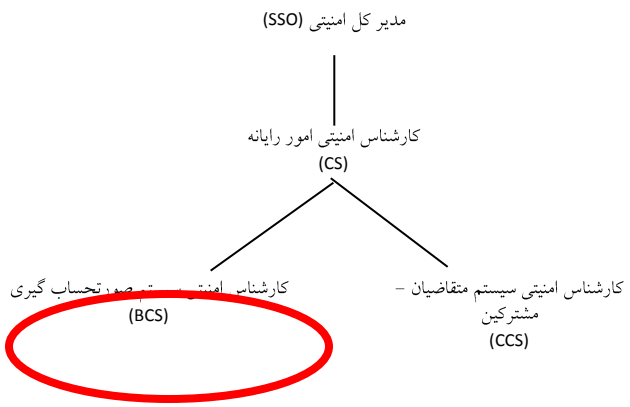
Admin. Role	Prereq. Condition	Role Range
PSO1	ED	[E1, E1]
PSO1	$E1 \wedge \overline{QE1}$	[PE1, PE1]
PSO1	$E1 \wedge \overline{PE1}$	[QE1, QE1]
PSO2	ED	[E2, E2]
PSO2	$E2 \wedge \overline{QE2}$	[PE2, PE2]
PSO2	$E2 \wedge \overline{PE2}$	[QE2, QE2]
DSO	$ED \wedge \overline{PL2}$	[PL1, PL1]
DSO	$ED \wedge \overline{PL1}$	[PL2, PL2]
DSO	ED	(ED, DIR)
SSO	E	[ED, ED]
SSO	ED	(ED, DIR]

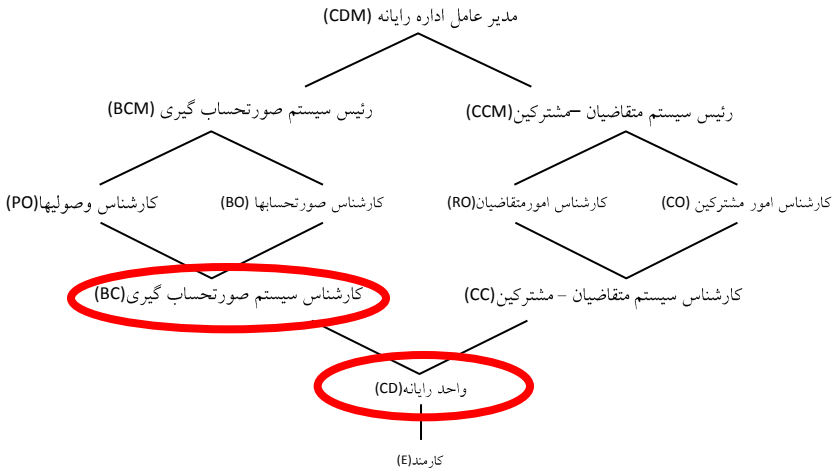
- تابع Can_assign دارای سه پارامتر ورودی است: X که نقش مدیریتی فردی که می خواهد عمل انتساب را انجام دهد را مشخص می کند. Y نقش پیش شرط فردی است که می خواهیم به او نقش را انتساب دهیم و Z که دامنه نقش های قابل انتساب را معین می کند.
- یعنی فرد دارای نقش مدیریتی X به یک کاربر که فعلا دارای نقش Y است می تواند هر نقشی در دامنه Z عطا کند.

مدل URA97 - رابطه Can_Assign

نقش مدیریتی	پیش‌شرط	دامنه نقش‌ها
BCS	CD	[BC,BC]
BCS	BC ∧ ¬BO	[PO,]
BCS	BC ∧ ¬PO	[BO,BO]
CCS	CD	[CC,CC]
CCS	CC ∧ ¬CO	[RO,RO]
CCS	CC ∧ ¬RO	[CO,CO]
CS	CD ∧ ¬CCM	[BCM,BCM]
CS	CD ∧ ¬BCM	[CCM,CCM]
CS	CD	(CD,CDM)
SSO	E	[CD,CD]
SSO	CD	(CD,CDM]

- اولین سطر این جدول بیان می‌کند که کاربری با نقش "کارشناس امنیتی سیستم صورتحساب گیری" (BCS) و در نتیجه "کارشناس امنیتی امور رایانه" (CS) و "مدیرکل امنیتی" (SSO) می‌تواند به کاربری که هم‌اکنون دارای نقش عادی "واحد رایانه" (CD) است, نقش "کارشناس سیستم صورتحساب گیری" (BC) را عطا کند.





مدل URA97

- تابع Can_Revoke دارای دو پارامتر ورودی است: X که نقش مدیریتی فردی که می خواهد عمل بازپس گیری نقش را انجام دهد را مشخص می کند و Z دامنه نقش هایی را که می تواند بازپس گیرد تعیین می کند.
- هیچ پیش شرطی برای این تابع تعریف نمی گردد.

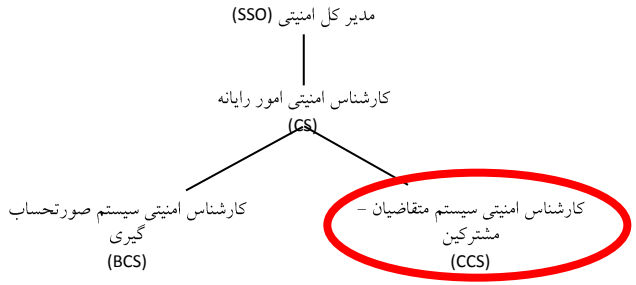
can-revoke(x, z) // x: administrative role,
z: role range

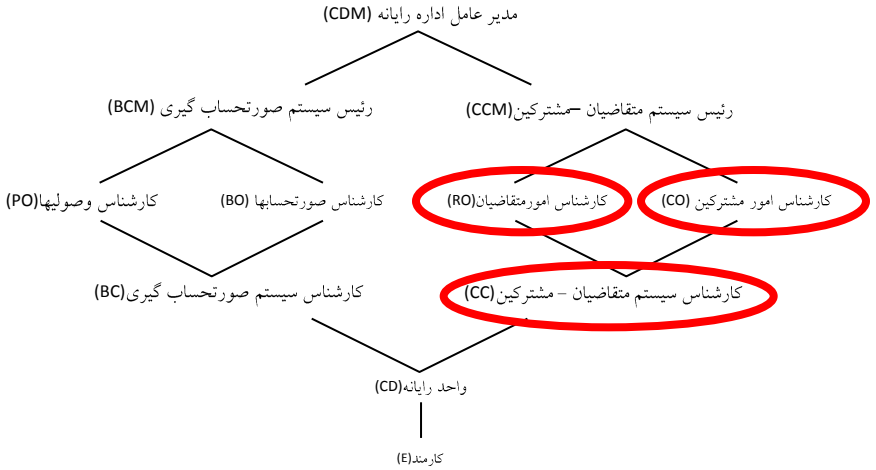
Admin. Role	Role Range
PSO1	[E1, PL1)
PSO2	[E2, PL2)
DSO	(ED, DIR)
SSO	[ED, DIR]

مدل URA97 - رابطه Can_Revoke

نقش مدیریتی	دامنه نقش‌ها
BCS	[BC,BCM)
CCS	[CC,CCM)
CS	(CD,CDM)
SSO	[CD,CDM]

- سطر دوم این جدول بیان می‌کند که "کارشناس امنیتی بخش متقاضیان و مشترکین" می‌تواند از تمام کاربران سیستم, نقش "کارشناس سیستم متقاضیان و مشترکین" (CC) , "کارشناس امور متقاضیان" (RO) و "کارشناس امور مشترکین" (CO) را بازپس‌گیرد.





مدل PRA97 یا مدل انتساب مجوزها به نقش

- این مدل نیز مشابه مدل قبلی دارای ۲ مؤلفه اصلی است:
  - مدل Grant برای اعطاء مجوزها به نقش ها ← Can-assignp
  - مدل Revoke برای بازپس گیری مجوز از نقش ها ← Can-revokep

- رابطه اول وظیفه تعیین افراد و شرط ها برای انجام عمل انتساب در یک حوزه خاص و رابطه دوم وظیفه تعیین افراد برای انجام عمل بازپس گیری در یک حوزه خاص را بر عهده دارد.

مدل PRA97

- تابع Can_Assignp دارای سه پارامتر ورودی است. X که نقش مدیریتی مجری عمل انتساب را مشخص می کند. Y نقش ای است که می توان مجوزهای آن را برای عمل انتساب انتخاب کرد. Z ، حوزه نقش هایی است که می توان مجوز انتخاب شده را به آن نسبت داد.

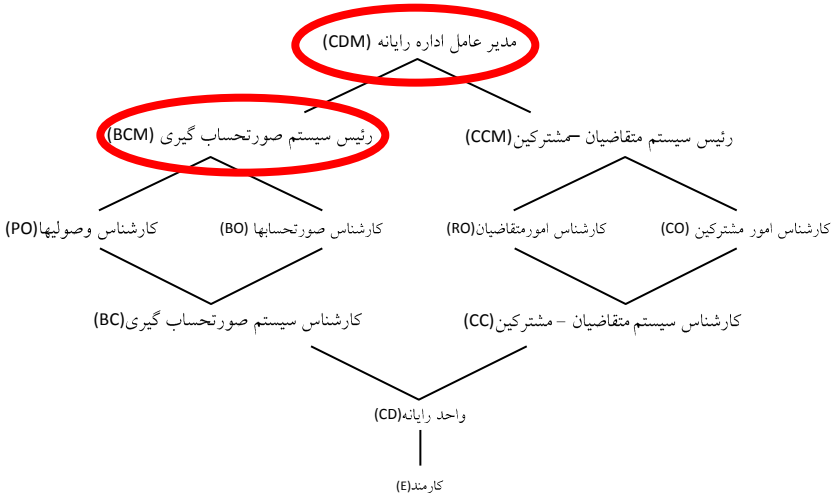
can-assignp(x, y, z) // x: administrative role,
y: prerequisite condition, z: role range

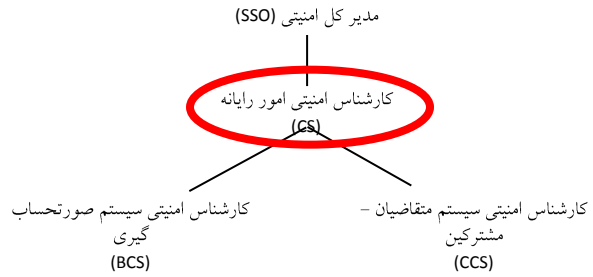
Admin. Role	Prereq. Condition	Role Range
DSO	DIR	[PL1, PL1]
DSO	DIR	[PL2, PL2]
PSO1	PL1 ∧ $\overline{QE1}$	[PE1, PE1]
PSO1	PL1 ∧ $\overline{PE1}$	[QE1, QE1]
PSO2	PL2 ∧ $\overline{QE2}$	[PE2, PE2]
PSO2	PL2 ∧ $\overline{PE2}$	[QE2, QE2]

مدل PRA97 - رابطه Can_AssignP

دامنه نقش‌ها	پیش‌شرط	نقش مدیریتی
[BCM,BCM]	CDM	CS
[CCM,CCM]	CDM	CS
[PO,]	BCM ^ ¬BO	BCS
[BO,BO]	BCM ^ ¬PO	BCS
[RO,RO]	CCM ^ ¬CO	CCS
[CO,CO]	CCM ^ ¬RO	CCS

- سطر اول این جدول بیان می‌دارد که کاربری با نقش "کارشناس امنیتی امور رایانه" (CS) یا "مدیرکل امنیتی" (SSO) می‌تواند تمام مجوزهای صریح و ضمنی نقش "مدیرعامل اداره رایانه" (CDM) یعنی در واقع تمام مجوزهای موجود در سیستم را به نقش "ریاست سیستم صورتحساب‌گیری" (BCM) عطا کند.





مدل PRA97

can-revokep(x, z) // x: administrative role, z: role range

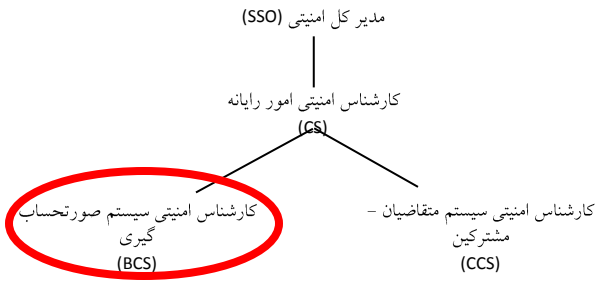
- تابع Can_Revokep دارای دو پارامتر ورودی است که x نقش مدیریتی مجری عمل بازپس گیری و z حوزه نقش هایی است که می توان در آن حوزه عمل بازپس گیری مجوز ها را انجام داد.
- y در PRA حوزه انتخاب مجوزها را در تابع Can_Assignp مشخص می کند. در حالی که در URA پیش شرط برای اخذ نقش بود. بنابراین در PRA می توان yرا به عنوان Permission pool یا حوزه ای برای انتخاب مجوز ها جهت انتساب دانست.

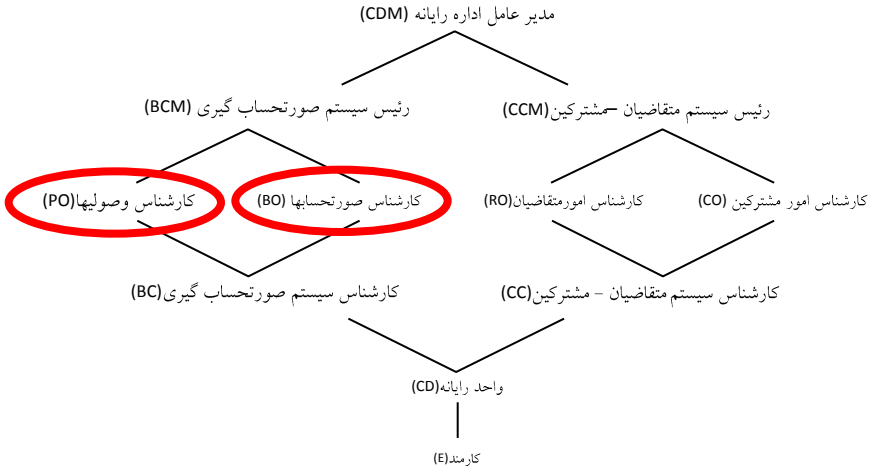
Admin. Role	Role Range
PSO1	(E1, PL1)
PSO2	(E2, PL2)
DSO	(ED, DIR)
SSO	[ED, DIR]

مدل PRA97 - رابطه Can_RevokeP

نقش مدیریتی	دامنه نقش‌ها
BCS	(BC,BCM)
CCS	(CC,CCM)
CS	(CD,CDM)
SSO	[CD,CDM]

- سطر اول این جدول بیان می‌کند که "کارشناس امنیتی سیستم صورتحساب گیری" (BCS) و طبیعتا "کارشناس امنیتی امور رایانه" (CS) و "مدیرکل امنیتی" (SSO) می‌توانند هر مجوزی را از "کارشناس صورتحساب ها" (BO) و "کارشناس وصولی ها" (PO) بازپس گیرند.





مدل RRA یا مدل انتساب نقش به نقش

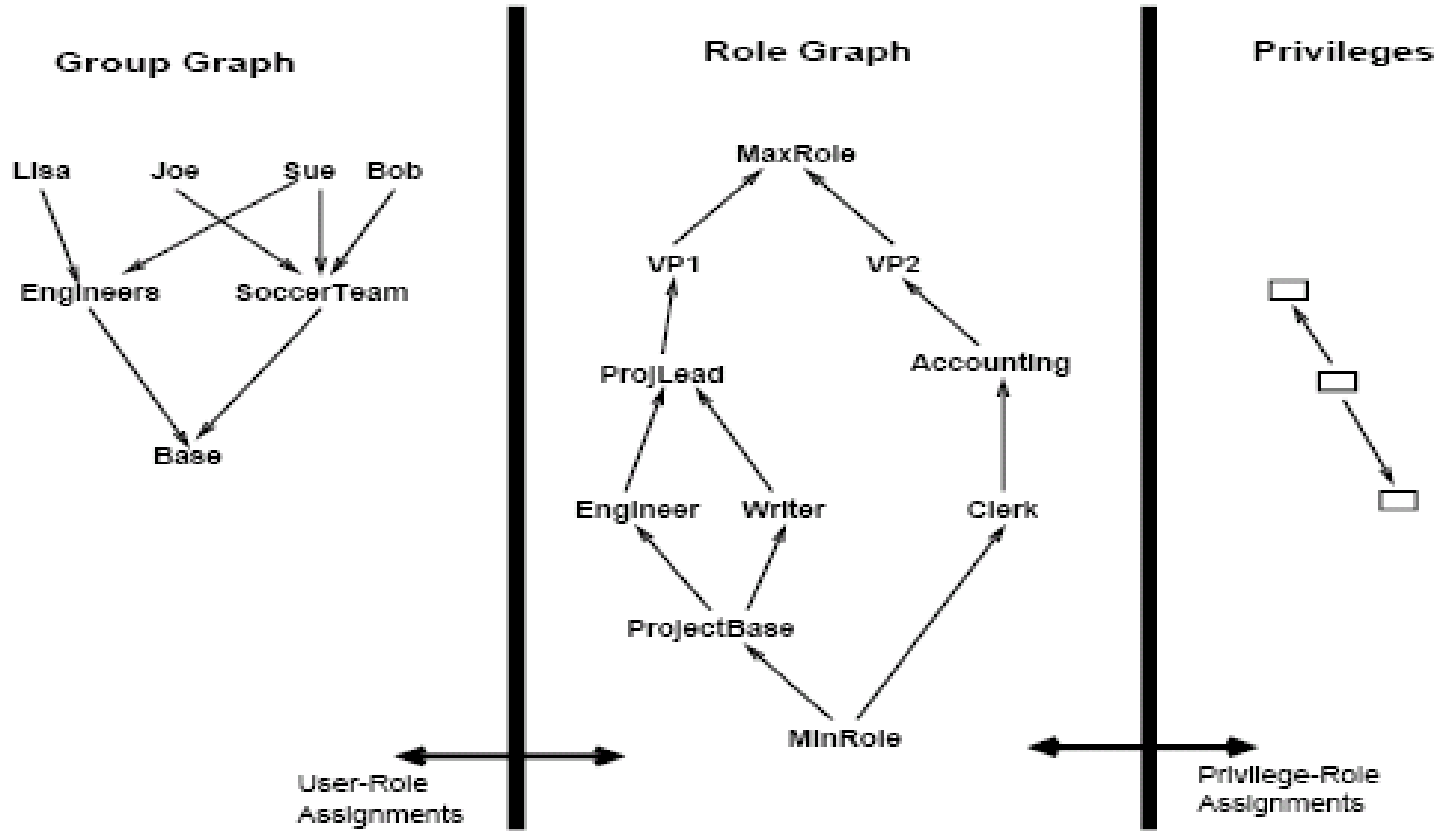
- ایجاد یک سلسله مراتب از نقش ها
- فراهم آوردن بستری برای ساخت مدل RBAC1

- وقتی نقشی، بالاتر از یک نقش دیگر قرار می گیرد، تمام مجوزهای نقش قبلی را به ارث می برد.
- این مدل سلسله مراتبی با توجه به ساختار سلسله مراتبی نقش های سازمانی می تواند شکل بگیرد و به هر چه بهتر مدل کردن نقش ها و نقش های مدیریتی موجود سازمان در سیستم کمک کند.

مدل Role Graph

- از دید دیگر مدل کنترل دسترسی نقش مبنا را مبتنی بر سه گراف در سه حوزه مختلف بررسی می کنند:
  - گراف اختیارات یا مجوز ها :
    - این گراف بیانگر سلسله مراتب حاکم برانواع مجوزهای مختلف است. ممکن است داشتن یک مجوز ، داشتن یک مجوز دیگر را ایجاب کند .
  - گراف گروه ها یا کاربران :
    - در این گراف کاربران یا گروهای کاربری و سلسله مراتب آنها نمایش داده می شود.
  - گراف نقش ها یا Role Graph :
    - در این گراف نقش های موجود سیستم ، گره های گراف را تشکیل می دهند و خط بین آنها ارتباط **شامل شدن** را معین می کند.

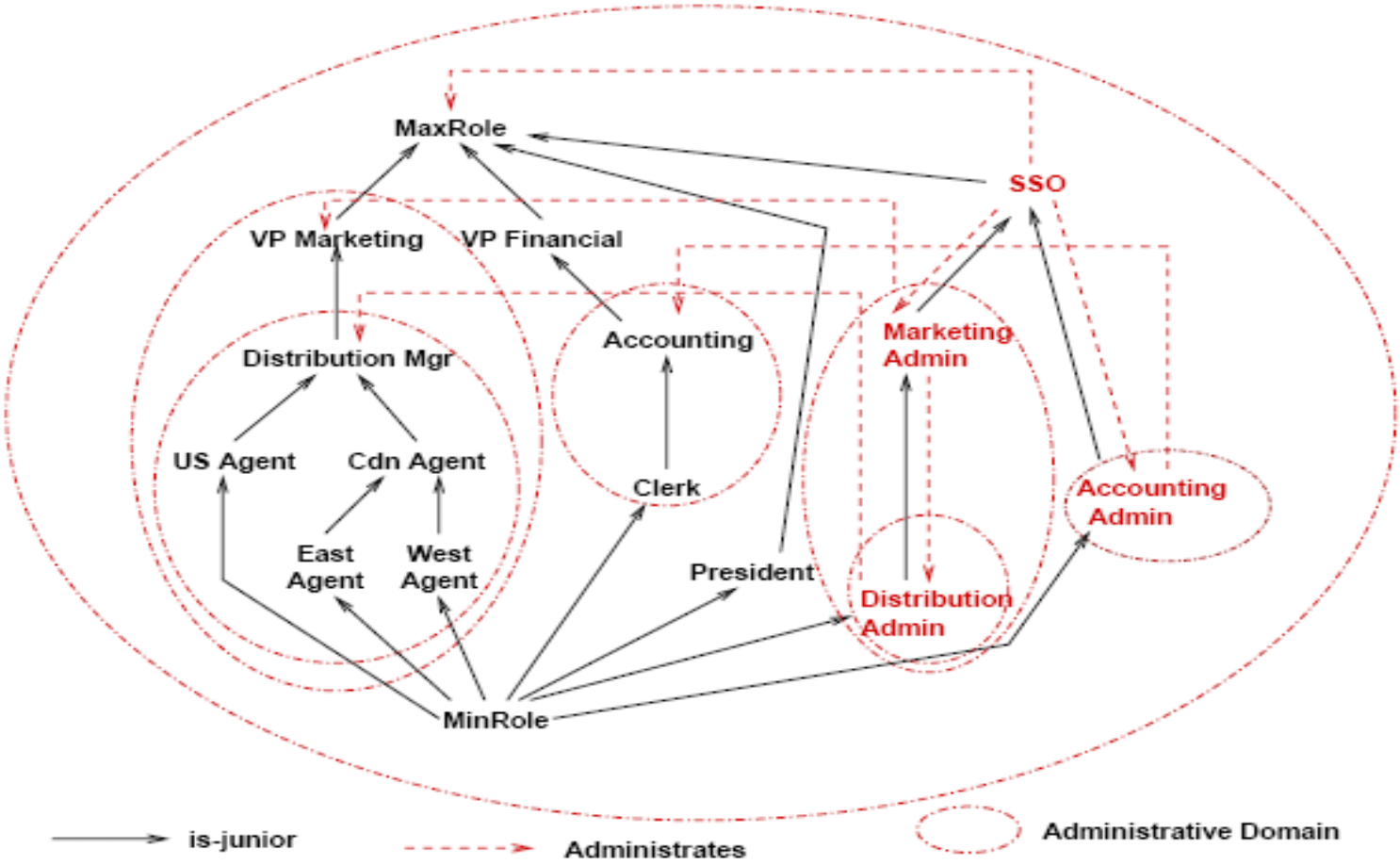
اجزاء مدل Role Graph



مدیریت غیرمتمرکز در مدل Role Graph

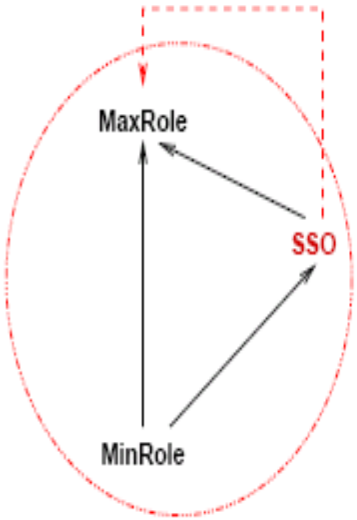
- مطابق همین دید به مدل کنترل دسترسی ، گراف نقش های مدیریتی نیز قابل ترسیم است. این گراف شامل نقش های عادی و نقش های مدیریتی است و دو رابطه در آن تعریف می گردد:
  - رابطه اول، Is-Junior است که رابطه ای بین نقش های عادی و یا بین نقش های مدیریتی است. این رابطه نشان دهنده شامل بودن یک نقش بر نقش دیگر است.
  - رابطه دوم، رابطه Administrates است که با خط های خط چین در شکل نشان داده شده است.
- این گراف با دو گره به نامهای MinRole و MaxRole و همچنین SSO که وظیفه مدیریت کل سیستم را بر عهده دارد ، در نظر گرفته می شود.

حوزه های مدیریتی در Role Graph

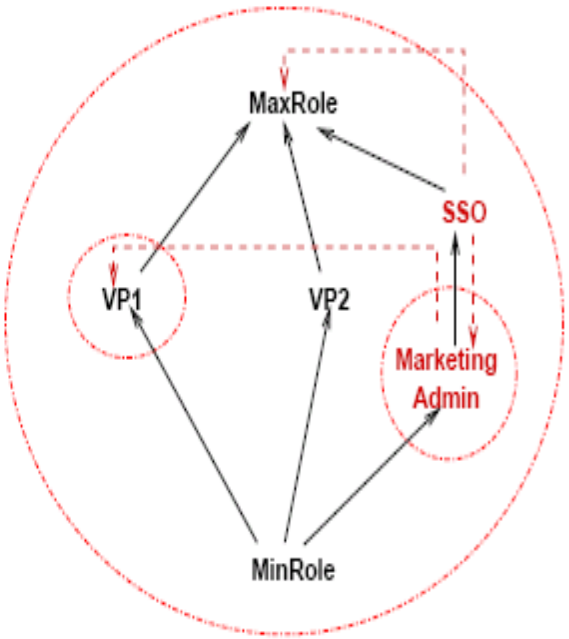


ساخت گراف نقشهای مدیریتی

- مطابق شکل a در ابتدا سه نقش و یک حوزه مدیریتی کلی با مدیریت SSO وجود دارد.
- مطابق شکل b بخشها به تدریج اضافه می شوند و گراف بزرگتر می شود.



a. Starting Role Graph



b. Intermediate Role Graph

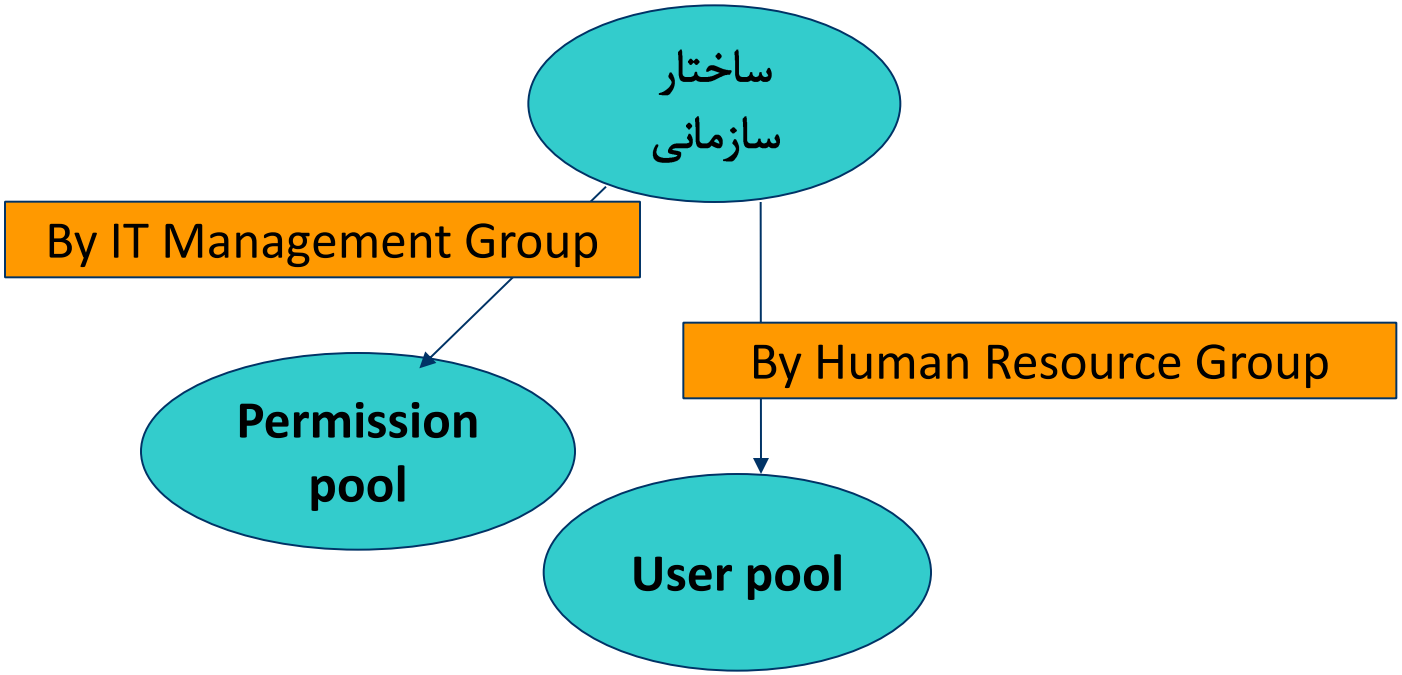
مدل مدیریتی RBAC توسعه یافته

- سعی شده است مشکلات مطرح شده، در مدل توسعه یافته یعنی ARBAC02 حل گردند. در این مدل مفاهیم User Pool و Permission Pool مطرح می شود و سعی می گردد تا با حل تداخل های غیرلازم موجود ، مشکلات مطرح شده کنار گذاشته شود.
- برای غلبه بر مشکلات مطرح شده در مدل قبل، دو استراتژی در این مدل اتخاذ شده است:
  - اول، از ساختار سازمانی به عنوان User pool و Permission pool استفاده می شود به جای اینکه از پیش شرط هایی در سلسله مراتب نقش ها استفاده کرد .
  - دوم، توسط این ساختار سازمانی یک روند پائین به بالا برای انتساب مجوز ها به نقش ها مطرح میشود.

ساختار سازمانی

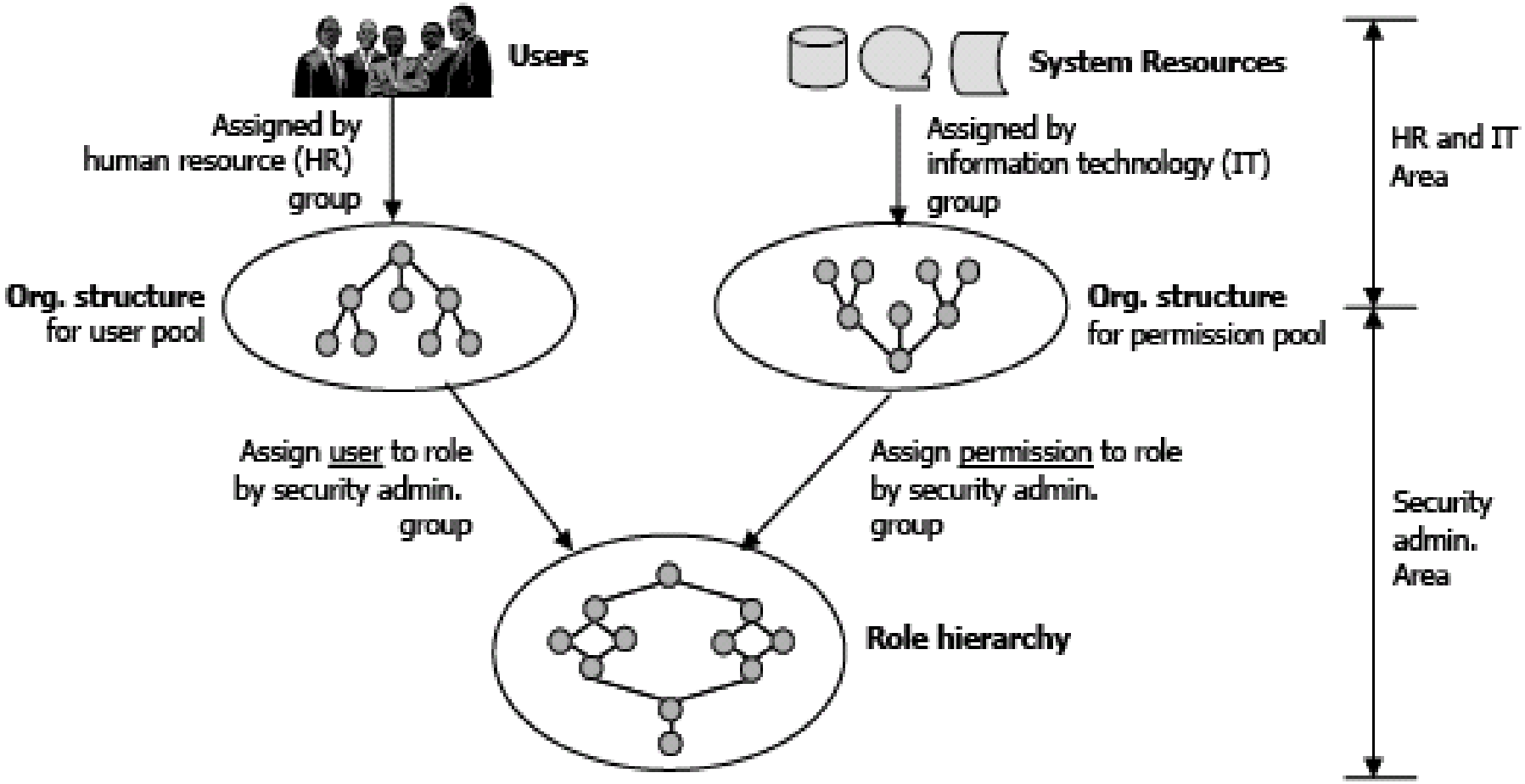
- برای توسعه سیستم های اطلاعاتی، ***"سازمان"*** یک مفهوم خوب برای تحلیل فعالیت های موجود در هر دامنه است.
- ساختار سازمانی یک ساختار درختی با ویژگی سلسله مراتبی است. این ساختار از عناصر سازمانی تشکیل می شود که افراد متعلق به هر یک دارای یک هدف مشترک در سازمان هستند و یک سری فعالیتهای خاص برای رسیدن به آنها انجام می دهند.
- کارهای انجام یافته با داده های مورد دسترسی ارتباط مستقیم دارد. پس فعالیت ها و کارهای یک بخش با مجوز های آن ارتباط دارد.
- پس می توان واحد سازمانی را به عنوان یک ***گروه از کاربران*** و ***مجوز ها*** برای رسیدن به هدف خاص تعریف کرد .

User & Permission Pool ساخت



- حال مدیر های امنیتی، کاربران و مجوزهای موجود در هر واحد سازمانی را به نقش ها نسبت می دهند.

ساختار مدل مدیریتی RBAC توسعه یافته

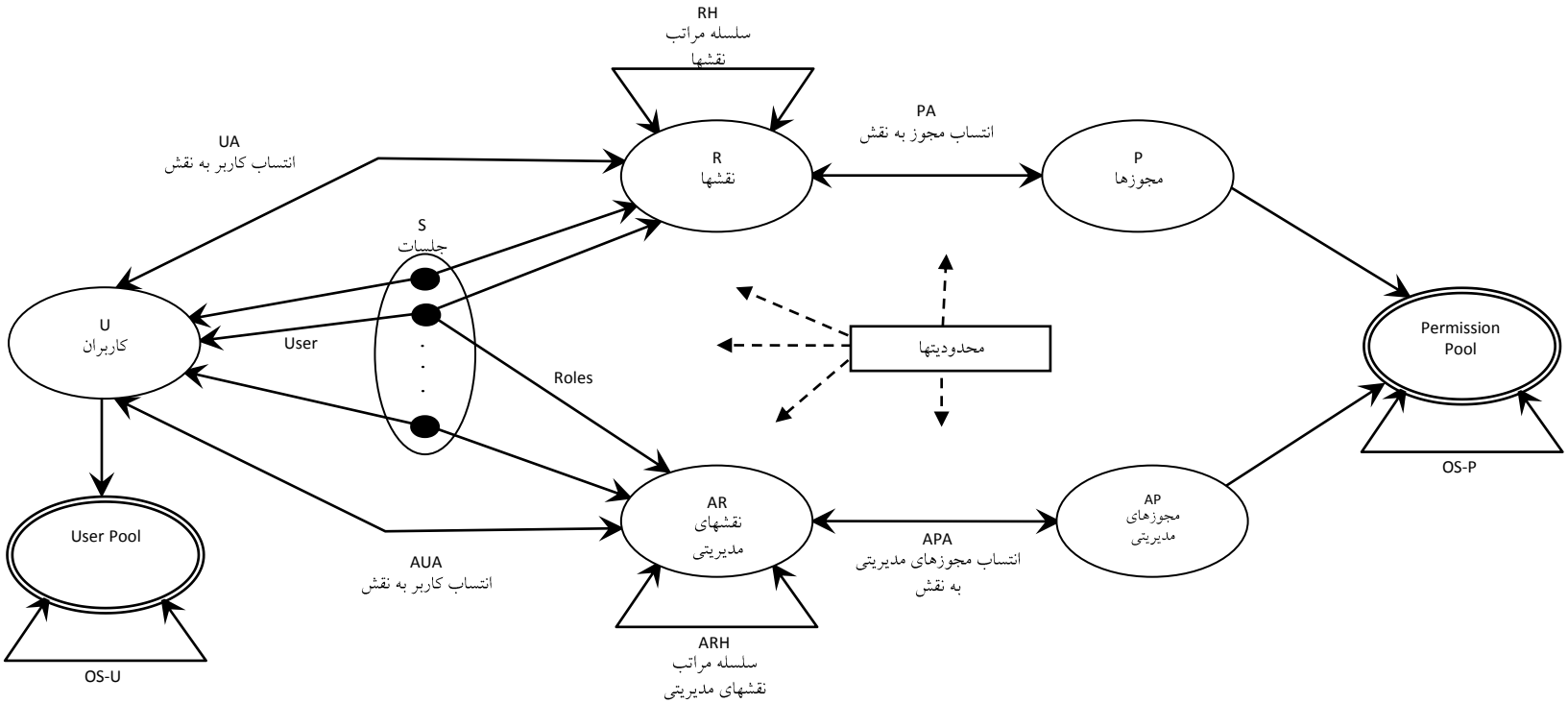


اصلاح مدل با اعمال مفهوم ساختار سازمانی

- توابع Can_Assign و Can_Assignp همان توصیف موجود در ARBAC97 را دارا هستند و فقط پیش شرط ها در آن مجددا تعریف شده است:
  - پیش شرط ها در URA یک عبارت با ترکیب عملگرهای And و Or روی نقش های عادی و یا واحد های سازمانی در ساختار سازمانی تهیه شده توسط گروه HR یعنی User Pool است.
  - پیش شرط ها در PRA یک عبارت منطقی از عملگر های And و Or روی عبارات x و x~ است که x یک نقش عادی یا یک واحد سازمانی در ساختار سازمانی تهیه شده توسط گروه فا یا Permission Poolاست .

اجزاء مدل ARBAC02

- ساخت Permission Pool با یک ساختار سازمانی به نام OS-P
- ساخت User Pool با یک ساختار سازمانی به نام OS-U



استفاده از RBAC برای اعمال DAC و MAC

- مکانیزم RBAC به اندازه ای کلی است که بتواند MAC و DAC را شبیه سازی کند.
- یک خصوصیت مهم اینکه خط مشی در طول چرخه عمر سیستم می تواند تغییر کند.
  - MAC : جریان یک طرفه اطلاعات
  - DAC : Owner Based Administration
- تعریف یک سری قوانین و محدودیت ها برای شبیه سازی MAC
- تعریف یک سری عملیات به ازای هر رخداد برای شبیه سازی DAC

مراجع

- *[1] S. Oh and R. Sandhu, "A model for role administration using organization structure", ACM SACMAT, 155-162, 2002.*
- *[2] S. Osborn, "Information flow analysis of an RBAC system", ACM SACMAT, 163-168, 2002.*
- *[3] Chandramouli Ramaswamy and Ravi Sandhu "Role-Based Access Control Features in Commercial Database Management Systems" , 21st National Information Systems Security , Jun 2005*
- *[4] Bertino, E.; Sandhu, R. "Database security - concepts, approaches, and challenges" , Dependable and Secure Computing, IEEE Transactions, March 2005*
- *[5] Ravi Sandhu and Venkata Bhamidipati "An Oracle Implementation of the PRA97 Model for Permission-Role Assignment" , ACM Workshop on Role-Based Access FairFax VA , 1998*
- *[6] He Wang and Sylvia L. Osborn "An Administrative Model for Role Graph Model" , Natural Sciences and Engineering Research Council of Canada.*
- *[7] Ravi s.Sandhu, Edward J.Coyne and Charles E.Youman, " Role-Based Access Cotrol Models" , IEEE, 38-47, February 1996*
- *[8] Ravi Sandhu, Venkata Bhamidipati, Edward Coyne, Sirinivas Ganta, and Charles Youman, "The ARBAC97 model for role-based administration of roles: Preliminary description and outline", In Preceeding of 2nd ACM Workshop on Role-Based Access Control, Fairfax, VA, November 6-7 1997. ACM.*
- *[9] He Wang and Sylvia L. Osborn "An Administrative Model for Role Graphs", In Data and Applications Security XVII, pages 39–44, Kluwer, 2003.*

مشخصات RBAC در DBMS های تجاری

- Oracle Enterprise Server version 8.0
- Informix Online Dynamic Server Version 7.2
- Sybase Adaptive Server release 11.5
- از سه جنبه مورد بررسی قرار خواهند گرفت:
  - اعطای نقش به کاربر
  - پشتیبانی ارتباطات و قیود در نقش
  - امتیازات قابل اعطا

Oracleو اعطای نقش به کاربر

- اوراکل ارتباط چند به چند بین کاربر و نقش را پشتیبانی می کند.
- PUBLICدر جمله GRANT
- ADMIN OPTION
- SET ROLE
- اگر نقش دارای رمز عبور باشد، رمز عبور با عبارت IDENTIFIED BY مشخص و فعال میشود.
- در اوراکل می توان بیش از یک نقش را در SET ROLE مشخص کرد.
- اوراکل دوگونه دیگر از جمله SET ROLE را دارد که به آن انعطاف پذیری بیشتری در فعالیت ها می دهد:
  - All & Except
  - None

Oracleو پشتیبانی *ارتباط و قیود در نقش*

- در اوراکل امکان دادن نقش به یک نقش در نتیجه ایجاد ساختار سلسله مراتبی نقش را دارد. گر چه نمی توان قیود اضافی یا ارتباطات را بین نقش ها در declaration تعریف کرد:
  - بنابراین اوراکل جداسازی وظایف یا SoD را پشتیبانی نمی کند .
- تعیین محدودیت در تعداد نقش ها برای اعضا ممکن نیست.
- امکان تعریف قیود فقط تا حدی وجود دارد.

Oracleو امتیازات قابل اعطا

- ***امتیازات سیستمی*** حقوقی هستند که با فرمانهایی نظیر CREATE SESSIONو CREATE TABLE و غیره اجرا می شوند .
- ***امتیازات شی ای*** به کاربران اجازه می دهد که یک عمل خاص را روی یک جدول خاص view یا دنباله اجرا کنند .
- هر دو شاخه امتیازات می توانند به نقش ها داده شوند. امتیازات سیستمی تنها می توانند توسط DBA یا یک کاربری که این امتیاز را با OPTION ADMIN دارد منتقل شوند. امتیازات شی ای تنها می توانند توسط صاحب شی یا کاربری که این امتیاز را با GRANT OPTION دارد منتقل شود .

مقایسه خصیصه ها در DBMS ها

مورد	خصیصه	Informix	Sybase	Oracle
۱	امکان دادن نقش به دیگر کاربران توسط**grantee**	√	-	√
۲	داشتن چند نقش فعال برای یک کاربر در یک نشست	-	√	√
۳	مشخص کردن نقش فعال بطور پیش‌فرض برای کاربر	-	√	√
۴	ایجاد ساختار سلسله مراتبی نقش	√	√	√
۵	جدا کردن استاتیک وظایف و قیود روی نقش ها	-	√	-
۶	جدا کردن دینامیک وظایف و قیود روی نقش ها	√	√	-
۷	مشخص کردن حداکثر و حداقل کاردینالیتی اعضای نقش	-	-	-
۸	دادن امتیازسیستمی **DBMS** به یک**Role**	-	√	√
۹	دادن امتیازشی‌ای **DBMS** به یک**Role**	√	√	√