Çoklu Etmen Sistemlerinde Rol Tabanlı Erişim Denetimi için Bir Yaklaşım Önerisi

Fatih TEKBACAK, Tuğkan TUĞLULAR, Oğuz DİKENELLİ

***Özet*—Bu çalışmada, şu anda yaygın olarak kullanılan sözdizimsel rol tabanlı erişim denetimi (RBAC) yaklaşımları ile beraber çalışabilecek ontoloji güdümlü bir rol hiyerarşisi ve katmanlı bir erişim denetimi mimarisini çoklu etmen sistemlerinin rol tabanlı etmenlerinde uygulayabilmek amaçlanmaktadır. Bu doğrultuda XACML (eXtensible Access Control Markup Language) modeline dayalı bir şekilde OWL anlamsal veb dili ve SWRL kural dili kullanarak ontoloji güdümlü erişim denetim modelinin geliştirilmesi düşünülmektedir.**

***Abstract*—In this study, our goal is to develop an ontology based role hierarchy that could collaboratively process with syntactic role based access control (RBAC) approaches and apply a multi-tier access control architecture on role based agents of multi agent environments. In this way, the focus is on the development of an access control model based on XACML (eXtensible Access Control Markup Language) by using OWL semantic web language and SWRL rule language.**

***Anahtar Kelimeler*—Ontoloji, OWL, politika, RBAC, XACML**

I. GİRİŞ

SON dönemlerde gelişen yüksek performanslı bilgisayarla hesaplama ve ağ teknolojileri, geniş ölçekli dağıtık uygulamaların fazlalaşmasına sebebiyet vermiştir. Bu uygulamaların özellikle evrensel olarak kullanılan internet üzerinde veriye ve diğer uygulamalara erişimi önemli bir gereksinim olarak karşımıza çıkmaktadır. Veb servisleri ve yazılım etmenleri gibi bu amaca hizmet eden ve bir ağ üzerinde bilginin paylaşılmasını sağlamak için standart protokolleri kullanan teknolojilerde de erişim denetimine ihtiyaç duyulmaktadır.

Erişim denetimi sıklıkla bir sistemdeki kaynaklara erişim kısıtı koymak şeklinde betimlenerek bu kaynakların sadece ayrıcalığa sahip olan elemanlar tarafından kullanılabildiği öngörülmektedir. Rol tabanlı erişim denetimi (RBAC) yaklaşımında ise sistemdeki elemanların erişim hakları, görevlerine ve gerçekleştirdikleri eylemlere göre düzenlenir [1].

RBAC modelleri; organizasyonel yapıları, modelleme yetenekleri ve yönetimsel yükleri azaltma potansiyelleri ile yoğun bir şekilde çalışılan araştırma konularındandır. Rollerin kalıtım hiyerarşisi sayesinde elemanların birbirleri ile olan ilişkileri rol seviyesinde düzenlenir. Bu yaklaşımla birlikte erişim kısıtları rollere verilerek bu rollere sahip elemanların yönetimi daha kolay bir şekilde gerçeklenir. RBAC modeli yaygın olarak kabul edilmekle birlikte varolan bir sisteme uygulanması bazı problemlerle karşılaşılmasına neden olabilir. Örneğin, varolan sistemlerin bir kısmı rol bazında organize edilmemiş olabilir. Fakat dağıtık ortamların çalışmasında kullanılan rol bazlı yaklaşımlar, özellikle yazılım etmenlerinin rol tabanlı ve hedef güdümlü mimarileri için uygun bir çözüm olanağı sunmaktadır [2].

Anlamsal Veb bağlamında ontolojiler, kavramların ve aralarındaki ilişkilerin biçimsel tanımlamalarını sağlar ve anlamsal seviyede heterojen ortamların birlikte çalışabilirliğinin üstesinden gelmek amacı ile kullanılır. Bu yaklaşım, şu ana kadar deklaratif veya sözdizimsel olarak gerçeklenmeye çalışılan erişim denetimi politika tanımlamalarının ilgili kavramlarının veya bu kavramlar arasındaki ilişkilerin, geliştirilebilecek ontolojiler yardımı ile standart bir gösterime sahip olmasını sağlayabilir. Böylece RBAC modeli anlamsal bir dil olan OWL ve kural dili olan SWRL yardımı ile geliştirilebilir [3].

II. GEÇMİŞ ÇALIŞMALAR

Özellikle veb servisleri üzerinde kullanılmak için tasarlanan XML tabanlı güvenlik dilleri, kapsamlı ve bütünleşik güvenlik çözümleri sunmayı hedeflemektedir. Bununla beraber erişim kontrolü alanında politikalar oluşturmak ve erişim kontrol kuralları belirlemek amacı ile bir dil ve bu yapıyı uygulamaya koymak için bir mimari üzerinde çalışmalar yapılmıştır. Bu çalışmalardan genel tanımlamaları bulunan XACML(Extensible Access Control Markup Language) [4] ve X-RBAC(XML Role Based Access Control) [5] ile veb servisleri üzerinde yapılan çalışmalarda kullanılabilen WS-Policy [6] kavramları akademide ve endüstride kabul görmüş yaklaşımlar olarak karşımıza çıkmaktadır.

XACML, erişim denetimi politikalarını ifade, değiş tokuş ve standartlaştırma amacı ile son dönemde OASIS tarafından önerilen bir yaklaşımdır. XACML, kendileri XML dili ile gösterilen nesnelerin yetkilendirme politikalarının da XML'de tanımlanması amacı ile tasarlanmıştır ve çoğu politika gösterim mekanizmalarının işlevselliklerini de tanımlayabilmektedir. Ayrıca farklı kuralları ve farklı politikaları değişik durumlar altında birleştirebilme ve RBAC ile kaynak koruma politikalarını konfigüre edebilme yetisine sahiptir.

Manuscript received November 28, 2008.

Fatih Tekbacak is with the Department of Computer Engineering, Izmir Institute of Technology, Izmir, 35430 Turkey. e-mail: (fatihtekbacak@iyte.edu.tr).

Asst. Prof. Dr. Tuğkan Tuğlular is with the Department of Computer Engineering, Izmir Institute of Technology, Izmir, 35430 Turkey. e-mail: (tugkantuglular@iyte.edu.tr).

Prof. Dr. Oğuz Dikenelli is with the Department of Computer Engineering, Ege University, Izmir, Turkey. e-mail: (oguz.dikenelli@ege.edu.tr).

X-RBAC ise RBAC modelinin genişletilmiş haline dayandırılmıştır. X-RBAC, farklı alanlarda bulunan ortamların arabuluculuk politikalarını belirtmek için bir çatı sunar ve RBAC modelini zamansal kısıtlar, rol nitelikleri, içeriksel haller ve durum değişimlerinin ön koşulları ile genişletmeye çalışır.

XACML ve X-RBAC, XML tabanlı yaklaşımlardır ve birçok dağıtık uygulamada kullanılmaktadırlar. Bununla beraber, XML politika tanımlamaları makine yorumlaması seviyesinde yeterli olamamaktadır. XML DTD ve şemaları verinin sözdizimsel belirtimlerinde yeteri kadar esneklik sağlar. Fakat verinin yorumlanması üzerindeki anlaşmalarda yeterli başarıyı gösterememektedirler, çünkü her uygulama alanı kendi terimlerinin anlamları üzerinde anlaşmaya varmak zorundadır. Güvenlik politikalarının makine tarafından yorumlanması, belirlenen bir politikanın yanlış kullanımını veya hatalı bir şekilde yayılımını engellemek açısından önem taşımaktadır.

Anlamsal veb araştırmaları, güvenlik gereksinimlerinin nasıl tanımlanacağı üzerinde de odaklanmaktadır. KAoS [7], veb servisleri, dağıtımlı hesaplama ve çoklu etmen sistem platformları için politikalar hakkında gösterim ve çıkarsama üzerine kurulu OWL tabanlı bir yaklaşımdır. KAoS; insan, etmen ve diğer işlevsel aktörlerin organizasyonlarını içeren alanlar hakkında gösterim ve çıkarsama yapmak amacı ile ontolojilerden yararlanır. Rei [8], politikaların anlamsal bir gösteriminin üzerine kurulu ödev mantığı tabanlı bir politika dilidir. Ponder [9], dağıtık sistemlerin ve ağların yönetimi için geliştirilmiş nesne tabanlı bir politika dilidir. Ponder dilinin geliştiricileri KAoS ve Rei'de kullanılan politika yönetim kavramlarının birçoğunun önderliğini yapmıştır. Fakat bu çalışmaların hiçbiri RBAC politika belirtimlerini içermediği için bu yöndeki ihtiyacı karşılamak gerekmektedir.

Son dönemde dağıtık bir ortamda gerçekleştirilebilecek uygulamaların güvenlik politikalarını doğru bir şekilde anlayabilmek ve yorumlayabilmek amacı ile, RBAC politikalarının gösterimi için OWL dili kullanılarak ontolojiler geliştirilmeye ve SWRL dili kullanılarak kurallar arasında sonuç çıkarılabilir ilişkiler kurulmaya çalışılmaktadır [3]. OWL, betimleme mantığının RDF dilindeki gösterimlerinin üzerine kurulu bilgi gösterim dilleri ailesindendir. Buna dayalı olarak politikalar tanımlarken OWL dilinin kullanımı birçok avantaj içermektedir. İlk olarak birçok politika dili; hedefleri, nesneleri, eylemleri ve zaman, mekan gibi diğer kısıtları içeren sınıfları tanımlamalıdırlar. Bir politika geliştirilirken onun mevcudiyeti bu sınıfların açık bir şekilde belirlenmesi ile oluşturulabilmektedir (tam zamanlı işçi veya umuma açık yazıcı kavramları gibi). Bu durum, özellikle politikaların kendi alanlarında şemaları ve veri modellerine sahip farklı organizasyonlar tarafından paylaşıldığında ciddi bir önem taşımaktadır. OWL'ın ikinci bir avantajı da analiz ve çalıştırma safhalarında kullanılmak amacı ile diğer mantıksal biçimlere dönüştürülebilmesidir. Bu yaklaşımla da rol tabanlı erişim denetiminin hangi kısımlarının betimleme mantığı ile, hangi kısımlarının ontolojik çıkarsama gereksinimleri ile özdeşleşebileceğine dair çalışmalar yapılmaktadır [10], [11].

Veb servisleri üzerinde kullanılan WS-Policy genel modeli ile XML tabanlı politika gösterimleri ve bu politikaların doğru çalıştıklarının gözlenebilmesi amacı ile sistem mimarileri üzerinde durulmuştur [12]. Ardagna ve arkadaşlarının yaptığı [12] çalışmada kullanılan politikaların yönetimi, değerlendirilmesi ve karar verilmesi aşamalarından oluşan üç katmanlı servis mimarisinin yazılım etmenlerinde de benzeri bir şekilde uygulanabileceği görülmüştür.

III. SİSTEM MİMARİSİ

Erişim denetimi politikalarını çoklu etmen mimarilerinde uygulayabilmek amacı ile karşılanmasına ihtiyaç duyulan; birimsellik, politika dilinden bağımsızlık, genişletilebilirlik, yeniden kullanılırlık, yüksek performans, donanım ve yazılımdan bağımsızlık gibi gereksinimler bulunmaktadır.

Şekil 1'de gösterilen mimari veb servisleri üzerinde gerçekleştirilen benzer bir yaklaşımın [12] çoklu etmen sistemlerine entegrasyonunu ele almaktadır. Sistem, kullanıcı etmeni, politika karar etmeni(PKE), politika değerlendirme etmeni(PDE), politika yönetim arayüzü ve politika veri havuzu isimli bileşenlerden oluşmaktadır.

A. Kullanıcı Etmeni

Rol tabanlı erişim hakkına sahip olabilmek amacı ile PKE'ye talepte bulunur. Bu talep sonucunda sahip olabileceği haklar veya eğer bir hakka sahip olamayacaksa bir istisna tanımı kullanıcı etmenine geri döner.

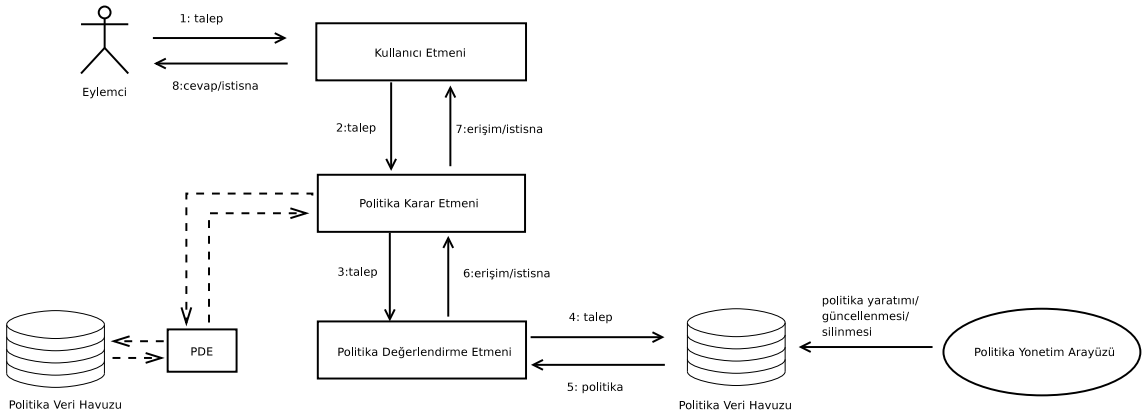
B. Politika Karar Etmeni

Kullanıcı etmeninden gelen talepleri PDE'ye aktarır ve geri dönen taleplerden erişim hakkı verilmişse erişim hakkını, verilmemişse sahip olduğu istisna işleme ontolojisi yardımı ile erişim kontrolüne neden sahip olmadığına dair bilgileri kullanıcı etmenine döndürür. Örneğin, kullanıcı sertifikasında PDE'nin kontrolü sırasında uygun olmayan bir parametre ile karşılaşıldığında erişim hakkının bu yüzden verilemeyeceği kullanıcı etmenine bildirilir. Ayrıca PKE'nin farklı PDE'lerin olduğu bir platformda talepleri birden çok PDE'ye yollama ve gelen verileri tümleştirip kullanıcı etmenine gönderme gibi arabuluculuk özelliğine de sahip olması düşünülmektedir.

C. Politika Değerlendirme Etmeni

PKE'den gelen talep doğrultusunda kullanıcı etmeninin sertifika bilgilerini kontrol ettikten sonra, Politika Veri Havuzu'na giderek o rolün diğer bir etmen üzerinde sahip olduğu yetkileri, gerektiğinde çıkarsama yaparak, döndürür. Politika veri havuzunda bulunan roller ve yetkilere göre PDE'de değerlendirme yapılmalıdır.

Farklı etmen platformlarında bulunan PDE'ler veya bu görevi üstlenen veb servislerine PKE'den gelen talep aynı anda iletilir. Bu taleplerin sonucunda farklı ortamlardan dönen bilgiler XACML politika entegrasyon algoritmaları [13], [14] yardımı ile PKE'de değerlendirilip yetkilere uygun kararlar verilir. Böylece çoklu etmen sistemlerinin otonom çalışan hedef güdümlü etmenleri, erişim denetimi hedefini gerçekleştiren etmenlerden aldıkları geri bildirim doğrultusunda çalışmalarına devam ederler.



Şekil 1. Çoklu etmen sistemleri için erişim denetimi mimarisi

D. Politika Veri Havuzu

Rolleri, rollerin sahip olduğu hedefleri ve rollerin sahip olduğu erişim politikalarını içeren bir bilgi tabanıdır. Sistem yöneticisi tarafından Politika Yönetim Arayüzü yardımı ile sürekli olarak yaratma/okuma/güncelleme/silme operasyonları ile sistemin güncel tutulmasına katkıda bulunur.

E. Politika Yönetim Arayüzü

Politikaların girilebilmesi için yöneticiye bir arayüz sunulacaktır. Bu aşamada varolan arayüzlerden de yararlanılması düşünülmektedir [15], [16]. Bu arayüzün önemli avantajlarından bir tanesi geçersiz politikalar girmenin engellenmesi veya bir role ait birbiri ile çelişen ekleme/güncelleme operasyonlarının yapılabilmesidir. Bu arayüzün içermesi düşünülen geçerlik sınaması işlevi sayesinde [17] politikaların iyi biçimlendirilmiş olarak Politika Veri Havuzu'nda tutulabilmesi sağlanacaktır.

Veb servisleri üzerinde uygulanan yaklaşımda [12], politikaların XML dosyaları içerisinde bulunması ve politika ambarı adı altında bir ilişkisel veritabanında tutulması öngörülüp bu veriler Servis, Metod ve Politika Konumu özelliklerine ait bir çokuzlu olarak tanımlanmakta iken çoklu etmen sistemlerinde bu yapının Rol, Hedef ve Yetki şeklinde tutulması planlanmaktadır.

IV. ERİŞİM DENETİMİ POLİTİKALARININ ANALİZİ

XACML, vebde bulunan kaynaklar üzerindeki politikaları belirtmek amacı ile kullanılan ifade yeteneği yüksek, OASIS standardı çerçevesinde tanımlanan XML tabanlı bir dildir. XACML gibi ifade yeteneği yüksek politika dilleri ile beraber çözülmesi gereken yeni durumlarla da karşılaşılmaktadır. Örneğin, kullanıcılar, güvenlik politikalarının içinde bulundukları sistemdeki etkisini ve sonuçlarını anlamakta güçlük çekmektedir. Bununla beraber, erişim denetimindeki en önemli durum, tanımlanan politikalardaki yetkilendirmelerin yetki dışı varlıklara izin vermesi ve bu durumun elle kontrol edilmeye çalışılmasıdır. Böylece eksik tanımlanmış güvenlik politikaları, saldırganlara istemeyerek de olsa erişim yetkisi verilmesine neden olabilmektedir.

Bahsedilen problemlerin üstesinden gelebilmek amacı ile, XACML biçimlendirmesinin doğrultusunda, birinci derece mantığın karar verilebilir alt kümelerinden oluşan bir dil ailesinin mensubu olan betimleme mantığı (DL) ve veb ontoloji dili (OWL) kullanılarak çalışmalar yapılmaktadır [18]. Politika analizinin, betimleme mantığının çıkarsama yetenekleri ile örtüşmesi, daha önceden betimleme mantığı kavramları temel alınarak gerçekleştirilmiş çıkarsama motorlarının kullanılabilmesine olanak sağlamaktadır. Tanımlanılacak çatıyı betimleme mantığı ve sonuç olarak veb ontoloji dili ile oluşturmaya çalışmanın bazı yararları vardır [18]:

- OWL dilinin doğası gereği isimlendirme amacı ile URI tanımlamaları kullanması ve ontolojiler arasında linkler gerçekleştirmeye izin vermesi, veb kaynakları üzerinde bir erişim denetim dili için uygun olabileceğini göstermektedir.
- OWL ile politikada kullanılan nesneler için ontoloji tabanlı tanımlamalar ile erişim denetim yaklaşımları geliştirilebilir. Betimleme mantığına dayalı çıkarsama motorları kullanılarak politikanın bulunduğu alanın tanımlamaları ile politikanın kendisi entegre edilebilir. Makine tarafından işlenebilir standart bir dil ile politikaların kendi alanlarında tanımladıkları kavramlar, veb üzerinde paylaşılabilir ve tekrar kullanılabilir.

Kolovski ve arkadaşlarının yaptığı çalışmalar [18] doğrultusunda, kendi çalışmamızda da açık kaynak kodlu bir betimleme mantığı çıkarsama motoru olan Pellet [19] kullanılması düşünülmektedir.

A. XACML'e Genel Bir Bakış

XACML politikalarının kökünde *Policy* ve *PolicySet* kavramları bulunmaktadır. *PolicySet*, diğer *Policy* ve *PolicySet* tanımlamalarını tutan bir barındıcıdır. Bunun yanında uzaktaki sistemlerde bulunan politikalara olan referansları da tutabilmektedir. Bir *Policy*, yalnızca bir erişim denetimi

politikasını gösterebilmekte ve *Rule* olarak tanımlanan kurallar kümesi ile ifade edilebilmektedir. Her XACML politika belgesi sadece ve sadece bir *Policy* veya *PolicySet* kök elemanını içermektedir. Bununla birlikte bir *Policy* veya *PolicySet*, birden fazla politika ve herbiri farklı erişim denetimi kararlarını etkileyebilecek kurallardan meydana gelebilmektedir.

Öznitelikler, bir XACML politikasının en temel birimidir. Erişim talebinin yapılmasında yardımcı olan *Subject*, *Resource*, *Action* veya *Environment* tanımlamalarının tipik özelliklerini gösterir. Örneğin, bir kullanıcının rolü, erişmek istediği dosya, şu anki zaman gibi kavramların hepsi öznitelik değerleridir. XACML'de gösterilen erişim talepleri, öznitelik-değer çiftlerinin bir listesini ifade etmektedir.

Rule olarak tanımlanan kurallar, XACML'in karar verme aşamasındaki en temel elemanıdır. Aslında bir *Rule*, erişim isteğini girdi olarak alan ve erişimle alakalı izin verip vermemeye karar veren bir fonksiyondur. Eğer bir kural, bir erişim talebine uygulanabilecek ise, *Target* elemanı kullanılır. Bir *Target*, verilen bir talebe uygulayabilmek amacı ile bir kurala uyan *Subject*, *Resource* ve *Action* tanımlamaları için kullanılan en basit durum setidir.

B. Politika İfadeleri

[11] ve [18] çalışmalarında kullanılan erişim politikalarındaki *Subject*, *Action* ve *Resource* kavramlarının betimleme mantığı tanımları ve rollerine eşleştirilmesi yaklaşımı, çalışmamıza da yol göstermektedir. Düşünülen yaklaşıma göre, varolan bir alanı modellemede kullanılan ontoloji ve altsınıf ilişkilerinden yararlanarak bu ontolojideki kavramlar ile politika varlıkları birbirine bağlanabilecektir.

Alan ontolojileri kullanarak yaygın politika tabirlerinin betimleme mantığında nasıl ifade edilebileceği bazı durumlar temel alınarak gösterilebilir [11], [18]:

- Rol hiyerarşileri altsınıf aksiyomları ile kolayca ele alınabilir. Bir altsınıf, mirasını aldığı üst sınıfın erişim ayrıcalıklarına sahiptir.
- Öznitelikler üzerindeki hiyerarşiler, betimleme mantığı kullanılarak ele alınabilir.
- Görev ayrımı kısıtları, ayrışık aksiyomları yardımı ile yakalanabilir.
- Nicelik kısıtları, verilen herhangi bir öznitelik üzerinde ifade edilebilir.

C. XACML'de Rol Tabanlı Erişim Denetimi

XACML'deki rol tabanlı erişim denetiminin politikaları ifade edebilmek için sahip olduğu profil, temelde beş tane veri elemanından meydana gelir [20]:

Kullanıcılar: XACML'in *Subject* tanımı kullanılarak gerçeklenirler.

Roller: XACML'in *Subject* özniteliklerinden bir veya birkaçını kullanarak ifade edilirler. Rollerin seti, uygulama ve politika alanına özgüdür.

Nesneler: XACML'in *Resource* tanımı kullanılarak ifade edilirler.

Operasyonlar: XACML'in *Action* tanımı kullanılarak ifade edilirler.

Yetkilendirmeler: XACML'in rol *PolicySet* ve yetki *PolicySet* örnekleri ile ifade edilirler.

V. ETMEN TABANLI MİMARİLERDE ROL KAVRAMI

Roller, etmenlerin yaygın olarak sahip olduğu davranışları tanımlayabilmek için yararlı bir soyutlama gösterimi olarak etmen sistemlerinde kullanılabilir. Rollerin çoklu etmen sistemlerinde iki farklı bakış açısından tanımlanabildiği görülmektedir [21]:

Kavramsal Bakış Açısı: Rol, bir etmenin bazı etkileşimler içinde yer aldığı ve belirli bir yönde evrimleştiği kısıtların bütünüdür. Özellikle çoklu etmen sistemlerinde, bir etmen yükümlü olduğu rollerin özelliklerine göre davranır.

Gerçekleştirim Bakış Açısı: Rol, onun yükümlü olduğu etmenin belirli özellikleri ve davranışlarının sarmalanmasıdır.

Roller, bir etmenin sistemde sahip olduğu hakları ve görevleri tanımlar. Bu tanımlar etmen sistemlerinin bazı karakteristik özelliklerinin altını çizmeye ihtiyaç duymamıza neden olmaktadır [21], [22]:

- Bir etmen aynı anda birden fazla role sahip olabilir.
- Etmenler dinamik olarak rollerini değiştirebilir.
- Etmenler (roller değil) eylemleri gerçekleştirirler.
- Roller birbirinden izole edilmiş değillerdir, diğer rollerle bağlantı halindedirler.
- Rol yardımı ile bir etmenin diğer etmenlerle nasıl etkileşim kurduğu bilinebilir.
- Roller, tekrar kullanıma yardımcı olur.

VI. ÇOKLU ETMEN SİSTEMLERİNDE ROL TABANLI ERİŞİM DENETİMİ YAKLAŞIMI

Bölüm IV-C'de anlatılan veri elemanları özel bir alandan bağımsız bir şekilde tanımlanmaktadır. Kolovski ve arkadaşlarının çalışmasında [18] ifade edilen betimleme mantığı yaklaşımı rol tabanlı erişim denetimi profiline uygulanabilir ve başka ortamlara adapte edilebilir. Çalışmamızın katkısı ise, bu profilde tanımlanan kavramların açıklanan yaklaşımın uygulanabileceği bir alan olarak çoklu etmen platformlarına entegrasyonudur.

XACML ifadelerinin temel tanımlamaları yanında yazılım etmenlerinde bulunan Hedef kavramının da anlatılan yaklaşıma eklenmesi düşünülmektedir. Böylece farklı rollere sahip olan etmenlerin, hedefler üzerinde yetkilerini kullanabilmesi, farklı rollerin sahip olduğu ortak hedefleri paylaşabilmesi, rollerin delegasyonu gibi yaklaşımların oluşturulabilmesi öngörülmektedir.

```
<owl:Class rdf:ID="Goal"/>
<owl:Class rdf:ID="CommonGoal"/>
```

Bu eklemeler ortamın alan ontolojisinde tanımlanacaktır. XACML-DL olarak adlandırılan XACML'e betimleme mantığının eklenerek geliştirildiği yapı ve ortamın sahip olduğu yazılım etmenlerine özgü alan ontolojisi Pellet [19] isimli bir çıkarsama motoru üzerinde çalıştırılacak; böylece erişim talebinde bulunan etmenin isteği Şekil 1'de tanımlanan mimarideki etmenlerden geçecektir. Sonuç olarak istekte

bulunan etmene rol tabanlı erişim detayları hakkında bilgi dönüşü sağlanacaktır.

Bir rolün diğer bir rolle ortak hedeflerini, farklı rollerin veya hedeflerin delegasyonu ilişkilerini belirleyecek atama ifadelerinin aşağıdaki gibi tanımlanması düşünülmektedir.

```
<owl:ObjectProperty rdf:ID="hasGoal">
 <rdfs:range rdf:resource="#Goal"/>
 <rdfs:domain rdf:resource="#Role"/>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="hasCommonGoal">
  <rdfs:subPropertyOf
    rdf:resource="#hasGoal">
  <rdfs:range rdf:resource="#Role"/>
  <rdfs:domain rdf:resource="#Role"/>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="hasRDelegate">
   <rdfs:range rdf:resource="#Role"/>
   <rdfs:domain rdf:resource="#Role"/>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="hasGDelegate">
   <rdfs:range rdf:resource="#Goal"/>
   <rdfs:domain rdf:resource="#Role"/>
</owl:ObjectProperty>
```

Üzerinde durulması gereken önemli noktalardan bir tanesi de bir etmenin tek bir role bağlı olmamasıdır. Etmenler, yaşam süreçleri boyunca görevlerini gerçekleştirebilmek amacı ile değişik rollere sahip olabilirler veya onları bırakabilirler. Rol ataması, etmen tasarım aşamasında statik olarak tanımlanabileceği gibi eğer etmen onun görevi için en uygun rolü seçebilme yetisine sahip ise dinamik de olabilir. Etmenlerin, tasarım aşamasında veya çalışma aşamasında sahip olduğu rollerin ve ilgili izinlerin birbirleri ile çelişmesi gibi durumları engellemek için statik görev ayrımı ve dinamik görev ayrımı kavramları bulunmaktadır. Bu kavramlar Şekil 1'de gösterilen etmenlerde var olan XACML yapılarında XACML-DL ile alan ontolojilerinde var olan OWL tanımlamalarında ise SWRL kural dili yardımı ile gerçeklenecektir. Böylece kullanıcı, rol ve görev arasındaki çelişkileri ortadan kaldırmaya yardımcı olacak kurallar tanımlayıp çalıştırabilecektir. Kurallar tanımlanmadan önce roller arasındaki çelişki ilişkisinin tanımlanması gerekmektedir [23].

Her etkileşim boyunca etmenler birden çok role sahip olabilirler. Çalışma süresi boyunca var olan rollere aktif rol denir. Odell ve arkadaşlarının yaptığı çalışmaya göre [24] etmenlerin sahip olduğu roller, dönemlik aktiviteye göre aktifleştirilebilir, askıya alınabilir veya iki rolün dinamik görev ayrımı kısıtlarına göre değiştirilebilir. Örneğin, aynı anda personel ve yönetici rollerini üstlenen bir etmen, hedef ve ihtiyaçlar doğrultusunda yönetici rolünü aktive ederek personele talepte bulunabilir veya yönetici rolünü askıya alıp personel rolünü aktifleştirerek üst kademesinde bulunan müdürüne raporlar hazırlama hedefine sahip olabilir.

Etmen tabanlı yazılım geliştirmede, rolleri kullanarak etkileşimi modellemenin farklı avantajları vardır [25]. İlki, etmen tabanlı uygulamaları geliştirirken, algoritmik kavramlar ile etkileşim kavramları arasındaki ayrılıkları belirlemektir. İkinci olarak çözümlerin ve deneyimlerin tekrar kullanımına olanak sağlanır. Aslında roller bir bağlam ile ilişkilidir ve tasarımcılar önceden tanımlanmış rolleri, uygulamalarının sahip olduğu bağlama göre kullanabilir. Böylece roller, bir çeşit tasarım desenine dönüşür [26].

VII. SONUÇLAR

Bu çalışmadaki amaç, yaygın olarak kullanılan sözdizimsel rol tabanlı erişim denetimi yaklaşımlarının entegre şekilde çalışabildiği ontoloji güdümlü bir rol hiyerarşisi ve katmanlı bir erişim denetimi mimarisini çoklu etmen sistemlerinin rol tabanlı etmenlerinde uygulayabilmektir. Bu mimarideki rollere sahip etmenlerin kimlik doğrulamasının ise TÜBİTAK Kamu Sertifikasyon Merkezi aracılığı ile elde edilen akıllı kartların içerdiği sertifikalar yardımı ile gerçeklenmesi düşünülmektedir.

Rol tabanlı erişim denetim mimarisi için, XACML modeline dayalı ontoloji güdümlü erişim denetim modelinin geliştirilmesi, örnek rol senaryoları üzerinden politika veri havuzuna bir arayüz yardımı ile politikalar eklenmesi amaçlanmaktadır.

Bu çalışma sürecinde, sistemin ihtiyaçlarını karşılamak amacı ile etmenlerin sorumluluklarını çözümleyebilecek farklı rol belirtim senaryoları üzerinden roller tanımlanacak, etmenler tarafından sahip olunan bu rollerin Şekil 1'de gösterilen sistem mimarisinde belirtilen katmanlardan geçebilmesi için gerekli olan görevler, iletişim yetenekleri ve etmenler arası bağımlılıklar SEAGENT [27] çoklu etmen sisteminin rol modeli temel alınarak gerçekleştirilmeye çalışılacaktır.

KAYNAKLAR

[1] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, C. E. Youman, *Role-Based Access Control Models*. IEEE Computer 29(2): 38-47, 1996.

[2] F. Chen, S. Li, H. Yang, *Enforcing Role-Based Access Control Systems with an Agent Based Service Oriented Approach*, Proceedings of the 2007 IEEE International Conference on Networking, Sensing and Control, April 2007.

[3] D. Wu, X. Chen, J. Lin, M. Zhu, *Ontology-Based RBAC Specification for Interoperation in Distributed Environment*, ASWC 2006: 179-190, 2006.

[4] T. Moses, ed., *OASIS eXtensible Access Control Markup Language (XACML) Version 2.0*, 24 July 2003.

[5] J. B. D. Joshi, *Access-control language for multidomain environments*. Internet Computing, IEEE, Vol 8, Is 6. IEEE Inc., Piscataway, pp. 40-50, 2004.

[6] D. Box, *Web Services Policy Framework (WS-Policy) version 1.1*, May 2003.

[7] A. Uszok, J. M. Bradshaw, M. Johnson, R. Jeffers, A. Tate, A. Dalton, S. Aitken, *KAoS Policy Management for Semantic Web Services*, IEEE Intelligent Systems, v.19 n.4, p.32-41, July 2004.

[8] L. Kagal, T. Finin, A. Johshi, *A Policy Language for Pervasive Computing Environment*. In proceedings of IEEE 4th International Workshop on Policies for Distributed Systems and Networks (POLICY 2003), Lake Como, Italy, 2003.

[9] N. Damianou, N. Dulay, E. Lupu, M. Sloman: *The Ponder Policy Specification Language*. In proceedings of Workshop on Policies for Distributed Systems and Networks (POLICY 2001). Springer-Verlag, Bristol, UK, 2001.

[10] T. W. Finin, A. Joshi, L. Kagal, J. Niu, R. S. Sandhu, W. H. Winsborough, B. M. Thuraisingham, *ROWLBAC: Representing role based access control in OWL*, in Proc. SACMAT, pp.73-82, 2008.

[11] M. Smith, A. Schain, K. Clark, A. Griffey, V. Kolovski: *Mother, May I? OWL-based Policy Management at NASA*, Proceedings of the OWLED 2007 Workshop on OWL: Experiences and Directions, Innsbruck, Austria, June 6-7, 2007.

[12] C. A. Ardagna, E. Damiani, S. D. C. Di Vimercati, P. Samarati, *A Web Service Architecture for Enforcing Access Control Policies*, Proceedings of the First International Workshop on Views on Designing Complex Architectures, Electronic Notes in Theoretical Computer Science, Volume 142, Pages 47-62, January 2006.
[13] P. Mazzoleni, E. Bertino, B. Crispo, S. Sivasubramanian, *XACML policy integration algorithms: not to be confused with XACML policy combination algorithms!*, Proceedings of the eleventh ACM symposium on Access control models and technologies, Pages: 219 - 227, 2006.
[14] P. Mazzoleni, B. Crispo, S. Sivasubramanian, E. Bertino, *XACML Policy Integration Algorithms*, ACM Transactions on Information and System Security (TISSEC), Volume 11, Issue 1, February 2008.
[15] Xacml Studio, http://xacml-studio.sourceforge.net/.
[16] Policy Studio, http://www.nextlabs.com/html/?q=policy-studio.
[17] Margrave, http://www.cs.brown.edu/research/plt/software/margrave/.
[18] V. Kolovski, J. Hendler, B. Parsia, *Analyzing Web Access Control Policies*, Security, Privacy, Reliability and Ethics Track, Access Control and Trust on the Web Session(WWW 2007), May 2007.
[19] E. Sirin, B. Parsia, B. C. Grau, A. Kalyanpur, Y. Katz, *Pellet: A practical OWL-DL reasoner*, J. Web Sem. 5(2): 51-53, 2007.
[20] A. Anderson, *Core and hierarchical role based access control (RBAC) profile of XACML v2.0*, OASIS Standard, 1 February 2005.
[21] Q. Yan, L. J. Shan, X. J. Mao, Z. C. Qi, *RoMAS: A Role-Based Modeling Method for Multi-Agent System*, Proceedings of International Conference on Active Media Technology, 2003.
[22] M. Puviani, G. Cabri, L. Leonardi, *Agent Roles: From Methodologies to Infrastructures*, International Symposium on Collaborative Technologies and Systems(CTS 2008), May 2008.
[23] W. Di, L. Jian, D. Yabo, Z. Miaoliang, *Using Semantic Web Technologies to Specify Constraints of RBAC*, Proceedings of the Sixth International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT'05), 2005.
[24] J. Odell, H. V. D. Parunak, S. Brueckner, J. Sauter, *Changing Roles: Dynamic Role Assignment*, Journal of Object Technology, volume 2, pages 77-86, 2003.
[25] G. Cabri, L. Ferrari, L. Leonardi, *Supporting the Development of Multi-Agent Interactions via Roles*, AOSE 2005: 154-166, 2005.
[26] Y. Aridor, D. Lange, *Agent Design Pattern: Elements of Agent Application Design*, International Conference on Autonomous Agents, ACM Press, 1998.
[27] O. Dikenelli, *SEAGENT MAS platform development environment*, AAMAS (Demos) 2008: 1671-1672, 2008.