Ontologías en la gestión de redes basada en políticas

Leandro Tabares Martín[1]

Universidad de las Ciencias Informáticas, La Habana, Cuba
ltmartin@uci.cu

Abstract. The main goal of the current work was to create a mechanism for detecting and classifying events coming from the statistical information gathered when operating a computer's network. To achieve this goal we have developed a Web ontology and used HermiT reasoner on the instance of the ontology. The created mechanism allowed to make explicit the classification of events raised on the network for applying policies.

Keywords: Reasoning, Ontologies, Policy-based Network Management

Resumen El presente trabajo tuvo como objetivo desarrollar un mecanismo que permitiese la detección y clasificación de eventos a partir de la información estadística que se obtiene al operar una red informática. Para cumplimentar el objetivo se desarrolló una ontología que describe los eventos a detectar en el Lenguaje de Ontologías Web y se aplicó el razonador HermiT sobre la instanciación de la ontología desarrollada. El mecanismo creado permitió explicitar la ocurrencia de los eventos monitoreados en la red y clasificarlos con el fin de aplicar políticas ante su ocurrencia.

Keywords: Gestión de redes basada en políticas, Razonamiento, Ontologías

1 Introducción

En su formulación, un enfoque basado en políticas presenta tres elementos básicos: eventos de políticas, acciones de políticas y reglas de políticas. Los eventos de políticas representan los estados del sistema que son relevantes en el contexto de los objetivos de negocio y su realización operacional. Las acciones de políticas son las respuestas deseadas por la organización en caso de que ocurra uno o más eventos de políticas. Las reglas de políticas son los mecanismos que enlazan los eventos de políticas con las acciones de políticas [2].

Un enfoque basado en políticas para la gestión también puede permitir la separación de las reglas que gobiernan el funcionamiento de un sistema de la

funcionalidad provista por el sistema. Por lo tanto, una gestión basada en políticas permite adaptar el comportamiento de un sistema sin refactorizar sus funcionalidades [1].

Una arquitectura de sistemas basados en políticas sencilla pero poderosa fue formulada por [10] y se ilustra en la figura 1. El Punto de Decisión de Políticas (PDP) es el módulo donde reside la experticia sobre políticas. El PDP utiliza un repositorio que contiene los parámetros y estructuras de datos necesarios para que el PDP evalúe las políticas y pueda realizar decisiones basadas en ellas. El repositorio de políticas comúnmente se implementa como un directorio que se comunica usando un protocolo de acceso a directorios ligero (LDAP) [2].

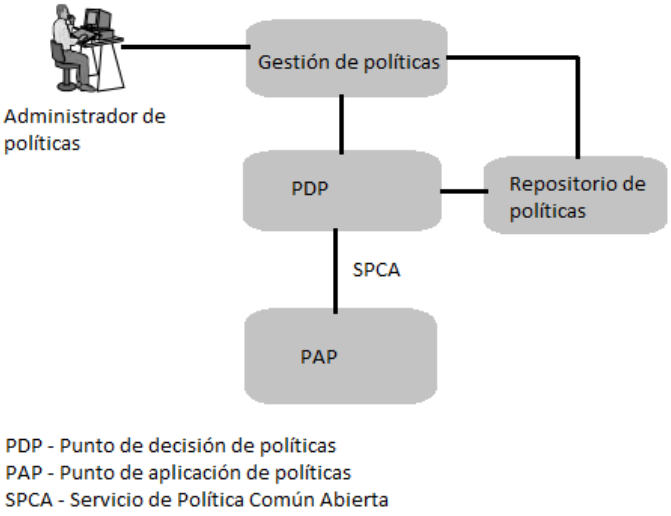

Figura 1. Arquitectura de referencia de políticas.

Cada objeto gestionado soporta un agente de políticas conocido como punto de aplicación de políticas (PAP). Se diseña e implementa un PAP por cada objeto gestionado en el sistema basado en políticas. El PAP se comunica con el PDP por medio de un protocolo denominado servicio de políticas común abierto (SPCA) [4]. Sin embargo, según plantea [2], la lógica dentro de las reglas que soportan las políticas deben ir más allá del simple si-entonces de la lógica que permiten las políticas basadas en tablas. Una forma de expresar reglas basadas en lógica descriptiva es la utilización de ontologías.

Una ontología es una especificación explícita y formal de una conceptualización compartida [7].

- Es explícita porque define los conceptos, propiedades, relaciones, funciones, axiomas y restricciones que la componen.
- Es formal porque es legible e interpretable por computadoras.
- Es una conceptualización porque es un modelo abstracto y una vista simplificada de los elementos reales que representa.
- Es compartida porque se ha arribado previamente a un consenso sobre la información y es aceptada por un grupo de expertos.

Uno de los escenarios más exitosos de uso de las ontologías es la Web Semántica [9]. Las ontologías en la Web Semántica se expresan en un lenguaje denominado Lenguaje de Ontologías Web (OWL) basado en lógica descriptiva. Debido a que OWL es un lenguaje de ontologías lo suficientemente completo puede ser usado directamente para especificar información sobre la gestión de redes porque presenta la mayoría de los elementos incluidos en los lenguajes de gestión de información y, los que no están incluidos, pueden serlo al extender OWL [8].

Durante varios años la gestión de redes basada en políticas ha atraído tanto a la academia como a la industria [9]. El término política en el contexto de la gestión de redes significa una regla que gobierna la elección en el comportamiento de los elementos gestionados [9]. Existen varios lenguajes para la gestión de políticas, entre ellos se encuentra PONDER [3], que fue especialmente concebido para la gestión de redes basada en políticas. Sin embargo, la utilización de varios lenguajes de políticas puede causar dificultades en la implementación de sistemas de gestión. Al utilizar ontologías OWL las políticas de gestión de red pueden combinarse en el mismo modelo con la información de gestión de red, lo que apunta en una dirección promisoria para la automatización de la gestión de redes [9].

2 Materiales y métodos

Para la definición de reglas se creó una ontología que permite aplicar políticas en función de eventos que se puedan producir en la red y que se describe a continuación.

$$DoSAttacksNetwork \equiv DisallowedNetwork \sqcap \geq 2000 connectionsPerSecond$$
$$HotAttacksNumberNetwork \equiv DisallowedNetwork \sqcap \exists phoneNumber.Literal$$
$$DisallowedNetwork \equiv Network$$
$$MultipleFailedLoginUser \equiv DisallowedUser \sqcap \geq 4 failedLoginAttempts$$
$$BlockedUser \equiv DisallowedUser \sqcap \exists identifier.Literal$$
$$DisallowedUser \equiv User$$
$$DisallowPolicy \equiv DisallowedNetwork \sqcup DisallowedUser$$

Una vista en Protégé de las clases correspondientes a ontología creada se ilustra en la figura 2.

Para realizar las tareas de razonamiento e inferencia se utilizó el razonador sobre OWL 2 HermiT. HermiT soporta todas las características del lenguaje OWL 2 según el estándar del consorcio de la World Wide Web (W3C), se basa

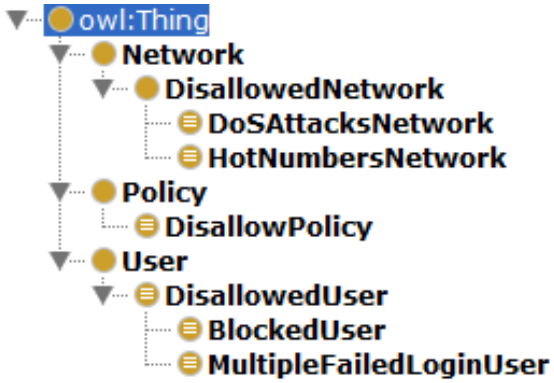

Figura 2. Clases de la ontología creada.

en cálculo hypertableau y soporta un amplio rango de optimizaciones estándares y otras novedosas para mejorar el rendimiento al razonar sobre ontologías en un ambiente no experimental [5].

Se decidió emplear un razonador con el objetivo de detectar y clasificar eventos de interés que ocurren en la red a partir de parámetros definidos en la ontología. Resulta válido anotar que los parámetros definidos en este trabajo tienen un propósito experimental, estos pueden ser variados en función de las necesidades de detección y clasificación de eventos en una instancia de red específica.

Para el desarrollo de la ontología propuesta se hizo un análisis de las diferentes metodologías existentes basado en los criterios propuestos por [6]. Los resultados de este análisis se ilustran en la tabla 1. Posterior al análisis de diferentes metodologías para el desarrollo de ontologías se decidió utilizar METHONTOLOGY ya que recomienda un ciclo de vida, es reutilizable y provee suficientes detalles sobre las técnicas y actividades empleadas en ella.

3 Resultados y discusión

Luego del diseño de la parte terminológica (T-BOX) de la ontología propuesta se procedió a instanciarla con el fin de verificar su consistencia y correcto funcionamiento para el propósito que fue creada. Con este fin se crearon en Protégé los individuos que se ilustran en las figuras 3, 4 y 5. Debe notarse que en esta instaciación solo se hacen explícitos parámetros estadísticos que pueden ser obtenidos a partir de las estadísticas de una red.

A partir de la aplicación de un razonador sobre la información estadística explícita descrita en la instanciación de la ontología (A-BOX), se hace explícita la información implícita que se ilustra en las figuras 6, 7 y 8. Al contrastar estas figuras con sus correspondientes en las figuras 3, 4 y 5 se evidencia que el razonador fue capaz de analizar la información explícita de los individuos, aplicar las

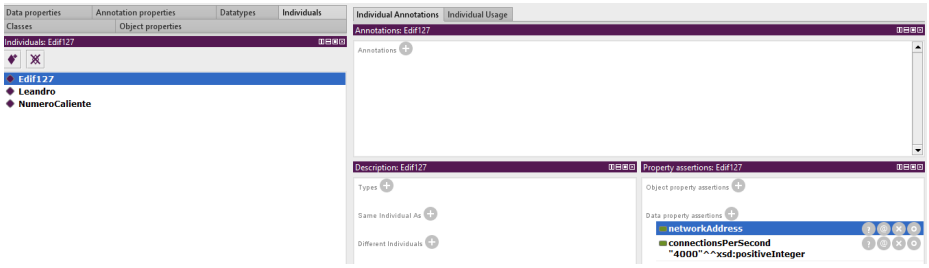

Figura 3. Individuo que representa a una subred.

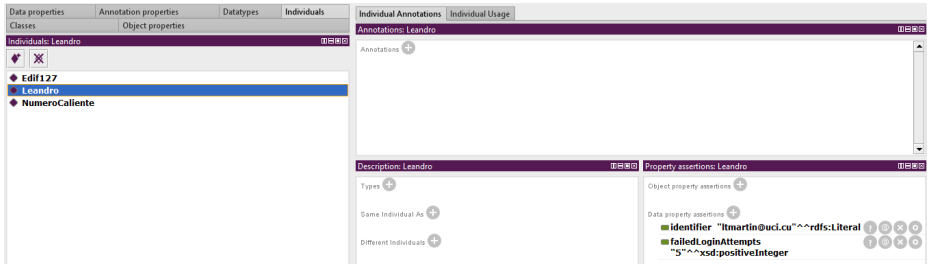

Figura 4. Individuo que representa a un usuario con múltiples intentos de autenticación fallidos.

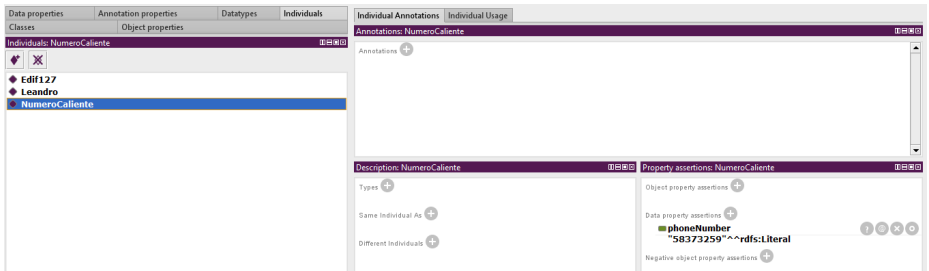

Figura 5. Individuo que representa a un número caliente sobre el que se deben realizar acciones.

Cuadro 1. Comparación de las metodologías analizadas.

Metodologías	Tipo de Desarrollo	Construcción colaborativa	¿Reutilizable?	Dependiente de la aplicación	Ciclo de vida	Estrategias para identificar los conceptos	Nivel de detalle	¿Interoperable?
TOVE	Basada en etapas	No	Sí	Semi independiente	No	Media	Algunos detalles	No
Enterprise model approach	Basada en etapas	No	Sí	Independiente	No	Media	Algunos detalles	No
METHONTOLOGY	Prototipo evolutivo	No	Sí	Independiente	Sí	Media	Suficientes detalles	No
KBSI IDEF5	Prototipo evolutivo	No	Sí	Independiente	No	No es clara	Algunos detalles	No
Ontolingua	Desarrollo modular	Sí	Sí	Independiente	No	No es clara	Algunos detalles	Sí
KACTUS	Desarrollo modular	No	Sí	Dependiente	No	Estrategia arriba-abajo	Insuficientes detalles	No
PLINIUS	Basada en guías	No	No	Independiente	No	Estrategia abajo-arriba	Algunos detalles	No
ONIONS	Desarrollo modular basado en guías	No	No	Dependiente	No	No es clara	Insuficientes detalles	Sí
Mikrokosmos	Basada en guías	No	No	Dependiente	No	Estrategia basada en reglas	Algunos detalles	No
MENELAS	Basada en guías	No	No	Dependiente	No	Grafos de conceptos	Insuficientes detalles	No
SENSUS	No menciona preferencias	Sí	Sí	Semi independiente	No	Abajo-arriba	Algunos detalles	Sí
Cyc	Prototipo evolutivo	No	Sí	Independiente	No	No es clara	Algunos detalles	No
UPON	Prototipo evolutivo	No	Sí	Independiente	Sí	Media	Algunos detalles	No
Método 101	Prototipo evolutivo	No	Sí	Independiente	No	Consenso del desarrollador	Algunos detalles	No
On-To-Knowledge	Prototipo evolutivo	No	No	Dependiente	Sí	Media	Algunos detalles	No

reglas definidas en la ontología y clasificar cada tipo de individuo en dependencia del evento que está siendo objeto de monitorización en la red. Al detectar las instancias de los eventos que se observan en la red, el punto de aplicación de políticas (PAP) definido en la arquitectura es capaz de desencadenar las acciones respectivas a cada tipo de evento.

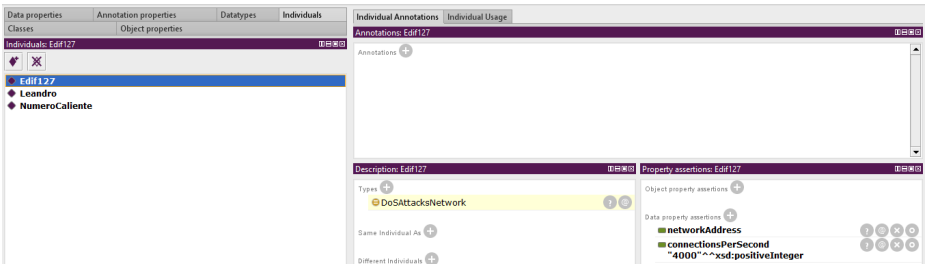

Figura 6. Individuo que representa a una subred luego de aplicar un razonador.

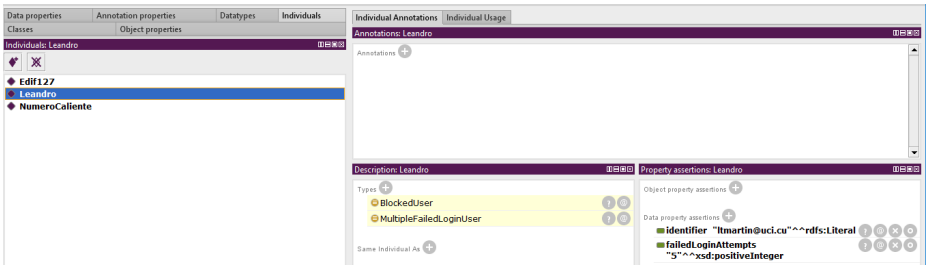

Figura 7. Individuo que representa a un usuario con múltiples intentos de autenticación fallidos luego de aplicar un razonador.

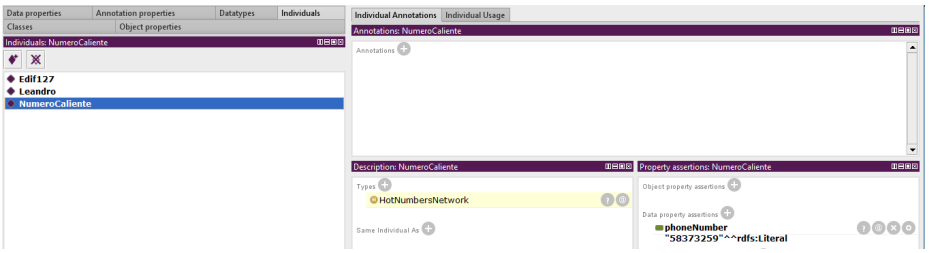

Figura 8. Individuo que representa a un número caliente sobre el que se deben realizar acciones luego de aplicar un razonador.

4 Conclusiones

A partir del análisis realizado se puede concluir que la metodología para el desarrollo de ontologías METHONTOLOGY satisface las necesidades para la creación de una ontología para la detección y clasificación de eventos en la red. La aplicación de un razonador compatible con el lenguaje OWL 2 permite hacer explícita información implícita en simples datos estadísticos recopilados en la red. A partir de la instanciación de la ontología propuesta fue posible detectar eventos producidos en la red y clasificarlos.

En el futuro resulta promisoria la investigación en el área de la actualización incremental de grafos RDF como instanciación de la ontología propuesta. Esto permitirá aumentar la eficiencia y eficacia en la detección y clasificación de los eventos que se producen en la red.

Referencias

1. Bandara, A.K., Lupu, E.C., Russo, A.: Using event calculus to formalise policy specification and analysis. In: Proceedings POLICY 2003. IEEE 4th International Workshop on Policies for Distributed Systems and Networks. pp. 26–39 (June 2003)
2. Choudhary, A.R.: Policy-based network management. Bell Labs Technical Journal 9(1), 19–29 (2004)
3. Damianou, N., Dulay, N., Lupu, E., Sloman, M.: The Ponder Policy Specification Language. In: Sloman, M., Lupu, E.C., Lobo, J. (eds.) Policies for Distributed Systems and Networks. pp. 18–38. Springer Berlin Heidelberg, Berlin, Heidelberg (2001)
4. Durham, D., Boyle, J., Cohen, R., Herzog, S., Rajan, R., Sastry, A.: The cops (common open policy service) protocol. Tech. rep. (1999)
5. Glimm, B., Horrocks, I., Motik, B., Stoilos, G., Wang, Z.: HermiT: An OWL 2 Reasoner. Journal of Automated Reasoning 53(3), 245–269 (oct 2014), `https://doi.org/10.1007/s10817-014-9305-1`
6. Iqbal, R., Murad, M.A.A., Mustapha, A., Sharef, N.M.: An analysis of ontology engineering methodologies: A literature review. Research journal of applied sciences, engineering and technology 6(16), 2993–3000 (2013)
7. Studer, R., Richards Benjamins, V., Fensel, D.: Knowledge engineering: Principles and methods. Data & Knowledge Engineering 25(1-2), 161–197 (1998)
8. de Vergara, J.E.L., Villagra, V.A., Berrocal, J.: Applying the web ontology language to management information definitions. IEEE Communications Magazine 42(7), 68–74 (July 2004)
9. Xiao, D., Xu, H.: An integration of ontology-based and policy-based network management for automation. In: 2006 International Conference on Computational Inteligence for Modelling Control and Automation and International Conference on Intelligent Agents Web Technologies and International Commerce (CIMCA'06). pp. 27–27 (Nov 2006)
10. Yavatkar, R., Pendarakis, D., Guerin, R.: A framework for policy-based admission control. Tech. rep. (1999)