Identificación de aplicaciones de red mediante técnicas de aprendizaje automático

Network Application Identification based on Machine Learning Techniques

Pere Barlet-Ros, Eva Codina, Josep Solé-Pareta

Resumen

El presente artículo describe un nuevo método de identificación de aplicaciones de red basado en técnicas de aprendizaje automático. Este método pretende solucionar las limitaciones de precisión de las técnicas tradicionales que utilizan los números de puerto y los problemas de rendimiento que presentan los algoritmos basados en el reconocimiento de patrones. Este método ha sido evaluado experimentalmente en la Anella Científica, obteniendo un porcentaje de identificación superior al 95%.

Palabras clave: identificación de aplicaciones, clasificación de tráfico, monitorización pasiva.

Summary

This paper describes a novel method for network application identification using machine learning techniques. This method tries to solve the accuracy limitations of traditional port-based techniques and performance issues of previous payload-based algorithms. This method has been experimentally evaluated in the Scientific Ring network, obtaining an identification accuracy greater than 95%.

Keywords: application identification, traffic classification, passive monitoring.

Nuevo método de identificación de aplicaciones de red basado en técnicas de aprendizaje automático

El método ha sido evaluado de forma experimental en la Anella Científica, obteniendo un porcentaje de identificación superior al 95%

1. Introducción

Los sistemas pasivos de monitorización de red permiten observar en tiempo real el tráfico de una red operativa sin interferir en su funcionamiento habitual. Estos sistemas son ampliamente utilizados por los operadores y administradores de red como herramientas de soporte a las tareas de dimensionado, evaluación de rendimiento, detección de fallos y anomalías, etc. Un posible ejemplo de sistema de monitorización pasiva es la plataforma SMARTxAC[1] desarrollada por la UPC y utilizada por el CESCA para la monitorización continua de la Anella Científica.

Una de las funcionalidades más interesantes de estos sistemas es la identificación de las aplicaciones de red. Es decir, la clasificación del tráfico de red según la aplicación que lo ha generado. Tradicionalmente, esta identificación se ha realizado utilizando únicamente los números de puerto presentes en las cabeceras de los paquetes, debido a que cada aplicación tenía asignada un número (o rango) de puerto/s predefinido/s (p.ej. well-known ports asignados por la IANA).

Sin embargo, actualmente está ampliamente aceptado que los números de puerto ya no proporcionan información suficientemente fiable para la identificación precisa de las aplicaciones de red. Entre las principales causas destacan la aparición de un gran número de aplicaciones basadas en web, aplicaciones que usan puertos dinámicos (p.ej. FTP en modo pasivo) o la utilización de túneles. Además, una práctica común entre los usuarios consiste en cambiar los números de puerto definidos para algunas aplicaciones (p.ej. P2P) para evadir cortafuegos, impedir su detección o evitar ataques de seguridad. Incluso, las nuevas generaciones de aplicaciones P2P ya incorporan estrategias de encriptación y ofuscación del protocolo para dificultar su detección.

Hasta el momento, varios trabajos de investigación han propuesto diferentes soluciones al problema [2-5], la mayoría de ellos con un éxito relativamente limitado. Una de las alternativas más comúnmente utilizada por los administradores de red es la técnica de reconocimiento de patrones (p.ej. L7-filter[6]). No obstante, esta técnica presenta tres limitaciones importantes. En primer lugar no es adecuada para la monitorización de redes de alta velocidad, ya que los algoritmos de búsqueda de patrones son muy costosos computacionalmente. En segundo lugar, no es aplicable cuando se utilizan técnicas de encriptación de la conexión. Y por último, la captura e inspección del contenido de los paquetes puede presentar problemas de privacidad.

En este trabajo presentamos un nuevo método de identificación de aplicaciones de red basado en técnicas de aprendizaje automático inductivo que soluciona los problemas descritos anteriormente. Este método ha sido implantado en el sistema SMARTxAC de monitorización de tráfico y evaluado de forma experimental en la Anella Científica, obteniendo un porcentaje de identificación superior al 95%.

2. ¿Qué es el aprendizaje automático?

El aprendizaje automático inductivo es una rama de la inteligencia artificial que permite a las computadoras extraer de forma automatizada conocimiento a partir de un número limitado de ejemplos, conocido como conjunto de entrenamiento. El campo del aprendizaje automático inductivo se divide en dos grandes grupos: los algoritmos supervisados y los no supervisados. Mientras que los algoritmos no supervisados consisten en encontrar la partición más adecuada del conjunto de entrada a partir de similitudes entre sus ejemplos, los algoritmos supervisados intentan extraer aquellas propiedades que permiten discriminar mejor la clase de cada ejemplo, y como consecuencia requieren de una clasificación previa (supervisión) del conjunto de entrenamiento. En este caso, los ejemplos que forman el conjunto de entrenamiento normalmente se componen por pares del estilo <objeto de entrada, clase del objeto>, donde el objeto de entrada suele estar representado por un vector de atributos (o propiedades del objeto). La misión de los algoritmos de aprendizaje automático supervisados es por tanto encontrar el conjunto de atributos que permite predecir con mayor precisión la clase de cada objeto del conjunto de entrenamiento.

El campo del aprendizaje automático inductivo se divide en dos grandes grupos: los algoritmos supervisados y los no supervisados

3. Método de identificación de las aplicaciones de red

En el presente trabajo, la identificación de las aplicaciones de red se realiza mediante el algoritmo de aprendizaje automático supervisado C4.5 [7]. Este algoritmo es una extensión del conocido algoritmo ID3, ambos desarrollados por Ross Quinlan, que consiste en la construcción de un árbol de clasificación basándose en el concepto de Entropía de Shannon. El proceso de construcción de este árbol consiste en seleccionar recursivamente un atributo para dividir el conjunto de entrenamiento en subconjuntos menores. Concretamente, el criterio básico utilizado para seleccionar el atributo más apropiado a cada paso del algoritmo es la elección de aquél que obtiene la máxima ganancia de información.

En nuestro caso, el conjunto de entrenamiento está formado por flujos de tráfico reales, donde el vector de atributos incluye características particulares de estos flujos. En este trabajo definimos como flujo de tráfico a una conexión a nivel de transporte (p.ej. TCP o UDP), aunque otras definiciones también serían posibles. El requisito principal de los atributos utilizados por nuestro método es que el sistema de monitorización de tráfico los pueda calcular de forma simple y en tiempo real (p.ej. longitud media de los paquetes del flujo, duración media de la conexión, tiempo medio entre llegadas de paquetes, etc.), y que no sea necesario el análisis del contenido total de los paquetes. En [8] puede encontrarse una descripción detallada de los 25 atributos utilizados en este trabajo. Adicionalmente, se ha estudiado, aunque sin obtener una mejora significativa, la inclusión de un atributo definido como el resultado de aplicar el método de búsqueda de patrones en los primeros bytes del contenido de los paquetes (proceso que aún puede realizarse en tiempo real).

En este trabajo definimos como flujo de tráfico a una conexión a nivel de transporte (p.ej. TCP o UDP)

Una vez obtenidos los flujos entrenamiento, es necesario identificar la aplicación que los ha generado (es decir, la clase a la que pertenecen) para poder utilizar un algoritmo de aprendizaje supervisado. Esta fase se realiza inspeccionando el contenido de los paquetes de forma manual, con la ayuda, por ejemplo, de métodos de reconocimiento por patrones. Este análisis es posible durante la fase de entrenamiento, dado que se realiza de forma offline donde es viable la inspección detallada del contenido de los paquetes. En este trabajo, el conjunto de entrenamiento se obtuvo a partir de trazas de tráfico reales, aunque si esto no fuera posible (p.ej. por problemas de privacidad o por la presencia de tráfico encriptado) este conjunto podría obtenerse igualmente generando tráfico de forma controlada para cada una de las aplicaciones que se desee clasificar.

La fase de entrenamiento finaliza con la generación del árbol de clasificación usando el conjunto de flujos de entrenamiento como entrada del algoritmo C4.5. Para esta fase se utilizó el software de libre distribución Weka [9], desarrollado por la Universidad de Waikato (Nueva Zelanda).

Finalmente, el sistema de monitorización de red utiliza este árbol para identificar en tiempo real las

aplicaciones que han generado el tráfico capturado, sin necesidad de inspeccionar el contenido de los paquetes ni confiar únicamente en la información sobre los números de puerto. Además, una vez entrenado el sistema, el método de identificación es muy eficiente, permitiendo su uso en redes de alta velocidad. El único requerimiento es la extracción en tiempo real de los atributos de cada flujo, que ya fueron seleccionados cuidadosamente de forma que fuesen simples de calcular.

En resumen, nuestro método de identificación de aplicaciones de red basado en el algoritmo supervisado de aprendizaje automático inductivo C4.5 se divide en seis fases principales:

1. Captura de flujos de tráfico reales que sean suficientemente representativos del entorno a monitorizar (o alternativamente generados artificialmente en un entorno controlado).
2. Extracción de los atributos para cada flujo del conjunto entrenamiento.
3. Clasificación manual de cada flujo de entrenamiento a su correspondiente aplicación (este paso puede simplificarse si se opta por utilizar un conjunto de entrenamiento generado artificialmente en el paso 1).
4. Construcción de un árbol de clasificación utilizando el algoritmo C4.5 (p.ej. con el software Weka).
5. Implantación del árbol generado en el paso 4) en el sistema de monitorización pasiva de red.
6. Reentrenamiento del sistema desde el paso 1) cada cierto tiempo, para adaptarlo a cambios en las características del tráfico o a la aparición de nuevas aplicaciones de red.

Una vez entrenado el sistema, el método de identificación es muy eficiente, permitiendo su uso en redes de alta velocidad

4. Resultados

La Figura 1 muestra los resultados obtenidos con una implementación real de este método en el sistema SMARTxAC. En la figura se puede observar la precisión de clasificación obtenida en el enlace que conecta la Anella Científica y RedIRIS para diferentes grupos de aplicaciones (la precisión total media es de un 97.14%). La Figura 2 muestra la diferencia entre los resultados obtenidos con la versión original de SMARTxAC (que utiliza los números de puerto) y el método presentado. Mientras que en la versión original casi el 50% del tráfico no se pudo clasificar (A_UKNWN), utilizando la nueva técnica la mayoría de este tráfico se identificó como P2P.

El único requerimiento es la extracción en tiempo real de los atributos de cada flujo, que ya fueron seleccionados cuidadosamente de forma que fuesen simples de calcular

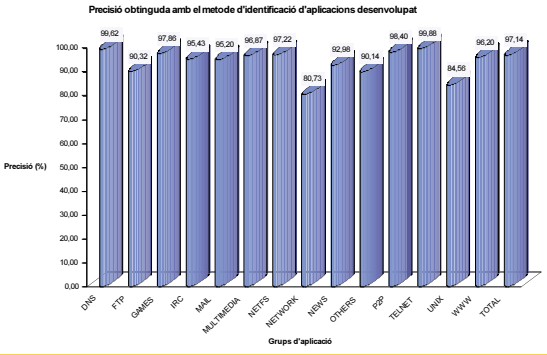

FIGURA 1. PRECISIÓN DE CLASIFICACIÓN POR GRUPO DE APLICACIÓN UTILIZANDO EL MÉTODO PRESENTADO BASADO EN TÉCNICAS DE APRENDIZAJE AUTOMÁTICO.

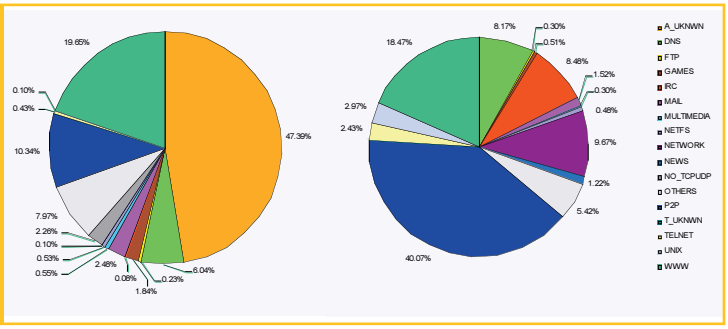

FIGURA 2. PORCENTAJE DE TRÁFICO TOTAL POR GRUPO DE APLICACIÓN UTILIZANDO LOS NÚMEROS DE PUERTO (IZQUIERDA) Y EL MÉTODO PRESENTADO BASADO EN TÉCNICAS DE APRENDIZAJE AUTOMÁTICO.

5. Conclusiones

La identificación de las aplicaciones de red es de especial interés para los operadores y administradores de red. Sin embargo, diversos estudios han demostrado que la clasificación tradicional basada en los números de puertos ya no es efectiva. Aunque recientes investigaciones han propuesto diferentes alternativas al problema, la mayoría requieren el análisis del contenido de los paquetes, y en consecuencia son de difícil aplicación en enlaces de alta velocidad, además de presentar posibles problemas de privacidad.

En este artículo hemos presentado un nuevo método de identificación de aplicaciones de red basado en técnicas de inteligencia artificial que pretende solucionar estas limitaciones. Este método ha sido implantado en el sistema SMARTxAC y evaluado experimentalmente en la Anella Científica. Los primeros resultados son muy prometedores y actualmente se está trabajando en la validación de su eficacia con conjuntos de datos más extensos. También estamos investigando posibles técnicas que permitan automatizar la fase de entrenamiento, que actualmente debe realizarse de forma manual.

Agradecimientos:

Este trabajo ha sido financiado parcialmente por el CESCA (convenio SMARTxAC) y por el Ministerio de Educación y Ciencia (MEC) dentro de los proyectos TSI2005-07520-C03-02 (CEPOS) y TEC2005-08051-C03-01 (CATARO).

Los primeros resultados son muy prometedores y actualmente se está trabajando en la validación de su eficacia con conjuntos de datos más extensos

Este trabajo ha sido financiado parcialmente por el CESCA y por el Ministerio de Educación y Ciencia (MEC)

Referencias

[1] *BARLET-ROS, P; SOLÉ-PARETA, J; BARRANTES, J; CODINA, E; DOMINGO-PASCUAL, J. "SMARTxAC: A passive monitoring and analysis system for high-speed networks". "Campus-Wide Information Systems". 2006. 23(4):283:296.*

[2] *KARAGIANNIS, T; BROIDO, A; FALOUTSOS, M; CLAFFY, K. C. "Transport layer identification of P2P traffic". Presentada en: Internet Measurement Conference. 2004. Taormina, Italia.*

[3] *MOORE, A. W; ZUEV, D. "Internet traffic classification using bayesian analysis techniques". Presentada en: ACM Sigmetrics. 2005. Banff, Canadá.*

[4] *MOORE, A. W; PAPAGIANNAKI, K. "Toward the accurate identification of network applications". Presentada en: Passive and Active Measurement Conference. 2005. Boston, Estados Unidos.*

[5] *KARAGIANNIS, T; PAPAGIANNAKI, K; FALOUTSOS, M. "BLINC: Multilevel traffic classification in the dark". Presentada en: ACM Sigcomm. 2005. Philadelphia, Estados Unidos.*

[6] *L7-FILTER. Application layer packet classifier for Linux. Consultado en: (http:///l7-filter.sourceforge.net) 21-10-2007.*

[7] *QUINLAN, J. R. "C4.5: Programs for Machine Learning". Morgan Kaufmann Publishers. 1993.*

[8] *CODINA, E. "Identificación de aplicaciones de red basada en técnicas heurísticas". Proyecto Final de Carrera. Facultat d'Informàtica de Barcelona. Universitat Politècnica de Catalunya. 2006.*

[9] *WEKA 3. Data mining with open source machine learning software in Java. Consultado en: (http://www.cs.waikato.ac.nz/~ml/weka/) 21-10-2007*

Pere Barlet-Ros
(pbarlet@ac.upc.edu)
Eva Codina
(ecodina@ac.upc.edu)
Josep Solé-Pareta
(pareta@ac.upc.edu)

Centre de Comunicacions Avançades de Banda Ampla (CCABA)
Dept. Arquitectura de Computadors. Universitat Politècnica de Catalunya (UPC)