Entre Ciencia e Ingeniería, ISSN 1909-8367
Año 7 No. 13 - Primer Semestre de 2013, página 61 - 69

Identificación de Virus Informáticos Usando Sistemas Expertos[1]

Identification of Computer Viruses Using Expert Systems

L. Gutiérrez, H. Tabares

Recibido Abril 5 de 2013 – Aceptado Mayo 30 de 2013

Resumen **- Se presenta en este artículo la implementación de un sistema experto para la identificación de virus informáticos. El sistema propuesto modela el conocimiento específico del experto en virus informáticos mediante relaciones entre las variables de entrada (antecedentes) y objetivos (consecuentes). Se espera que el sistema informático desarrollado aumente la protección contra las infecciones software.**

Palabras clave **- Sistema experto, Virus informáticos.**

Abstract - This article discusses the implementation of an expert system for the identification of computer virus. The proposed system models the specific knowledge of computer virus expert through relationships between the input variables and targets. It is expected that the system developed increased software protection against infections.

Key Words — Expert system, Computer Virus.

I. Nomenclatura

SE: Sistema Experto.
BC: Base de Conocimiento.
BD: Bases de Datos.
MI: Motor de Inferencia.
IC: Ingeniero de Conocimiento.
EDC: Experto en el Dominio de Conocimiento.
UF: Usuario Final.

II. Introducción

Un virus informático es un programa de computadora generalmente de pequeño tamaño que se ejecuta en un ordenador huésped y su característica más relevante es que puede replicarse (es decir, hace copias de sí mismo) y propagarse a otras computadoras.

Los efectos secundarios de la actividad vírica son conocidos como problemas informáticos, algunos simplemente irritantes, benignos o genéricos (extraños mensajes de error) y algunos destructivos o malignos (formateo del disco duro, borrar o modificar archivos, desestabilizar el sistema, permitir el acceso remoto desautorizado). Para mayor información, consúltese [1], [2].

Los múltiples problemas producidos por las infecciones de virus informáticos llevó a plantear un trabajo de grado para optar al título en Ingeniería de Sistemas en el que se desarrolló un sistema experto (SE) del tipo determinista titulado ExpertoVIRUS_ITM para la detección y eliminación de este tipo de programas maliciosos. Se descubrió entonces que los SE brindan grandes posibilidades para la protección contra software maligno.

[1] Este trabajo deriva del proyecto de investigación interinstitucional "NUEVAS HERRAMIENTAS PARA REALIZAR PRONOSTICOS DE CONSUMOS DE ENERGIA Y POTENCIA" con código P12208 aprobado según convocatoria interna realizada por el INSTITUTO TECNOLOGICO METROPOLITANO (ITM) cofinanciado con la UNIVERSIDAD DE ANTIOQUIA (U. de A.) según convenio específico celebrado entre las partes No. 8703-002-2012.
L. Gutiérrez, Ingeniera de sistemas, Instituto Tecnológico Metropolitano. ladyjgutierrez@hotmail.com
H. Tabares, M.Sc. en Ingeniería de sistemas, docente ocasional, Instituto Tecnológico Metropolitano. hectortabares@itm.edu.co

Un SE es una aplicación informática que sobre una Base de Conocimientos (BC) posee información de uno o más expertos para solucionar un conjunto de problemas en un área específica. La BC es un tipo especial de Bases de Datos (BD) para la gestión del conocimiento que posee una considerable capacidad de deducción a partir de la información que contiene. La diferencia entre la BD y la BC consiste en que el primero almacena únicamente hechos (afirmaciones que sirven para representar conceptos, datos, objetos, etc.) y las funciones del motor de la BD son las de edición y consulta de los datos. El segundo, por otra parte, puede almacenar, además de hechos (base de hechos que describen un problema), un conjunto de reglas. Una regla es una estructura condicional que relaciona lógicamente la información contenida en la parte del antecedente con otra información contenida en la parte del consecuente. En una BC, las reglas se sirven de esos hechos para que el motor de inferencia (MI), obtenga razonamiento deductivo automático, seleccionando las reglas posibles para solucionar un determinado problema y así conseguir información que no se encuentra almacenada de forma explícita.

Tanto con la BD como con la BC pueden realizar consultas dinámicas. En el primer caso, el usuario final tiene la posibilidad de configurar la consulta en tiempo de ejecución. En el segundo caso, el usuario introduce la información del problema actual en la base de hechos y el sistema empareja esta información con el conocimiento disponible en la base de conocimientos para deducir nuevos hechos [3].

Los SE pueden clasificarse en dos tipos principales según la naturaleza de problemas para los que están diseñados: deterministas y estocásticos. Los SE que tratan problemas deterministas son conocidos como "sistemas basados en reglas", porque sacan sus conclusiones basándose en un conjunto de reglas utilizando un mecanismo de razonamiento lógico. [4]

Los sistemas que tratan problemas estocásticos, es decir, problemas que involucran situaciones inciertas, necesitan introducir medios para medir la incertidumbre. Algunas medidas de incertidumbre son los "factores de certeza" y la "probabilidad". Estos SE utilizan una distribución de probabilidad conjunta de un grupo de variables para describir las relaciones de dependencia entre ellas y así sacar conclusiones usando fórmulas de la teoría de probabilidad.

Para una definición más eficiente de la distribución de probabilidad conjunta se utilizan modelos de redes probabilísticas, entre los que se incluyen las redes de Markov y las redes Bayesianas.

Un detallado estudio sobre SE está más allá del ámbito de esta sección. En [5] se ofrece un excelente estudio con referencias específicas.

III. Desarrollo del artículo

Antecedentes

La literatura presenta numerosos estudios de aplicación usando inteligencia artificial para la determinación de virus informáticos.

En razón de que un virus informático posee un conjunto de atributos que describe su actividad y comportamiento, en [6] se diseñó una red neuronal artificial del tipo perceptrón multicapa como un componente de seguridad para el reconocimiento y la clasificación de ataques de virus informáticos. Se demostró que las RNA son prácticos dispositivos eficaces para discriminar patrones de entrada maliciosos.

Una amenaza seria a la seguridad de la información residente en un computador son los correos electrónicos maliciosos que suelen tener archivos adjuntos infectados, ocasionando graves problemas como intrusiones no autorizadas, ataques de denegación del servicio y virus informáticos. En [7] se propone un método de clasificación para la detección automática de virus en mensajes de correo electrónico malintencionados. Consiste en utilizar redes de probabilidad bayesiana y árboles de decisión, implementadas en un sistema experto (Antimailvirus), que actúan como filtros de archivos adjuntos maliciosos de Windows.

En [8] emplean un modelo dinámico para la detección de virus informáticos. Consiste en utilizar el clásico modelo epidemiológico SAIC (Susceptible, Antídotos, Infecciosas, Contaminada) para evaluar la propagación de enfermedades, adaptado a los efectos que producen los virus informáticos en una computadora mediante el uso de sencillas técnicas de identificación de síntomas de infección. Para la validación del modelo se usaron datos reales de síntomas de infección que producen los virus informáticos.

El uso creciente de los sistemas de computadoras ha aumentado el problema de accesos no autorizados, la manipulación de datos y la ocurrencia de malas prácticas, como en el caso de los usuarios autorizados que intentan sobrepasar sus límites de restricción de acceso a la información. Por lo tanto, la detección de intrusiones está despertando considerable interés en la comunidad investigadora. Nuevas metodologías computacionales se están utilizando para tal fin, como es el caso de la inteligencia computacional. Su característica principal está relacionada con la adaptación, tolerancia a fallos, alta velocidad computacional y la capacidad de generalización ante información ruidosa o incompleta, la hace una herramienta ideal para la construcción de un modelo de detección de intrusos. En [9] se presenta una visión general del progreso de las investigaciones en la aplicación de métodos para la detección de intrusos, incluyendo redes neuronales artificiales, sistemas difusos, computación evolutiva, sistemas inmunológicos artificiales, inteligencia de enjambre

y Soft Computing. Las contribuciones de investigación en cada campo se resumen y se comparan de forma sistemática, lo que permite definir con claridad los retos actuales y pone de relieve nuevas líneas de investigaciones.

La prevalencia creciente de intrusiones en la red es un problema bien conocido que puede afectar la disponibilidad, confidencialidad e integridad de la información crítica para los individuos y las empresas. En [10] se propone un enfoque para detectar intrusiones en tiempo real utilizando máquinas de aprendizaje supervisado. El método es simple y eficiente, utiliza la técnica de árbol de decisión para clasificar los datos en línea de la red, como normales o de ataque. Se identificaron 12 características esenciales de la red de datos que son relevantes para la detección de ataques, distinguiendo entre actividades normales de red de los tipos principales de ataque (intrusiones no autorizadas, ataques de denegación del servicio y virus informáticos). También se desarrolló un procedimiento de post-procesamiento para reducir la tasa de falsas alarmas, así como aumentar la precisión y fiabilidad de detección del sistema de detección de intrusos.

Hay dos métodos de detección de intrusos: uso indebido y basado en anomalías. En [11] se propone incluir ambos métodos para su detección, ya que el rendimiento de un motor de detección individual es raramente satisfactorio. En el mismo artículo se sugiere la lógica difusa, el soft computing y otras técnicas de Inteligencia artificial, para reducir la tasa de falsas alarmas mientras se mantiene la alta tasa de detección.

Técnicas más complejas para la detección de intrusos han requerido la utilización de sistemas expertos debido a que los escenarios de intrusiones se pueden identificar con un conjunto de reglas que reflejan la secuencia parcial y ordenada de acciones de la intrusión. El estado del sistema se representa como una base de conocimiento consistente de una base de reglas y una base de hechos. Una base de conocimiento es una colección de aserciones que pueden ser hechas basándose en datos acumulados de los registros de auditoría o directamente de la observación de la actividad de un sistema. La base de reglas contiene las reglas que describen los escenarios de intrusión conocidos o técnicas genéricas. Esto puede causar la emisión de una alerta para el administrador del sistema. En la literatura se encuentra referencias de diferentes sistemas expertos creados para la detección de intrusos (MIDAS [12], IDES [13], HayStack [14]). El más reciente, IIDS [15] (Detección de intrusos inteligente) es un sistema para evaluar la autenticación a través de una red UNIX, basada en reglas específicas, errores o actividad maliciosa. El módulo de mapa cognitivo (FCM, Fuzzy cognitive Maps), núcleo del IIDS, provee una forma natural de adquisición de conocimiento, que representa el conocimiento de un experto de manera tal que es muy fácil de entender por un experto humano.

En la arquitectura IIDS, los sensores de detección de anomalías y formas no autorizadas de uso que están permanentemente monitoreando el sistema, sirven como expertos, en las estaciones de trabajo de usuarios finales, y de tráfico de la red.

Finalmente en [16] se propone un sistema de detección de intrusos usando agentes inmunes artificiales basados en una adaptación del sistema inmunológico humano. El sistema inmunológico está siempre alerta para detectar y atacar agentes infecciosos. Sea cual fuere el agente, el sistema inmunológico lo reconoce como un cuerpo ajeno, antígeno y libera un grupo de células llamadas macrófagos. Enzimas en el interior del macrófago destruyen al antígeno procesándolo en péptidos antigénicos. En el último paso de este proceso, una célula llamada fagocito se encarga de remover el antígeno del cuerpo.

El sistema inmunológico humano es un sistema distribuido, robusto, dinámico, diverso y adaptable, lo cual lo convierte en una excelente base para la protección de sistemas computacionales. Ou Ch-M utilizó los conceptos sobre inmunología previamente explicados para crear un modelo de detección de intrusos, reconociendo dinámicamente patrones de ataques de agentes patógenos, en los llamados al sistema de privilegios. Para detectar los patrones de ataques típicos o las tentativas de explotación de vulnerabilidades conocidas, el modelo monitoriza las actividades que ocurren en el sistema y las compara con una serie de firmas de ataques previamente almacenadas en una base de datos. Cuando se monitorizan actividades que coinciden con las firmas se genera una alarma. Este tipo de análisis se atiene al conocimiento previo de las secuencias y actividades que forman un ataque. El sistema presenta desventajas como tener que conocer a priori el patrón de ataque, lo que provoca que nuevas intrusiones pasen desapercibidas por el detector, o la facilidad con la que se podría engañar al sistema con pequeñas variaciones de los patrones de ataque conocidos.

Hasta el momento se conocen más de 60.000 agentes infecciosos y variantes, circulando en su mayor parte a través de Internet y en los que se suelen aplicar técnicas de ingeniería social con el fin de vencer las reticencias de las personas más precavidas. En cambio, poco menos de la mitad de los sistemas se encuentran protegidos mediante un antivirus con definiciones de virus actualizadas. La consecuencia inmediata es que alrededor del 70% de los ordenadores instalados en el planeta sufrieron en el año 2012 algún tipo de incidente por ataque vírico. Es por eso que en el INSTITUTO TECNOLOGICO METROPOLITANO se ha venido trabajando modelos computacionales usando SE como el expuesto en [17]. Para los propósitos de este trabajo investigativo, el artículo antes referido fue tomado y adaptado para detección de virus informáticos.

Implementación software del sistema experto

Los actores que intervienen en el aplicativo ExpertoVIRUS_ITM son: el ingeniero de sistemas que

en este caso hace las veces de ingeniero del conocimiento (IC), el experto en el dominio de conocimiento sobre virus informáticos (EDC) y los usuarios finales (UF) que interactúan con el sistema con miras a determinar si un computador se encuentra infectado por la presencia de un virus informático.

La metodología abordada para la implementación del SE contempló las siguientes etapas:

a. Identificación del problema, en este caso, la determinación del tipo de virus informático que puede estar infectando un computador.
b. Selección de la variable objetivo y sus valores. El EDC definió como variable objetivo identificar los diferentes tipos de virus informáticos. Los valores posibles que puede asumir la variable son: Gusano, Troyano, Camaleón, Bombas de tiempo y lógicas, reproductores y los falsos.
c. La selección de las variables de entrada y sus posibles valores. Sobre la base del conocimiento del EDC, se seleccionaron el conjunto de variables de entrada relevantes. En el modelo abordado en este trabajo las preferencias están determinadas por los indicadores básicos de síntomas de infección de virus informáticos que se detallan en la tabla 1. Su clasificación es muy variada, por lo que para los propósitos de este artículo, se listan los más destacados según sus efectos a nivel del hardware o software.

TABLA I
SINTOMAS DE INFECCION A NIVEL DE HARDWARE Y SOFTWARE

Síntoma en Hardware
Dificultad para arrancar el PC.
Ralentización en la velocidad de ejecución de los programas.
El PC se reinicia frecuentemente.
El PC no reconoce el disco duro.
El PC pide palabras claves, “passwords”, no configuradas por el usuario.
El sistema operativo del PC presenta mensajes de error inesperados.
El sistema operativo deniega el acceso a discos duros locales o unidades de almacenamiento extraíble.
Es imposible imprimir documentos de forma correcta.
Reducción del tamaño de la memoria RAM[2].
Los mapas de memoria revelan nuevos programas residentes en memoria de origen desconocido.
El disco duro aparece con sectores en mal estado[3].
El número de sectores dañados de disco aumenta constantemente.
Reducción del espacio disponible del disco[4].
Bloqueo o aparición de anomalías en el teclado[5].
Aparición de anomalías en el video[6].
Los programas dirigen los accesos a los discos en tiempos inusuales o con una frecuencia mayor.
Síntoma software
Aparición de menús y cuadros de diálogos distorsionados.
Archivos que se ejecutan mal.
Programas que funcionan de modo anormal o se cierran inesperadamente.
Programas de juegos que se materializan misteriosamente sin haber sido previamente instalados[7].
Se borran archivos inexplicablemente haciendo imposible su recuperación.
Aparecen archivos de datos o directorios de origen desconocido.
Los archivos son sustituidos por objetos de origen desconocido.
Nombres, extensiones, fechas, atributos o datos cambian en archivos o directorios que no han sido modificados por los usuarios.
Cambios en las características de los archivos ejecutables[8].

[2] Un virus, cuando entra en una computadora, debe situarse obligatoriamente en la memoria RAM y por ello ocupa una porción de ella. Por tanto, el tamaño útil operativo de la memoria se reduce en la misma cuantía que tiene el código del virus.

[3] Algunos virus usan sectores del disco para camuflarse, lo que hace que aparezcan como dañados o inoperativos.

[4] Ya que los virus se van duplicando de manera continua, es normal pensar que esta acción se lleve a cabo sobre archivos del disco, lo que lleva a una disminución del espacio disponible por el usuario

[5] Existen algunos virus que definen ciertas teclas, las cuales al ser pulsadas, realizan acciones perniciosas en la computadora

[6] Muchos de los virus eligen el sistema de video para notificar al usuario su presencia en la computadora. Cualquier desajuste de la pantalla o de los caracteres de ésta, nos puede notificar la presencia de un virus.

[7] Por ejemplo, agujeros negros, pelotas que rebotan, caras sonrientes o caracteres alfabéticos «lluviosos» empiezan a aparecer en la pantalla.

[8] Casi todos los virus, aumentan el tamaño de un archivo ejecutable cuando lo infectan. También puede pasar, que cambien la fecha del archivo a la fecha de infección.

Si se activa cualquiera de los síntomas a nivel del hardware y/o del software, entonces se infiere que hay en el sistema un virus informático.

Los síntomas listados no constituyen la universalidad de casos, por lo que el programa ExpertoVIRUS_ITM provee los medios para la recolección de nuevos síntomas.

Es importante recalcar que los síntomas antes listados no siempre son indicios de la presencia de virus y malware en el sistema, dado que pueden deberse de igual forma a problemas con los componentes de hardware o su incorrecta configuración. Así mismo, presentar uno o dos de estos síntomas de forma aislada tampoco significa necesariamente la presencia de alguna amenaza al sistema. Por lo general para considerar una infección, se requieren tres o más de dichos síntomas en forma conjunta.

d. Diseño de las preguntas. El desarrollo del cuestionario es la parte más complicada y en general es una actividad iterativa. El proceso se puede comenzar identificando los rasgos y características representadas en las variables de entrada, presentes o no en los valores que asume la variable objetivo identificada para el dominio del problema. En esta fase intervinieron el IC y el EDC. La adquisición de la información relevante se obtuvo consultando con el EDC y el IC extrajo las principales características de los diferentes tipos de virus informáticos. Con base en los rasgos identificados en el paso anterior, el IC y el EDC diseñan las reglas a evaluar y las preguntas a formular a los UF del sistema cuando interactúa con el subsistema de interfaz de usuario final. En la tabla 2 se presentan las reglas a evaluar.

El SE intercambia datos entre las tablas 1 y 2 para inferir el tipo de virus que está siendo analizado. La clasificación de los virus es muy variada, agrupándose por la entidad que parasitan –sector de arranque o archivos ejecutables-, por su grado de dispersión, por su comportamiento, por su agresividad, por sus técnicas de ataque, por la forma en que se ocultan. Aunque en la actualidad casi todos los virus tienen comportamientos complejos e incorporan características de varias clases, se podrían diferenciar los tipos de virus basándose en el daño que causan y efectos que provocan.

Los virus tienen diferentes finalidades: algunos sólo "infectan", otros alteran datos, otros los eliminan y algunos sólo muestran mensajes. Dependiendo del tipo de virus, del lugar donde se alojan, la técnica de replicación o la plataforma en la cual trabajan, el proceso de infección varia sensiblemente. Por lo tanto, la tabla 2 no presenta una clasificación completa de virus informáticos, sino que existen tantas clasificaciones como expertos en el tema, siendo todas válidas y muchas veces complementarias unas de otras. De lo anterior se deduce que la BC del programa ExpertoVirus_ ITM, permite registrar nuevos tipos de clasificaciones.

Tabla II
SUBCLASES DE VIRUS

No	Regla antecedente (Daños o efectos)	Consecuente (Tipo de virus)
1	El virus se registra para correr cuando inicia el sistema operativo ocupando la memoria y volviendo lento al ordenador, pero no se adhieren a otros archivos ejecutables. Utiliza medios masivos como el correo electrónico para esparcirse de manera global.	Gusano[9]
2	Programa informático que se hace pasar como programa genuino (salvapantallas, juegos, música).	Troyano[10]
3	Programa que se disfraza de software conocido.	Camaleón[11]
4	Programa creado para activar el trigger de su módulo de ataque en una fecha determinada o luego de un determinado número de ejecuciones del programa.	Bomba de tiempo
5	Programa que actúa si se dan ciertas condiciones lógicas.	Bomba lógica
6	No atacan los archivos. Son programas autónomos cuyo propósito es agotar los recursos del sistema al reproducirse continuamente mientras el disco y la memoria tengan espacio disponible.	Reproductores
7	Los virus falsos son simplemente mensajes que circulan por e-mail que advierten de algún virus inexistente[12].	Falsos

[9] Se aprovechan de las redes de computadoras para propagarse de manera automática, es decir, sin la intervención del usuario, usando el correo electrónico, con habilidad para replicarse en grandes números lo que provoca un efecto dominó de intenso tráfico de red que puede hacer más lenta las redes empresariales e Internet en su totalidad.

[10] Permanecen en el sistema no ocasionando acciones destructivas sino que actúan como espía, capturando passwords y datos confidenciales, renviándolos a una dirección externa. Debido a esta particular característica, son muy utilizados por los ciberdelincuentes para, por ejemplo, robar datos bancarios.

[11] Un software camaleón podría por ejemplo, emular un programa de acceso a sistemas remotos (rlogin, telnet) realizando todas sus acciones, pero como tarea adicional (y oculta a los usuarios) va almacenando en algún archivo los diferentes logins y passwords para que posteriormente puedan ser recuperados y utilizados ilegalmente por el creador del virus camaleón. Este virus cada vez que contamina un computador tienen una mutación para evitar ser detectado.

[12] Estos virus falsos no afectan el sistema, solo son advertencias que se multiplican y mandan por internet con una gran velocidad. No tienen ningún código oculto ni instrucciones para ejecutar. Funcionan de manera muy sencilla: un usuario recibe un email con la advertencia de algún virus raro, estos usuarios lo reenvían para advertir, entonces se genera un tráfico de email sobre una amenaza inexistente.

e. Implementación software. En las especificaciones de diseño del aplicativo ExpertoVIRUS_ITM se manejaron tres capas: interfaz, lógica de programa, datos; esto para lograr que la interfaz sea completamente independiente del sistema y la BC pueda alterarse sin tener que hacer cambios en la programación. La BC reside en un manejador de bases de datos. La base de reglas y el MI se implementaron en lenguaje C# para ambiente de escritorio, utilizando el escenario de desconectado para tener acceso al motor de la BD.

La arquitectura básica del aplicativo ExpertoVIRUS_ ITM se puede observar en la Fig. 1.

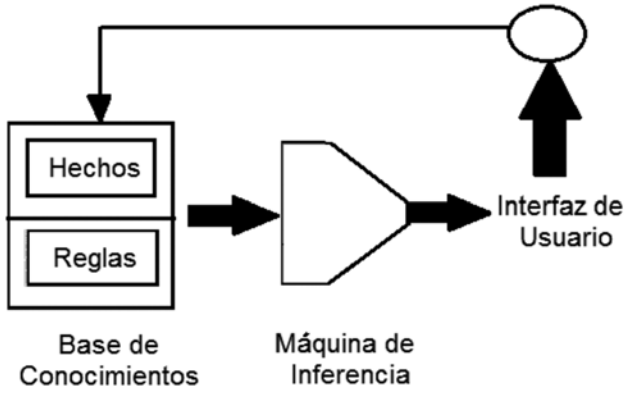

Fig. 1 Arquitectura del aplicativo ExpertoVIRUS_ITM.

El programa ExpertoVIRUS_ITM está compuesto por tres subsistemas: i) el editor de variables de entrada y salida, ii) la base de conocimiento, iii) la interfaz de usuario final.

Editor de variables de entrada y salida.

Con base en los rasgos identificados en las variables de entrada, el IC y el EDC las registran utilizando el subsistema editor de variables de entrada (antecedentes) y salida (consecuente). En la Fig. 2, se ilustran algunas formulaciones que estarán disponibles.

El control combo titulado "Tipo Variable" permite ingresar una nueva variable del tipo antecedente o consecuente.

Editor base de conocimiento

El archivo de las variables de entrada y salida se carga en el subsistema de adquisición del conocimiento. Inicialmente el sistema no contiene posibles reglas. Como se presenta en la Fig. 3, el EDC carga las variables generadas por el IC y configura las reglas en la base de conocimiento. Este proceso se repite hasta que se hayan configurado todas las soluciones posibles para cada una de las preguntas formuladas.

Es frecuente que diferentes sucesos estén correlacionados, siendo necesario realizar cierta pregunta si la respuesta seleccionada a la pregunta anterior lo requiere. El subsistema permite establecer dependencias por medio de su opción "Crear Dependencia".

Finalizado este proceso, las preguntas formuladas, las dependencias y la importancia asignada son guardadas en una base de datos.

Fig. 2 Editor de variables de entrada y salida. El IC implementa el editor y el EDC lo diligencia en tiempo de ejecución.

Fig. 3 Editor adquisición del conocimiento. Diligenciado en tiempo de ejecución por el EDC.

Interfaz de Usuario Final

La Fig. 5, ilustra el subsistema de interacción con el UF. El subsistema de interacción con el UF permite el acceso al conocimiento especializado que puede consultarse interactivamente en cada paso, luego de seleccionar la variable de entrada y su valor en el recorrido desde el inicio a la meta.

El cuestionario primero se presenta al EDC cuando utiliza el subsistema de adquisición del conocimiento para completar la BC y verificar cómo están relacionadas cada una de las opciones disponibles para recomendar una solución.

El funcionamiento general del subsistema de interacción con el UF es el siguiente: el UF selecciona el conjunto de variables evidenciables relevantes sobre síntomas de infección de virus informáticos y responde las preguntas planteadas por el SE. En el modelo abordado en este trabajo, las preferencias están determinadas por las variables: i) síntomas de infección, ii) clasificación del virus informático.

La interfaz que maneja la interacción entre el SE y el UF es altamente interactiva y sigue el patrón de la conversación a nivel de escritura entre seres humanos. Por lo tanto, un requerimiento básico de la interfaz es la habilidad de hacer preguntas. Para obtener información fiable del usuario, es necesario prestar especial cuidado en el diseño del cuestionario. Esto requirió diseñar el interfaz usando menús o gráficos.

La consulta transcurre en general según el esquema siguiente: Primero se plantea al usuario la pregunta sobre los síntomas de infección para alcanzar una determinación aproximada del contexto. El dialogo, por parte del sistema, está a menudo dimensionado para ir confirmando o rechazando hipótesis (por ejemplo, comprobación de la presencia de un virus informático), o para realizar una aproximación sucesiva hacia un objetivo introducido de antemano (por ejemplo, subclases de virus informáticos).

Una vez finalizado el diálogo, el componente explicativo suministra, si es necesario, el historial completo de la consulta.

La configuración realizada por el UF es empleada por el MI para obtener razonamiento deductivo automático, seleccionando las reglas posibles para especificar el tipo de virus que está infectando un computador.

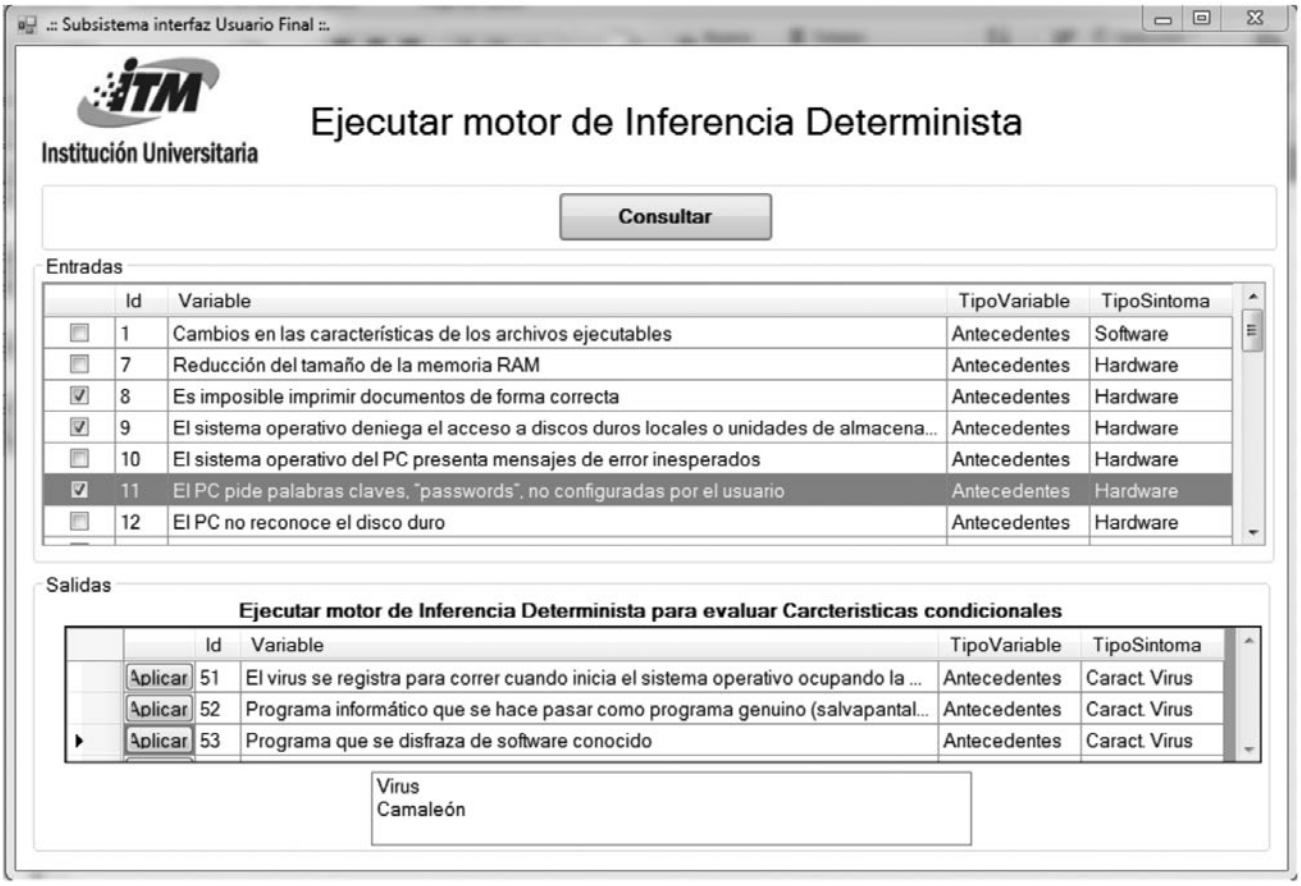

Fig. 5. Interfaz UF

Resultados

Para probar el sistema, se empleó el subsistema de interfaz de interacción con el UF. Con los datos definidos en los filtros, el sistema ExpertoVIRUS_ITM simula al experto humano usando el MI, aplicando un encadenamiento de reglas hacia adelante para contrastar los hechos particulares de la base de hechos con el conocimiento contenido en la BC y obtener conclusiones acerca de la consulta realizada. El algoritmo consiste en asignar a los objetos sus valores conocidos tales como lo dan los hechos o evidencias. Seguidamente se ejecuta cada regla de la base de conocimiento y se concluye nuevos hechos si es posible. Finalmente, se repite el paso anterior hasta que no puedan ser obtenidos nuevos hechos. Para la evaluación de cada regla, el MI la convierte de notación infijo a posfijo. Seguidamente, el MI aplica una subrutina evaluadora de expresiones en notación posfijo. En la tabla 3, se presenta la inferencia del SE ante 4 posibles síntomas de infección de virus.

Tabla III
Operación Del Se.

SISTEMA EXPERTO VIRUS				
Variables de entrada: Síntomas de infección		**Variables de salida activadas por el motor de inferencia**		
Hardware	Software	Virus	Regla	Tipo
x		x	1	Gusano
	x	x	2	Troyan
x	x	x	5	Bomba lógica
x	x	x	7	Falsos

Como se observa, el SE determina exitosamente el tipo de virus en función de las variables de entrada seleccionadas.

Para lograr un mejor rendimiento del MI del sistema informático, ExpertoVIRUS_ITM maneja la transaccionalidad de la base de datos y su procesamiento de manera local, usando el escenario de desconectado, empleando los recursos de la máquina en vez del servidor designado.

IV. Conclusiones

El objetivo de este trabajo consistió en desarrollar el prototipo software ExpertoVIRUS_ITM para la detección de virus informáticos.

El sistema propuesto modela el conocimiento específico del experto en virus informáticos, mediante relaciones entre conceptos explicitados en las variables de entrada (síntomas de infección) y la variable de salida u objetivo (identificar el tipo de virus informático).

El prototipo de SE permite obtener información sobre el tipo de virus de acuerdo a los indicadores de infección a nivel de software o hardware.

ExpertoVIRUS_ITM se encuentra registrado en Colombia, Dirección Nacional de Derecho de Autor, con número 13-33-469 del 26 de Junio de 2012.

El prototipo de sistema desarrollado está accesible en la dirección: http://www.itm.edu.co/trabajosdegradodestacados/index.html.

Es de interés realizar otros estudios en el que se usen SE en problemas que requieran inferencia determinista para realizar clasificaciones de antivirus, intrusiones y delitos informáticos.

Agradecimientos

Esta sección reconoce la ayuda por su asistencia al señor profesor Jorge Iván Bedoya Restrepo, docente de la asignatura Bases de Datos.

Referencias

[1] A. Nikishin, ***"Malicious software – past, present and future"***, Vol. 2, p.p. 6-18, 2004. Disponible en http://www.sciencedirect.com/science/article/pii/S1363412704000202. Fecha de consulta 19 de Junio de 2012.

[2] C. Zhongqiang, R. Mema, L. Zhanyan, Z. Yuan, C. Zhongrong, A. Delisb, "***Malware characteristics and threats on the internet ecosystem"***, The Journal of Systems and Software, *ISBN 0-8186-2420-5. Vol. 85. p.p 1650-1672, 2012.*

[3] J. Giarratano, G. Riley. ***"Sistemas expertos. Principios y programación"***. International Thomson Editors, 2001.

[4] S. *Sahin, R. Tolun, "Expert System with applications. International Journal", ELSEVIER, Volumen 39, edición 4, p.p. 4609-4617, 2012.*

[5] J. Liebowitz. "***Expert Systems with applications. An International Journal"***. ELSEVIER, Vol. 40, edición 15, 2013.

[6] A. Doumas, K. Mavroudakis, D. Gritzalis, S. Katsikas, ***"Design of a neural network for recognition and classification of computer viruses"***, Computers & Security, ELSEVIER, Vol. 14, p.p. 435-448, 1995.

[7] S. Dong-Her, C. Hsiu-Sen, Y. David, ***"Classification methods in the detection of new malicious emails"***. Information Sciences. ELSEVIER, Vol. 172, p.p. 241-261, 2005.

[8] J. Piqueira, A. Vasconcelos, V. Araujo, ***"Dynamic models for computer viruses"***, Computers & Security, ELSEVIER, Vol. 27, p.p. 355-359, 2008.

[9] S. Wu, W. Banzhaf, ***"The use of computational intelligence in intrusion detection systems: A review"***. Applied Soft Computing. ELSEVIER. Vol. 10, p.p. 1-35, 2010.

[10] P. Sangkatsanee, N. Wattanapongsakorn, C. Charnsripinyo, "***Practical real-time intrusion detection using machine learning approaches"***. Computer Communications, ELSEVIER, Vol. 34, p.p. 2227-2235, 2011.

[11] H. Elshoush, I. Osman, ***"Alert correlation in collaborative intelligent intrusion detection systems—A survey"***, Applied Soft Computing, ELSEVIER, Vol. 11, p.p. 4349-4365, 2011.

[12] M. Sebring, E. Shellhouse, M. Hanna, R. Whitehurst, ***"Expert systems in intrusion detection: A case study"***. Proceedings of the 11th National Computer Security Conference. Octubre 1988.

[13] T. Lunt, R. Jagannathan, R. Lee, A. Whitehurst, S. Listgarten, ***"Knowledge-based intrusion detection"***. Proceedings of the Annual AI Systems in Government Conference, p.p. 102-107. 1989.

[14] S. Smaha, ***"HayStack. An intrusion detection system"***. Disponible en http://people.scs.carleton.ca/~soma/id/readings/smaha-haystack.pdf. Fecha de consulta, 21 de Junio de 2012.

[15] A. Ajith, T. Johnson, ***"Distributed intrusion detection systems: a computational intelligence approach"***. Disponible en http://wsc10.softcomputing.net/hussein_chapter.pdf. Fecha de consulta, 21 de Junio de 2012.

[16] Ou Ch-M, ***"Host-based intrusion detection systems adapted from agent-based artificial immune systems"***, Neurocomputing, ELSEVIER, Vol. 88, p.p. 78-86, 2012.

[17] C. Primorac, S. Mariño, ***"Un sistema experto para asistir decisiones turísticas. Diseño de un prototipo basado en la web"***. Revista de investigación en turismo y desarrollo local. vol. 4, No 10. Disponible en http://www.eumed.net/rev/turydes/10/index.htm. Fecha de consulta, 9 de Mayo de 2012.

Héctor Tabares Ospina es Magister en Ingeniería de Sistemas, Especialista en Ingeniería del Software e Ingeniero electricista, integrante del grupo de Investigación Automática, Electrónica y Ciencias Computacionales, Docente Instituto Tecnológico Metropolitano, h.tabares@pascualbravo.edu.co

Lady Gutiérrez Benjumea es estudiante Ingeniería de Sistemas, integrante del grupo de Investigación Automática, Electrónica y Ciencias Computacionales del Instituto Tecnológico Metropolitano, ladyjgutierrez@hotmail.com