Evaluación de Ciberseguridad: Análisis de Riesgos a Sistemas de Ciberseguridad Convencionales

Cybersecurity assessment: Risk analysis to conventional cybersecurity systems

Este artículo corresponde a una investigación en proceso y que es parte de la Tesis de Maestría en Ingeniería en Sistemas de Información

Eduardo A. Leiva[1], Hernán D. Merlino[2]

[1] Programa de Maestría de Ingeniería en Sistemas de Información. Escuela de Posgrados - Universidad Tecnológica Nacional (UTN) - Facultad Regional de Buenos Aires – Argentina

[2] Laboratorio de Investigación y Desarrollo en Sistemas de Inteligencia Artificial (LIDSIA) Universidad de Lanús - Buenos Aires – Argentina

eduaraleiva@hotmail.com.ar, hmerlino@gmail.com

Abstract. *This article aims to provide a model for assessing cybersecurity vulnerabilities and the countermeasures implemented by a medium-sized organization, based on the proposed Conventional Cybersecurity Systems architecture. The main beneficiaries of the use of this vulnerability assessment model based on risk analysis are the decision makers who have the responsibility or interest in maintaining cybersecurity. Therefore, it is considered that this proposal will help to assess the vulnerabilities and the countermeasures implemented in order to obtain a status of the situation, and from it, adopt the necessary measures to improve the unfavorable aspects.*

Resumen. *En este artículo se pretende proporcionar un modelo para evaluar las vulnerabilidades de ciberseguridad y las contramedidas implementadas una organización de mediana envergadura, basándose en la arquitectura de Sistemas de Ciberseguridad Convencionales propuesta. Los beneficiarios principales de la utilización de este modelo de evaluación de vulnerabilidades basada en el análisis de riesgos, son los tomadores de decisiones que tienen la responsabilidad o el interés en mantener la ciberseguridad. Por lo tanto, se considera que esta propuesta ayudará a evaluar las vulnerabilidades y las contramedidas implementadas a fin de obtener un estado de la situación, y a partir de la misma, adoptar las medidas necesarias para mejorar los aspectos desfavorables.*

1. Introducción

En la actualidad el uso de las Tecnologías de Información y de Comunicación se ha incorporado de forma general a la vida cotidiana. Este nuevo escenario facilita un desarrollo sin precedentes en el intercambio de información y comunicaciones, pero al mismo tiempo conlleva serios riesgos y amenazas que pueden afectar la seguridad de los sistemas de información.

Varios son los factores que contribuyen a la proliferación de acciones delictivas en el ciberespacio, como, por ejemplo, la rentabilidad que ofrece su explotación en términos económicos, políticos o de otro tipo, la facilidad y el bajo costo de las herramientas utilizadas para la consecución de ataques y la facilidad de ocultación del atacante, hacen posible que estas actividades se lleven a cabo de forma anónima, desde cualquier lugar del mundo y con impunidad. Esto tiene un impacto notable sobre las distintas organizaciones en los sectores público y privado, y los propios ciudadanos. Los distintos perfiles de atacantes explotan las vulnerabilidades tecnológicas con el objeto de recabar información de valor para cometer ilícitos, como así también para amenazar los servicios básicos que pueden afectar al normal funcionamiento de un país.

Desde el auge de ciberdelincuencia, los actores involucrados como los terroristas, el estado, el ciberespionaje corporativo y los activistas cibernéticos convierten al ciberespacio en el quinto elemento de la guerra (junto con la tierra, el mar, el aire y el espacio) la cuestión de la ciberseguridad se ha vuelto un importante problema a nivel mundial [Pakalniškis, S., 2012].

De hecho, algunos aportes sobre el tema de la ciberseguridad han declarado que la carrera armamentista cibernética ha comenzado. Puede haber buenas razones para tales declaraciones, ya que muchas naciones han declarado en sus estratégicas de ciberseguridad la importancia de desarrollar capacidades cibernéticas ofensivas, entre ellos están algunas de las principales potencias mundiales, como Estados Unidos, Rusia y China [Pakalniškis, S., 2012].

Por ejemplo EE.UU. ha establecido su Comando Cibernético con el objetivo de proteger sus redes militares y de defensa a través de un continuo bombardeo de ataques cibernéticos [Pakalniškis, S., 2012].

Dentro de este contexto global, las organizaciones y las empresas juegan un papel importante, ya que deben tener sus propias políticas de ciberseguridad para resguardar sus sistemas de información.

Se puede observar que la ciberseguridad ya no es un problema de seguridad informática solamente, se debe ver a la ciberseguridad como una cuestión que afecta a todos los niveles de la sociedad, las organizaciones públicas y privadas y las empresas, ya que el uso ilícito del ciberespacio podría tener impactos negativos en el bienestar económico, político y social.

Es útil apreciar cómo la ciberseguridad ha evolucionado, la evolución de Internet permitió una mayor productividad al permitir procesar negocios mediante las transacciones en línea. Esta capacidad se conoce como el comercio electrónico (e-commerce). Desde el siglo XX la economía se volvió tan dependiente del e-commerce que es un objetivo frecuente de los delincuentes cibernéticos, por otro lado, simultáneamente la tecnología

de seguridad evoluciona constantemente para proteger los datos que se podrían utilizar para cometer fraudes. Esta tecnología se conoce generalmente como contramedidas, porque son las medidas de seguridad diseñadas para contrarrestar una amenaza específica [Bayuk, 2012].

Se dedicó un esfuerzo considerable de investigadores y profesionales para preservar los sistemas de TI en un estado confiable y predecible. Esto sin embargo es un tema difícil de manejar, ya que una arquitectura de TI moderna se compone de un gran número de sistemas, procesos e individuos conectados. A esto último hay que agregarle que las amenazas hacia los sistemas surgen de errores durante el desarrollo y el mantenimiento de los sistemas de TI [Holm H. et al., 2013].

La presencia de individuos decididos a explotar estos errores añade otra capa de complejidad al problema. Para estimar la vulnerabilidad de un sistema, es necesario considerar una enorme cantidad de factores. No es suficiente con considerar todas las vulnerabilidades, también hay una necesidad de comprender cómo estas vulnerabilidades se relacionan entre sí y como se relacionan con las contramedidas de seguridad adoptadas. En consecuencia, es una tarea difícil para los tomadores de decisiones de una organización gestionar la ciberseguridad de sus sistemas. Un medio común para evaluación de la ciberseguridad es consultar a expertos, por ejemplo, de ethical hacking, informática forense y seguridad informática. Aunque las evaluaciones de los expertos sin duda son valiosas, tienen tres limitaciones importantes:

a. Solo son válidas para el tiempo en que se llevaron a cabo.
b. Solo son válidas para las partes de la arquitectura de la empresa que fueron estudiadas por el experto.
c. Solo son válidas para las evaluaciones efectuadas y los resultados obtenidos, de acuerdo a la experiencia profesional del experto.

Estas limitaciones son problemáticas dada la naturaleza dinámica de los sistemas de TI y la falta de recursos disponibles para el análisis. Por lo tanto, los tomadores de decisiones de una empresa se ven en la necesidad de contar con herramientas que puedan ayudar a realizar la evaluación de ciberseguridad de sus sistemas de una manera fácil. Aunque existen varias herramientas disponibles para este propósito, no se adaptan a los objetivos de este trabajo, se pueden mencionar COMMON CRITERIA, CORAS, CRAAM, ISRAM [Holm H. et al., 2013].

En este artículo se proporciona a los responsables de la seguridad informática de una organización, un modelo para evaluar las vulnerabilidades de ciberseguridad y las contramedidas implementadas, basándose en la arquitectura de Sistemas de Ciberseguridad Convencionales propuesta. Los beneficiarios principales de la utilización de este modelo de evaluación de vulnerabilidades basada en el análisis de riesgos, son los tomadores de decisiones que tienen la responsabilidad o el interés en mantener la ciberseguridad en las organizaciones, inevitablemente este público es amplio, debido a que la ciberseguridad toca prácticamente todas las formas de actividad social y económica. Por lo tanto, se considera que esta propuesta ayudará a cualquier organización con una arquitectura de Sistema de Ciberseguridad Convencional a evaluar sus sistemas de ciberseguridad, es decir evaluar las vulnerabilidades y las contramedidas implementadas a fin de obtener un estado de la situación y a partir de la misma, adoptar las medidas necesarias para mejorar los aspectos desfavorables. Esto se traduce en

mejorar la resistencia a los ataques cibernéticos, mejorar la capacidad de respuesta y asegurar la continuidad del funcionamiento de los sistemas de información.

2. Estado del Arte

Los Modelos Relacionales son la representación más común de estructuras de datos. La información de negocios de las empresas, marketing y datos de ventas, registros médicos y conjunto de datos científicos son almacenados en bases de datos relacionales. Actualmente ha crecido el interés en extraer patrones estadísticos de grandes cantidades de datos, estos patrones proveen un entendimiento más profundo del dominio y las relaciones. Extraer patrones puede ser usado para lograr conclusiones acerca de atributos importantes cuyo valor puede ser desapercibido [Geetor et al., 2001].

Los modelos gráficos probabilísticos y particularmente las redes bayesianas se muestran como el camino para representar patrones estadísticos en el dominio del mundo real. Trabajos realizados mencionados en [Geetor et al., 2001] desarrollan técnicas para aprender de estos modelos directamente de los datos y muestran lo interesante que son los patrones que a menudo emergen de este proceso de aprendizaje. Sin embargo, estas técnicas de aprendizaje se aplican solo para la representación de datos en texto plano y no para datos relacionales que se encuentran en la mayoría de las aplicaciones. Una solución sencilla es tomar la base de datos relacional y aplanarla, creando archivos de texto plano donde los algoritmos de aprendizaje estándares de las redes bayesianas pueden correr, pero este método tiene sus deficiencias.

Los Modelos Probabilísticos Relacionales MPR actuales, extienden el estándar de representación de redes bayesianas, ya que se basan en atributos para representar mucho mejor las estructuras relacionales. Estos modelos permiten la especificación de un modelo de probabilidad para clases y no solamente para atributos simples, sino que también permiten que propiedades de una entidad dependan probabilísticamente de propiedades de otras entidades relacionadas. El modelo probabilístico relacional representa una dependencia genérica, la cual es luego instanciada por circunstancias específicas como conjuntos particulares de entidades y relaciones entre ellas [Geetor et al., 2001].

Un MPR provee un modelo estadístico que puede descubrir dependencias probabilísticas interesantes que se espera en un dominio y especifica una distribución en conjunto sobre un dominio relacional. El algoritmo de aprendizaje de un MPR intenta descubrir las más significantes dependencias directas en los datos. El modelo resultante provee un alto nivel, esquemas cualitativos de la estructura del dominio, y adicionalmente provee información cuantitativa para las distribuciones de probabilidad [Geetor et al., 2001].

A continuación, se describirán los principios básicos sobre los que se construyen los MPR, comenzando por los modelos probabilísticos y continuando con los modelos relacionales, [Geetor et al., 2001].

2.1. Modelos Probabilísticos

El enfoque basado en la lógica tradicional de representar el conocimiento, está escrito bajo una base de conocimientos en forma de axiomas lógicos sobre el dominio. La base de conocimientos restringe el conjunto de mundos posibles, o modelos con axiomas consistentes, es decir hechos que son verdaderos en todos estos mundos posibles, están lógicamente ligados a la base de conocimientos. Un marco lógico estándar se limita a la representación única de hechos que son absolutamente verdaderos, por lo tanto, este marco no puede representar y razonar con información incierta, esto es una brecha significativa en el poder expresivo del marco, y una gran barrera para su uso en muchas aplicaciones del mundo real. La incertidumbre es inevitable en el mundo real, nuestra información es a menudo imprecisa y siempre incompleta, y sólo unas de las pocas reglas que utilizamos para el razonamiento son verdaderas. Esta limitación, que es fundamental en muchos dominios, por ejemplo, el diagnóstico médico, ha llevado en la última década al resurgimiento del razonamiento probabilístico en la inteligencia artificial como los modelos de teoría de probabilidad de la incertidumbre, mediante la asignación de una probabilidad para cada uno de los estados del mundo que un agente considera posible. En el razonamiento probabilístico los estados son el conjunto de posibles asignaciones de valores a un conjunto de atributos o variables aleatorias. Considere, por ejemplo, un modelo simple del rendimiento de un estudiante en un curso. Hay seis variables aleatorias: la Inteligencia, la Dificultad, un buen Profesor, el Entendimiento del Material, la Nota del Examen y la Nota de la Tarea para la Casa. De estas variables, la Inteligencia, el buen Profesor y el Entendimiento del Material son variables booleanas, la Dificultad toma valores de (baja, media, alta), la Nota del Examen y la Nota de la Tarea para la Casa toma valores de {A, B, C, D, F}. Los mundos posibles son todas posibles asignaciones de valores a estas variables, (2 x 2 x 2 x 3 x 5 x 5) 600 valores posibles en este caso.

Un modelo probabilístico especifica una distribución conjunta sobre todos los mundos posibles. Por lo tanto, se especifica implícitamente la probabilidad de cualquier evento como una asignación de valores para algún subconjunto de variables. A diferencia de muchos modelos, como un conjunto de reglas utilizadas para predecir algún atributo particular, un modelo probabilístico no se limita a conclusiones acerca de un conjunto predefinido de atributos, sino se puede utilizar para responder a las consultas sobre cualquier variable o un subconjunto de variables. Por ejemplo, un modelo probabilístico del rendimiento de un estudiante se puede utilizar para predecir la distribución sobre la nota del examen de los estudiantes, dada su inteligencia. A medida que se obtiene nueva evidencia, por ejemplo, aproximadamente su grado de preparación, esta se puede utilizar para actualizar esta probabilidad, entonces la probabilidad de obtener una alta nota de examen va a subir si observamos buenas notas en las tareas para la casa.

Por otra parte, un modelo probabilístico puede realizar explicaciones alejadas que utilizan evidencia de apoyo a una causa para disminuir la probabilidad de otra, no porque las dos sean incompatibles, sino porque simplemente la primera causa explica a distancia la evidencia, quitando el apoyo para la otra causa. Por ejemplo, si se observa que la nota del examen del estudiante es alta, la convicción de que es inteligente va a subir. Si entonces nos enteramos de que la clase, es conocida y es fácil, este hecho proporciona una explicación alternativa para los estudiantes, reduciendo nuestra creencia de que un

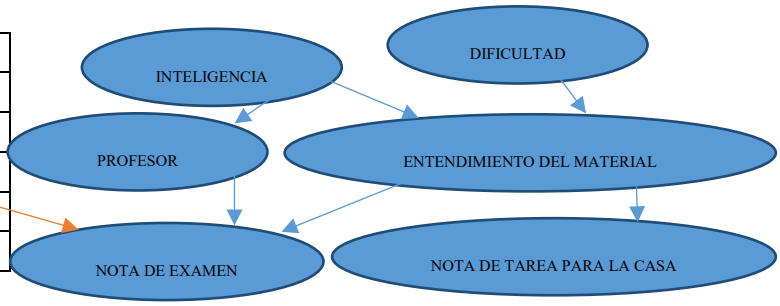
estudiante es inteligente. El mismo modelo probabilístico soporta todos estos patrones de razonamiento, lo que le permite ser utilizado en muchas tareas diferentes.

La objeción tradicional a los modelos probabilísticos ha sido su costo computacional. Una distribución completa de probabilidad conjunta sobre un conjunto de variables al azar debe especificar una probabilidad para cada uno de las exponencialmente diferentes instancias del conjunto. Incluso en el ejemplo muy simple anterior, tenemos que especificar 600 números para especificar la distribución conjunta. Este tipo de representación es poco práctica desde el punto de vista de ingeniería del conocimiento, ya que es casi imposible para una persona especificar una entrada en una compleja distribución conjunta, mucho menos especificar un número exponencial de ellos, y desde una perspectiva de razonamiento cualquier cálculo obliga a enumerar un exponencial número de eventos.

Por lo tanto, una representación ingenua de una distribución conjunta es posible hasta incluso para los dominios más simples. Las redes bayesianas utilizan una estructura subyacente del dominio para superar este problema. La idea clave es la influencia presente en muchos ámbitos de la vida real, es decir cada variable es directamente influenciada por sólo unas pocas variables seleccionadas. Por ejemplo, un estudiante inteligente induce una mejor comprensión del material, que a su vez conduce a una nota superior en las tareas para la casa. Pero, el efecto de la inteligencia en la nota de la tarea para la casa es una variable indirecta, ya que, si el estudiante no entiende el material, su inteligencia no ayuda a conseguir mejores calificaciones.

Los nodos de una red bayesiana representan las variables aleatorias y las flechas representan las dependencias directas. La Figura 1 muestra una red bayesiana para el dominio del rendimiento de un estudiante.

P	C	P (N | P,C)				
		A	B	C	D	F
t	t	0.7	0.25	0.03	0.01	0.01
t	f	0.3	0.4	0.2	0.05	0.05
f	t	0.4	0.3	0.2	0.08	0.02
f	f	0.05	0.2	0.3	0.3	0.15



P (I, D, P, E, N, C) = P (I) P (D) P (P | I) P (E | I, D) P (N | P, E) P (C | E)

Figura 1. Una red bayesiana para el dominio del rendimiento de un estudiante que muestra la descomposición de la distribución conjunta, en un producto de Distribuciones de Probabilidad Condicional (DPCs), y una (DPC) para uno de los nodos de la red.

La red afirma que cada nodo o variable aleatoria es condicionalmente independiente de sus no-descendientes dado los valores de sus padres. Por ejemplo, si sabemos que el estudiante no entiende el material, la distribución sobre sus notas no tiene gran influencia en la información que podamos tener acerca de su inteligencia. Estos supuestos de independencia condicional permiten una representación muy concisa de la distribución de probabilidad conjunta sobre las variables aleatorias, se asocia con cada nodo una Distribución de Probabilidad Condicional DPC, la cual específica para cada nodo X la distribución de probabilidad sobre los valores de X, dada cada combinación de valores

para sus padres, representados como Pa(X). La independencia condicional supone asociarse con las redes bayesianas, lo que implica que estos números son suficientes para determinar unívocamente la distribución de probabilidad sobre las variables aleatorias. Más precisamente, la distribución conjunta sobre todas las variables puede ser factorizada en un producto de las DPCs de todas las variables a través de la siguiente regla de la cadena para Redes Bayesianas.

$$P(X_1,\ldots, X_n) = \prod P\ (X_i \mid Pa\ (X_i))$$

En el caso planteado, el rendimiento de un estudiante depende de su inteligencia y de la dificultad de la clase. Su nota de examen depende de si tiene un buen profesor, y su entendimiento del material, mientras que su nota de tarea para la casa depende de su comprensión del material. La estructura de la red codifica un número de afirmaciones de independencia condicional. Por ejemplo, la nota de examen de los estudiantes es condicionalmente independiente de su inteligencia, dada su capacidad para rendir exámenes y la comprensión del material.

Estos supuestos de independencia permiten factorizar la distribución conjunta en forma de un producto tal como se muestra en la Figura 1. Cada una de las probabilidades condicionales en forma de producto es una DPC de una de las variables. En este ejemplo, la DPC es simplemente una tabla, tal como la que se muestra para P (N | P, C) en la Figura. Esta DPC muestra que, si un estudiante tiene un buen profesor y entiende el material, entonces tiene una probabilidad 0,7 de conseguir una A en el examen, mientras que, si el estudiante tiene un mal profesor y no entiende el material, su probabilidad de obtener una A sólo 0.05.

Para el aprendizaje de la estructura de una red bayesiana, los enfoques típicos usan una función de puntuación que se basa generalmente en consideraciones bayesianas para puntuar cómo las diferentes estructuras coinciden con los datos de entrenamiento. El proceso de aprendizaje luego se reduce a la tarea de la búsqueda de la estructura de puntuación más alta. Estas técnicas permiten descubrir una estructura de red bayesiana desde los datos. La estructura aprendida a menudo nos puede dar una idea acerca de la naturaleza de las conexiones entre las variables en el dominio. Además, la estructura gráfica puede ser interpretada causalmente, lo que nos permite inducir la causa y efecto, que puede ser muy útil para la comprensión de nuestro dominio y para llegar a conclusiones sobre las consecuencias de la interpretación en el dominio [Geetor et al., 2001].

El aprendizaje de la red bayesiana se ha aplicado con éxito a las aplicaciones minería de datos, por ejemplo, muestran cómo una red bayesiana puede aprender de los datos que describen las preferencias de la gente a través de una variedad de artículos. Las dependencias aprendidas corresponden a las correlaciones entre las preferencias de una persona para diferentes elementos. Además de su capacidad de predicción, la red bayesiana tiene la ventaja de que proporciona una visualización de las más significantes correlaciones directas en el dominio, clarificando la estructura del dominio para el usuario [Breese et al., 1998].

2.2 Modelos Relacionales

Durante la última década, las redes bayesianas se han utilizado con gran éxito en una amplia variedad de aplicaciones del mundo real y de investigación. Sin embargo, a pesar de su éxito, las redes bayesianas son a menudo insuficientes para modelar adecuadamente aspectos de dominios relacionales complejos. Una red bayesiana para un dominio dado implica un conjunto predefinido de variables aleatorias, cuya relación entre ellas se fija de antemano. Por lo tanto, una red bayesiana no se puede utilizar para tratar con dominios donde podríamos encontrar varias entidades en una variedad de configuraciones. Esta limitación de las redes bayesianas es una consecuencia directa del hecho de que les falta el concepto de "objeto" o entidad de dominio. Por lo tanto, no pueden representar los principios generales sobre múltiples objetos similares, que luego se pueden aplicar en múltiples contextos.

La lógica relacional, ha formado tradicionalmente la base para los sistemas de representación del conocimiento a gran escala, abordando problemas mencionados anteriormente. La noción de los individuos, sus propiedades y las relaciones entre ellos proporcionan un marco elegante y expresivo para razonar acerca de diversos dominios. El uso de la cuantificación nos permite representar de forma compacta reglas generales, que se pueden aplicar en muchas situaciones diferentes. Por ejemplo, razonando acerca de la transmisión genética de ciertas propiedades (enfermedades de transmisión genética), podemos escribir las normas generales que se esperan para todas las personas y muchas propiedades.

Un esquema de un modelo relacional describe un conjunto de clases, $X = \{X1, \ldots, Xn\}$. Cada clase está asociada con un conjunto de atributos descriptivos y un conjunto de slots de referencia. Existe una correspondencia directa entre esta representación y la de las bases de datos relacionales. Cada clase corresponde a una sola tabla, los atributos descriptivos corresponden a atributos estándares de una tabla, y los slots de referencia corresponden a atributos que son claves foráneas (atributos clave de otra tabla).

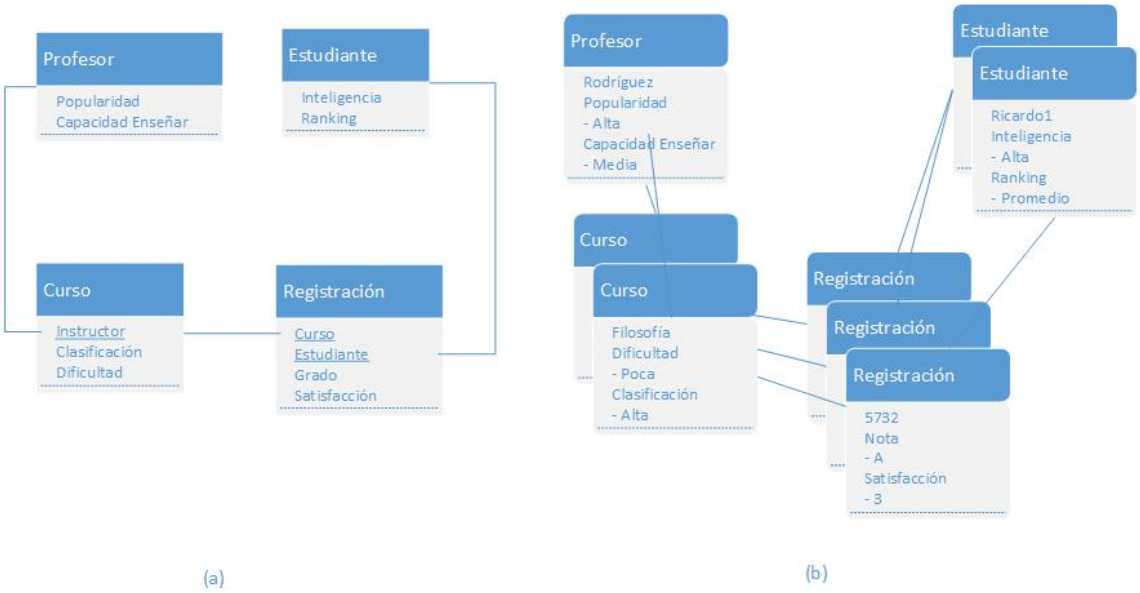


Figura 2 (a) Representa un esquema relacional para un dominio simple de universidad. Los atributos subrayados son slots de referencia de la clase y las líneas discontinuas indican los tipos de los objetos de referencia. Figura 2 (b) Representa una instancia ejemplo de este esquema. Aquí no se muestran los slots de referencia y se utilizan líneas discontinuas para indicar las relaciones que se mantienen entre los objetos.

La figura anterior muestra un esquema de un dominio simple de una universidad, y contiene profesores, estudiantes, cursos y registros del curso. Las clases en el esquema son el ***Profesor***, el ***Estudiante***, el ***Curso*** y la ***Registración***. El conjunto de atributos descriptivos de una clase ***X*** se representa *A(**X**)*. El atributo *A* de una de clase ***X*** se representa ***X***.*A*, y su espacio de los valores se representa V(***X***.*A*). Por ejemplo, la clase ***Estudiante*** tiene los atributos descriptivos *Inteligencia* y *Clasificación*, el espacio de valores para ***Estudiante***.*Inteligencia* podría ser (alto, bajo). El conjunto de slots de referencia de una clase ***X*** se representa R(***X***). La notación ***X***.*p*, se utiliza para representar el slot de referencia *p* de ***X***. Cada slot de referencia *p* es escrito, es decir, el esquema especifica el tipo de rango de objeto que puede ser referenciado. Por ejemplo, la clase ***Curso*** tiene como slot de referencia *instructor* con el tipo de rango ***Profesor*** y clase de ***Registración*** tiene los slots de referencia *curso* y *estudiante*.

Por ejemplo, la Figura 2 (b) muestra un ejemplo del esquema, en esta simple instancia de objetos hay un ***Profesor***, dos ***Cursos***, tres ***Registraciones*** y dos ***Estudiantes***. Las relaciones entre ellos muestran que el Profesor es el Instructor en ambos Cursos, y que un Estudiante ("Ricardo1") se ha registrado sólo para el curso ("Filosofía"), mientras que el otro estudiante se ha registrado para ambos cursos [Geetor *et al*., 2001].

2.3 Modelos Probabilísticos Relacionales

Los Modelos Probabilísticos Relacionales MPR, amplían el trabajo de las redes bayesianas con los conceptos de las entidades, sus propiedades y relaciones entre ellas. En cierto modo estos modelos, son a las redes Bayesianas como la lógica relacional es a la lógica proposicional. Las redes bayesianas tienen una semántica formal en términos de distribuciones de probabilidad sobre conjuntos de interpretaciones proposicionales que son asignaciones de valores a atributos. Los MPR tienen una semántica formal en términos de distribución de probabilidad sobre conjuntos de interpretaciones de lógica relacional [Getoor *et al*., 2001].

Un MPR es similar a un diagrama de clases en el Lenguaje Unificado de Modelado (UML) y contiene clases, atributos y relaciones. Además, asocia el modelo probabilístico a las relaciones que se definen entre los atributos de las clases. Más específicamente, un MPR hace que sea posible definir cómo el valor de un atributo depende del valor de otros atributos en un modelo. Con estos elementos un MPR permite en un sentido general, acoplar un modelo de arquitectura a un motor de inferencia probabilística. Por ejemplo, en el ámbito de la ciberseguridad se puede especificar de qué manera las diferentes arquitecturas de redes y propiedades de sus usuarios influyen en el riesgo de seguridad que enfrenta una organización. Si los modelos se expresan utilizando el formalismo MPR, se puede especificar cómo el riesgo de seguridad debe deducirse de las instancias del

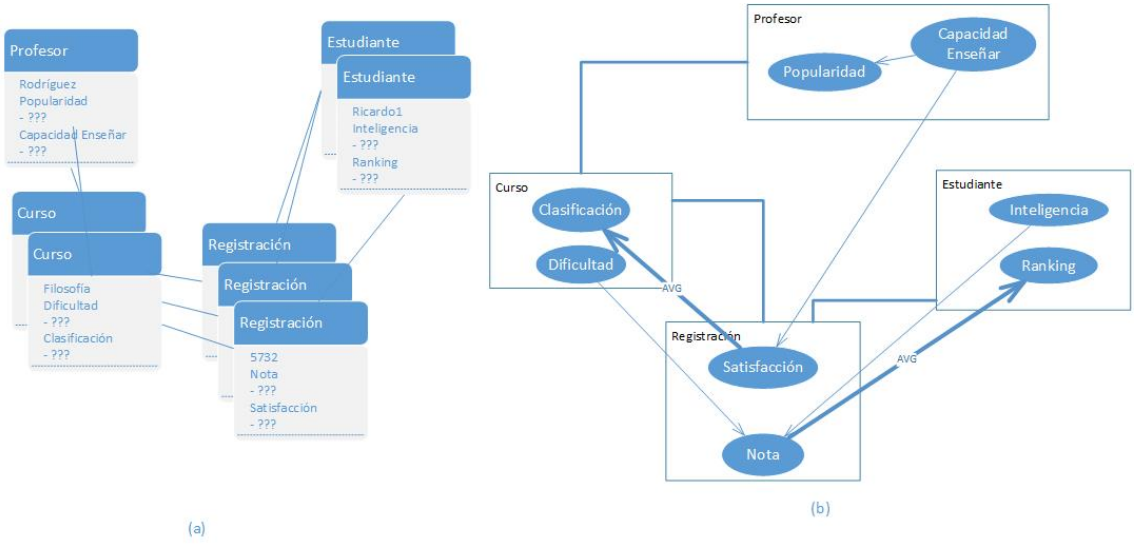
modelo. Sin embargo, existe un número infinito de maneras en que un MPR puede estructurarse para el análisis de riesgos de seguridad.

Las clases del modelo MPR son abstractas y no pueden ser directamente instanciadas en un modelo de arquitectura. Sin embargo, pueden ser utilizadas si están especializadas en subclases según un conjunto de restricciones. Es posible inferir el riesgo de seguridad de un modelo de arquitectura si las instancias de las clases son concretas. Esta inferencia también se puede realizar en modelos de arquitectura que simplemente representan activos y sus relaciones [Sommestad *et al*., 2010].

Es decir, un modelo probabilístico relacional MPR especifica una plantilla para una distribución de probabilidades sobre un modelo de arquitectura. La plantilla describe un meta modelo para un modelo de arquitectura y las dependencias probabilísticas entre los atributos de objetos. Un MPR, junto con un modelo de arquitectura de los objetos y las relaciones, define las distribuciones de probabilidad sobre los atributos de los objetos.

2.4 Lenguaje Básico de los Modelos Probabilísticos Relacionales

Los MPR proporcionan un lenguaje para especificar una distribución de probabilidad sobre un conjunto interpretaciones de relacionales. Más precisamente, un MPR especifica una distribución sobre un conjunto de instancias de un esquema determinado. Se podría considerar que un MPR especifica una distribución sobre todas las posibles instancias de un esquema, es decir, todas las posibles bases de datos sobre ese esquema. Este conjunto de bases de datos es infinitamente grande, ya que incluye todas las posibles variaciones en el número de objetos en cada clase y las posibles relaciones entre ellos. Es claramente muy difícil colocar una distribución sobre este tipo de espacio, y no es obvio que tal distribución de propósito general sea útil. Un MPR es una plantilla, dado un conjunto de objetos y especifica una distribución de probabilidad sobre un conjunto de interpretaciones que incluyen estos objetos y tal vez otros objetos.



La Figura 3 (a) muestra el esqueleto relacional de la instancia que se muestra en la Figura 3 (b). La Figura 3 (b) muestra un ejemplo de la estructura del MPR para el dominio de una universidad.

Un MPR especifica la distribución de probabilidad utilizando los mismos principios utilizados en la especificación de las redes bayesianas. Se supone que cada variable aleatoria en el MPR, en este caso los atributos de $\underline{\boldsymbol{X}}.A$ del objeto individual $\underline{\boldsymbol{X}}$ están directamente influenciado por solo unos pocos otros. El MPR por lo tanto define para cada $\underline{\boldsymbol{X}}.A$ un conjunto de padres, los cuales están directamente influenciados por este y se especifica la dependencia sobre estos padres. Sin embargo, existen dos diferencias principales entre los MPR y las redes bayesianas. En primer lugar, un MPR define el modelo de dependencias al nivel de clase, lo que le permite ser utilizado para cualquier objeto en la clase. En este sentido, el modelo de dependencias de clase es universalmente cuantificado e instanciado para cada elemento en el dominio de clase. En segundo lugar, el MPR utiliza explícitamente la estructura del modelo relacional, que permite que el modelo probabilístico de un atributo de un objeto dependa también de los atributos de objetos relacionados.

En la Figura 3 (b), la estructura de dependencias se define mediante la asociación con cada atributo $\boldsymbol{X}.A$ a un conjunto de padres $Pa(\boldsymbol{X}.A)$. Estos corresponden a los padres formales y serán instanciados en diferentes formas para diferentes objetos en $\boldsymbol{X}$. Intuitivamente, los padres son los atributos que son "influencias directas" sobre $\boldsymbol{X}.A$. En la figura 3 (b), las flechas definen la estructura de dependencia.

Se distingue entre dos tipos de padres formales. El atributo $\boldsymbol{X}.A$ puede depender de otro atributo probabilístico B de X. Esta dependencia formal induce una dependencia correspondiente para objetos individuales, para cualquier objeto $\underline{\boldsymbol{X}}$ en la clase $\boldsymbol{X}$, $X.A$ dependerá probabilísticamente de $X.B$.

Por ejemplo, en la Figura 3 (b), la *Popularidad* de un ***Profesor*** depende de su *Capacidad de Enseñanza*. Este modelo de dependencia se duplica por cada ***Profesor*** en el esqueleto. Por lo tanto, esencialmente se supone que se aplica el mismo modelo probabilístico a todos los Profesores de nuestro dominio. Adicionalmente un atributo $\boldsymbol{X}.A$ puede también depender de atributos de objetos relacionados $\boldsymbol{X}.\tau.B$, donde τ es un slot cadena. En la Figura 3 (b), la *Calificación* de un ***Estudiante*** en un curso, ***Registracion***.*Calificacion* depende de ***Registracion.Estudiante***.*Inteligencia* y de ***Registracion.Curso***.*Dificultad*.

El lenguaje del MPR también permite usar grandes slots cadena, por ejemplo, la dependencia del objeto ***Estudiante***.*Satisfacción* sobre el objeto ***Registracion.Curso.Profesor***.*HabilidadEnseñanza*. Tales slots cadena están instanciados por cada objeto siguiendo las referencias que son asignadas para el esqueleto. Por ejemplo, para el objeto Registración #5639, ***Registracion.Estudiante***.*Inteligencia* referencia a ***Ricardo1***.*Inteligencia*, y el slot ***Registracion.Curso***.*Dificultad* referencia a ***Filosofia***.*Dificultad*.

Este ejemplo de MPR también contiene una dependencia ***Estudiante***.*Clasificación* de ***Estudiante***.*Nota-Registrada*. Tenga en cuenta que un estudiante normalmente será registrado en varias clases, el modelo especifica una dependencia de ***Estudiante***.*Clasificación* sobre las notas que recibe en todas ellas.

Al igual que en las redes bayesianas, el segundo componente de un MPR es el parámetro asociado con la estructura cualitativa. Un MPR contiene una DPC para cada uno de los

atributos de cada clase. Como en las dependencias se asume que los parámetros son compartidos por cada objeto en una clase. Se asocia con cada atributo $\boldsymbol{X}.A$ con una Distribución de Probabilidad Condicional $P(\boldsymbol{X}.A \mid Pa(\boldsymbol{X}.A))$.

2.5 Comparación de Lenguajes de Modelado para Evaluaciones de Seguridad

Existen varios lenguajes de modelado como Secure UML [Lodderstedt *et al*., 2002] y SPML (Lenguaje de Modelado para Políticas de Seguridad, de sus acrónimos en ingles Security Policy Modeling Language) [Trevisani, & Garcia, 2008]. Estos ofrecen un lenguaje para describir amenazas y contramedidas, pero no están asociados con métodos para evaluar cuantitativamente la seguridad.

Otros dos lenguajes de modelado analizados son Secure Tropos [Mouratidis H., 2002] y UMLsec [Jürjens J., 2005], ambos poseen un lenguaje y una metodología que pueden proporcionar una base para las evaluaciones de seguridad. Secure Tropos puede utilizarse para especificar problemas de seguridad asociados con los sistemas planificados. La extensión de UML, UMLsec, proporciona un lenguaje para representar información relevante para la seguridad en diagramas que describen una especificación de un sistema [Jürjens J., 2005]. Ambos proporcionan soporte para la verificación automatizada de modelos de arquitectura, SecureTropos se asocia con un conjunto de reglas que se pueden usar para verificar si se cumplen los objetivos relacionados y UMLsec permite evaluar si el diagrama UMLsec cumple con un conjunto de requisitos estipulados. Sin embargo, la salida de estos métodos de análisis automatizados es un resultado positivo / negativo que indica si la arquitectura cumple con los requisitos. Estas verificaciones pueden admitir el análisis de riesgos de seguridad, pero no hay medios automatizados para calcular los riesgos de seguridad directamente de ellos.

El resultado positivo / negativo también es una característica de COMMON CRITERIA que ofrece un método para especificar los requisitos de seguridad y evaluar su cumplimiento [Herrmann D., 2002]. Se describen las relaciones entre propietarios, activos, riesgos, contramedidas, agentes de amenazas y amenazas, sin embargo, la evaluación que utiliza no cuantifica el riesgo. En su lugar, proporciona un resultado positivo / negativo junto con una calificación del nivel de seguridad.

Un método desarrollado específicamente para analizar y cuantificar el riesgo es CORAS, se crea una descripción gráfica del escenario de amenaza y se utiliza como soporte para determinar cómo deben tratarse los riesgos identificados [Hogganvik I., 2007]. Esto se hace modelando las relaciones entre activos, amenazas, vulnerabilidades, eventos no deseados, riesgos y tratamientos. Aunque el riesgo en CORAS se define como el producto de la probabilidad y la consecuencia, no existe un marco de análisis acoplado y, por lo tanto, no hay un método algorítmico para calcular el riesgo basado en una descripción gráfica. Tampoco hay una descripción de los diferentes tipos de tratamientos de riesgo que deben ser modelados, o cómo los tratamientos de riesgo influyen en los riesgos.

Otros métodos de análisis también dependen de analistas especializados para cuantificar el riesgo. La metodología de la CCTA (Central Computing and Telecommunications Agency) del gobierno británico denominada CRAMM (Método de Análisis y Gestión de Riesgos CCTA de sus acrónimos en ingles CCTA Risk Analysis and Management

Method) ofrece un método estructurado para evaluar cualitativamente el riesgo al identificar:

a. La frecuencia con la que ocurre un incidente.
b. La probabilidad de que los incidentes resulten en el peor escenario.
c. Los valores de pérdida.

Estos tres valores se utilizan para producir un valor monetario para la expectativa de pérdida anual [Yazar Z., 2002].

También ISRAM (Método de Análisis de Riesgos de Seguridad de la Información de sus acrónimos en ingles Information Security Risk Analysis Method) de forma similar a CRAMM, evalúa las probabilidades de que ocurran incidentes de seguridad y evaluar las posibles consecuencias de estos [Karabacak & Sogukpinar, 2005].

Los métodos mencionados anteriormente proporcionan un vínculo entre las vulnerabilidades y las posibles consecuencias. Su estructura también facilita medios sencillos para el análisis, por ejemplo, la probabilidad de que un ataque tenga éxito o si se puede alcanzar un paso de ataque. Sin embargo, ninguno de estos métodos mencionados ofrece medios para evaluar la probabilidad de que un adversario intente combinaciones de pasos de ataque. Por lo tanto, la probabilidad de que se realice un ataque determinado no se puede inferir y el riesgo de seguridad no se puede calcular [Liu & Man, 2005].

A diferencia de los enfoques mencionados anteriormente, la teoría aquí presentada hace posible especificar una arquitectura que genera un modelo de dependencia probabilística cuando se crean instancias. La estructura conceptual de este modelo de dependencia extiende el modelo conceptual presentado en COMMON CRITERIA al definir los pasos de ataque como parte de una amenaza. Los pasos de ataque están relacionados con contramedidas y también pueden asociarse entre sí de forma probabilística. Es decir, el modelo propuesto permite inferir la probabilidad de los pasos de ataque.

3. Elaboración de la Teoría Propuesta

3.1 Principios de la Teoría Propuesta

En base a la arquitectura de los Sistemas de Ciberseguridad Convencionales propuesta, se evaluaron los aspectos más relevantes para poder elaborar una teoría que pueda ser modelada, a continuación, se detallan los aspectos principales:

La nomenclatura utilizada en este artículo para representar una clase, es **(Clase1)** y para representar los atributos de una clase, es (**Clase1.***atributo).* Una clase instanciada se representa (**ClaseInstanciada**).

CLASES PRINCIPALES

Hay cuatro tipos de clases principales en el modelo propuesto y del cual se heredan subclases: **Atacante**, **PasoDeAtaque, Defensa** y **Activo**. Cada **PasoDeAtaque** que pone en peligro un **Activo** está conectado a él y la **Defensa** que protege un **Activo** también se

conecta a él, ver Figura 4 (a). Por ejemplo, los atributos **PasoDeAtaque.***EcontrarServicioDesconocido* y **Defensa**.*Antimalware* están conectados al **Activo**.**SistemaOperativo,** ver Figura 4 (b).

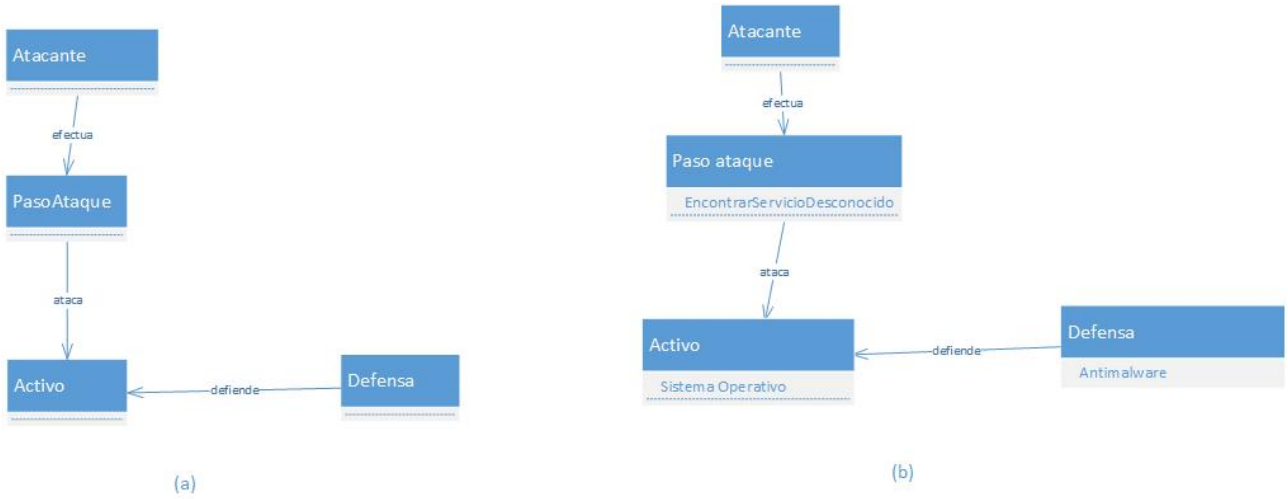


Figura 4 (a). Clases Principales del modelo propuesto.

Figura 4 (b). Ejemplo de clases Principales del modelo propuesto.

Atacante

En modelo propuesto, un atacante constituye un individuo que está decidido a comprometer los activos del modelo de objetos representado. Naturalmente, las características de este atacante influirán en los ataques que son posibles, y la probabilidad de que sus actividades tengan éxito [Liu & Cheng, 2009]. En el modelo propuesto, se supone que el atacante es un individuo con acceso a las herramientas y técnicas disponibles para el público para realizar pruebas de pentesting. En consecuencia, los pasos de ataque y estimaciones dentro del modelo deben ser vistos de acuerdo con el perfil del atacante y al tiempo disponible en días que tiene el atacante para concretar un ataque.

El atacante tiene un atributo *Tiempo*, esto específica el número de días de trabajo que un atacante tiene que gastar en cada paso de ataque en el modelo propuesto. Computacionalmente, se evalúa la probabilidad de que cada paso de ataque en el modelo sea VERDADERO, es decir se llegue a concretar con éxito en la cantidad de días de trabajo especificados.

Pasos de Ataque

En el modelo propuesto, son los distintos medios a los que tiene acceso un atacante para explotar una o más vulnerabilidades de un activo. Por lo tanto, el modelo propuesto describe la probabilidad de que un atacante sea capaz de concretar diferentes Pasos de Ataque.

Activo

Es una parte fundamental para el almacenamiento y gestión de la información, así como para el funcionamiento de una organización. La seguridad informática está concebida

para proteger los activos informáticos, en el modelo propuesto de agruparan instancias de la clase Activo, de acuerdo al tipo de defensa en el que están involucradas.

a. En la defensa basada en red:

- Zona de red
- Interfaz de red
- Servidor de aplicaciones
- Aplicación cliente
- Aplicación Web

b. En la defensa basada en Host:

- Sistema Operativo

Defensa

En el modelo propuesto, son las contramedidas adoptadas por los Sistemas de Ciberseguridad Convencionales para prevenir, detectar y contrarrestar los ataques maliciosos.

3.2 Modelado de la Teoría Propuesta

La instanciación y la creación del modelo propuesto se realiza utilizando la herramienta EAAT versión 1.0.0 y la herramienta CySeMoL versión 2.4. EAAT es una herramienta de modelado de arquitectura empresarial que utiliza notación UML para la representación visual y para dar un P^2AMF (Marco de Modelado de Arquitectura Probabilística y Predictiva de sus acrónimos en ingles Predictive, Probabilistic Architecture Modeling Framework) [Johnson *et al*., 2013]. La característica principal de P^2AMF es su capacidad para expresar incertidumbres de objetos, relaciones y atributos en modelos UML y realizar evaluaciones probabilísticas que incorporan estas incertidumbres, un uso típico de P2AMF sería crear un modelo para predecir, por ejemplo, la disponibilidad de una aplicación.

En cuanto a CySeMoL (Lenguaje de Modelado de Ciberseguridad de sus acrónimos en ingles Cyber Security Modeling Language), es una herramienta de gráfico de ataque que se puede utilizar para estimar la seguridad cibernética de las arquitecturas empresariales. En este artículo se utiliza para describir cómo los pasos ataque y contramedidas se relacionan entre sí y cómo pueden ser utilizados para evaluar la ciberseguridad de una arquitectura de sistema de TI. Cabe destacar, que existen más de 50 tipos de plantillas de ataque en el modelo CySeMoL original, y solo siete de ellas fueron seleccionadas como guía para trabajar en la construcción del modelo propuesto. Dos criterios se consideran cuando se seleccionan los pasos de ataque, la primera es la capacidad para poder efectuar pruebas, es decir el paso de ataque seleccionado debe ser adecuado para pruebas en un entorno de red estándar con herramientas públicas disponibles. El segundo criterio es el grado de dificultad, debido a la limitación del conocimiento y de la experiencia práctica

del pentester, en este caso el autor, de manera de manifestar que cualquier responsable de la seguridad de informática en una organización lo puede realizar, sin necesidad de contratar a un experto.

El modelo construido se muestra a continuación en la Figura 5, donde se identifican las clases que representan entidades del modelo, las relaciones entre las clases mencionadas y los atributos que componen las clases.

Figura 5 Modelo propuesto construido.

3.3 Construcción del Modelo Propuesto

En esta parte, se describen las fases de selección de aplicaciones y configuración del modelo propuesto. El método por seguir se divide en dos fases como se muestra en la siguiente Figura.

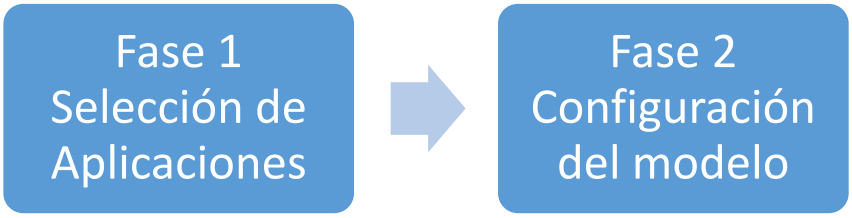


Figura 6 Fases de configuración del Modelo.

La primera fase del método se selecciona el Host (sistema operativo) y los servicios más importantes de este último, las aplicaciones típicas que se utilizan en un Sistema de Ciberseguridad Convencional y con los que se trabajará en el modelado. Por otra parte, se tomaron como marco de referencia las plantillas del modelo CySeMoL y se adaptaron para construir el modelo propuesto en este artículo.

La segunda fase es la configuración del modelo, se configuran los parámetros necesarios en la herramienta de modelado de objetos EAAT, es decir de acuerdo con la información recopilada, el modelo se construye, y se configura la seguridad en el modelo. Luego se describen los mecanismos de defensa de los activos representados en el modelo y se muestran los parámetros configurados en la construcción del modelo.

4. Solución

4.1 Selección De Aplicaciones Pc Victima

Un estudio estadístico de la Cuota de mercado de los principales sistemas operativos a nivel mundial en el primer semestre de 2018, muestra que Windows 10 fue el segundo sistema operativo más instalado, con una cuota de mercado de un 34,48%. Esto significa que aproximadamente 35 de cada 100 ordenadores tenían instalado este sistema operativo. Por su parte, Windows 7, registró una cuota de mercado superior al 42%, un valor que le sigue situando a la cabeza del ranking a pesar de los esfuerzos de Windows para potenciar el uso de Windows 10 [Statista, 2018]. Por lo mencionado anteriormente seleccionaremos como Sistema Operativo de Host Windows 7.

El Sistema Operativo Windows cuenta con servicios para su correcto funcionamiento, un servicio de Windows es un programa informático que se ejecuta en segundo plano, es similar a un demonio en un sistema UNIX. Los servicios de Windows proporcionan diversas funcionalidades como el logueo de eventos, la actualización del sistema, el control de errores y así sucesivamente [Liu & Cheng, 2009]. Debido a la numerosa cantidad de servicios que se ejecutan en segundo plano en el Sistema Operativo Windows,

y luego de una evaluación de los mismos, se seleccionan como entidades para el modelo propuesto los servicios Windows Update y Windows Time por considerarlos de mayor importancia.

a. PDF Reader

De acuerdo con el ranking de aplicaciones "WHATPULSE", Adobe Reader está en el puesto 17 en la lista total y es el número uno en lectores PDF en la lista [Bishop M., 2002]. Por lo tanto, Adobe Reader es elegida como una instancia de la clase **AplicaciónCliente** en el modelado. La versión de la aplicación Adobe Reader que ha sido probada es Adobe Reader XI versión 11.0.10.

b. Antivirus

OPSWAT compañía global de seguridad cibernética que ofrece soluciones de seguridad para empresas, publico un reporte sobre Antivirus y Dispositivos Comprometidos durante el año 2018. En la sección Cuota de Mercado de Productos Antivirus se grafica un ranking, que se muestra en la Tabla a continuación, donde Avast Software Antivirus lidera el mercado con una cuota del 18,44 % [Opswat, 2018]. Cabe resaltar que, Windows Defender ha sido eliminado en este cálculo de la cuota de mercado debido a que viene preinstalado en Windows, lo cual no es una selección de los usuarios. Por lo tanto, Avast Software Antivirus se selecciona como una instancia de la clase **Antivirus** en el modelo, y el número de versión es 17.6.2310.

Tabla 1 Ranking cuota de mercado de Antivirus.

Nombre del Antivirus	Cuota de Mercado
Avast software	18.44%
McAfee, Inc.	12.58%
Malwarebytes	12.23%
Bitdefender	10.64%
ESET	9.09%
Webroot Inc	7.03%
Kaspersky Lab	6.65%
Safer-Networking Ltd.	6.29%
Avira GmbH	4.5%

c. Navegador Web

Los datos de NetMarketShare, empresa que se dedica a proporcionar estadísticas de uso de la web en usuarios reales, muestran en la Tabla 2 que, durante el año 2018, Internet Explorer tuvo el 10,83 % de cuota de mercado entre los navegadores de escritorio/laptops y que es un jugador importante en el mercado [NetMarketShare, 2018].

Tabla 2 Ranking cuota de mercado de Navegadores Webs.

Navegador Web	**Cuota de Mercado (marzo 2015)**
Chrome	64.15%
Internet Explorer	10.83%
Firefox	9.89%
Edge	4.30%
Safari	3.80%
Opera	1.58%

Debido a que Internet Explorer ocupo el segundo lugar en el ranking de NatShareMarket y que es el navegador por defecto del Sistema Operativo Windows, fue seleccionado para ser utilizado en este articulo y representa una entidad en el modelo propuesto debido a la interacción que realiza entre el Host y la Zona de Red, el número de versión de software es 11.0.9600.17843.

d. Software de Oficina

De acuerdo con el ranking de aplicaciones [Whatpulse, 2018], Microsoft Word es la aplicación más utilizada como software de oficina, por lo tanto, se selecciona como instancia de la clase **AplicaciónCliente** para el modelo propuesto y forma parte de la Suite Microsoft Office Professional Plus 2013.

Tabla 3 Ranking cuota de mercado Software de Oficina.

Nombre de la Aplicación	Ranking de WhatPulse
Microsoft Word	16
Microsoft Excel	25
Microsoft PowerPoint	35

e. Cliente de correo

A partir de los datos proporcionados por un reporte de Litmus, empresa que se dedica a realizar testing de clientes de correo electrónico. Se puede observar en la siguiente Tabla que durante el año 2017 y mediados del año 2018, Microsoft Outlook ocupa el puesto número 5 en el ranking manteniendo una cuota de mercado del 7%. Se selecciona Microsoft Outlook como parte de las aplicaciones por defecto del Sistema Operativo Windows, representando una instancia de la clase **AplicacionCliente** en el modelo propuesto, la versión de Outlook elegida forma parte de la Suite Microsoft Office Professional Plus 2013.

Tabla 4 Ranking cuota de mercado Clientes de Correo.

Cliente de Correo	Cuota de Mercado
Apple iPhone	29%
Gmail	27%
Apple iPad	10%
Apple Mail	8%
Outlook	7%
Samsung Mail	4%
Android	3%
Outlook.com	3%
Yahoo! Mail	1%
Windows Live Mail	1%

f. Aplicación de Mensajería Instantánea

OPSWAT ha emitido un informe en el que cuenta la cuota de mercado de mensajería instantánea en todo el mundo. El informe especifica que Skype tiene una cuota de mercado del 37,7%, seguido de Windows Messenger con un 31,8% [Opswat, 2013]. Las estadísticas se presentan en la siguiente tabla a continuación:

Tabla 5 Ranking cuota de mercado Aplicaciones de Mensajería Instantánea.

Aplicación de Mensajería	Cuota de Mercado
Skype	37.7%
Windows Messenger	31.8%
Yahoo! Messenger	8.6%
Mail.Ru Agent	3.9%
Google Talk	3.9%
Facebook Messenger	3.9%

De acuerdo con la información anterior, se selecciona Skype como parte de las aplicaciones tradicionales que componen un Host, y representa una instancia de la clase **AplicacionCliente** en el modelo propuesto, y el número de versión de software de modelado es 7.40.0.103.

En resumen, la elección de las aplicaciones más importantes que componen un Host (Sistema Operativo Windows 7) en el modelo propuesto, se detallan a continuación:

- Adobe Reader XI version 11.0.10.
- Avast Free Antivirus version 17.6.2310.
- Internet Explorer version 11.0.9600.17843.
- Microsoft Office Word de la Suite Professional Plus 2013.
- Microsoft Outlook de la Suite Professional Plus 2013.
- Skype versión 6.16.60.105.

4.2 Selección de Servicios Pc Victima

Los servicios más conocidos del Sistema Operativo Windows 7, se enumeran con una breve descripción en la siguiente tabla.

Tabla 6 Servicios más conocidos del Sistema Operativo Windows 7.

N°	Servicio	Descripción
1	Windows Audio	Gestiona el audio de los programas basados en Windows.

2	Windows Audio Endpoint Builder	Gestiona los dispositivos de audio para los servicios de audio de Windows.
3	Windows Biometric Builder	Da las aplicaciones cliente la capacidad de capturar, comparar, manipular y almacenar los datos biométricos sin aumentar el acceso directo a cualquier biométrico hardware.
4	Windows Color System	El aumento del uso de contenido de color en todas las formas de comunicación digital requiere la aplicación de un sistema de gestión del color para los sistemas operativos Microsoft Windows.
5	Windows Connect Now	Implementación de Wi-Fi Protected de Microsoft (WPS).
6	Windows Connection Manager	Gestiona las opciones de conectividad de red.
7	Windows Defender Network Inspection	Controla intentos de intrusión.
8	Windows Defender Service	Protege a los usuarios del malware.
9	Windows Driver Foundation	Gestiona los procesos de driver en modo usuario.
10	Windows Encryption Provider Host Service	Evalua / aplica politicas EAS.
11	Windows Error Reporting Service	Reporta errores de programa.
12	Windows Event Collector	Gestiona eventos remotos.
13	Windows Event Log	Gestiona eventos y logs de eventos.

14	Windows Firewall	Protege a la computadora de accesos de usuarios no autorizados a través de la red.
15	Window Font Cache Service	Realiza los caches de las fuentes de datos.
16	Windows Image Acquisition	Provee la adquisición de imágenes para dispositivos.
17	Windows Installer	Gestiona los paquetes de aplicación.
18	Windows Location Framework Service	Monitorea las ubicaciones actuales.
19	Windows Management Instrumentation	Proporciona la interfaz y el modelo de objetos para acceder a la información de gestión.
20	Windows Modules Installer	Permite la manipulación de las actualizaciones de Windows y otros componentes.
21	Windows Presentation Foundation Font Service	Caches de fuentes de datos para Windows Presentation Foundation (WPF).
22	Windows Remote Management	Implementa el protocolo WS-Management para la gestión remota.
23	Windows Search	Indexación de contenido y búsqueda de resultados.
24	Windows Store Service	Provee la infraestructura de soporte para Windows Store.
25	Windows Time	Gestiona fechas y tiempos de sincronización.
26	Windows Update	Permite la manipulación de las actualizaciones de Windows y otros programas.

Como el propósito de este artículo está centrado en el área de ciberseguridad, los servicios que no ponen énfasis en la red y que no pueden llegar a impactar en la ciberseguridad son ignorados. Los servicios que no tienen conexiones a la red fuera de ámbito local son 1,2,3,4,9,15,16,17,19,20,21,23.

Además, hay que tener en cuenta que el Host tiene instalado Avast Free Antivirus y que Windows Defender está desactivado en el Sistema Operativo por defecto, por lo tanto, los servicios 7 y 8 no se están ejecutando en el sistema.

Algunos servicios sirven como una condición previa o pre-requisito para ejecutar otros servicios o programas, como los servicios 12, 17, 19, 24.

Tras la evaluación de los servicios que figuran en la lista, se seleccionaron 25 y 26 (Windows Update y Windows Time), porque estos dos servicios están orientados a conexiones de redes y al intercambio de mensajes con servidores en el exterior.

4.3 Selección de Aplicaciones Pc Atacante

En este caso seleccionaremos como Sistema Operativo Windows 10 Professional 64 bits, donde se instalaron las siguientes aplicaciones:

a. Sniffer

Un sniffer es un programa de captura de las tramas de una red de computadoras, se utilizan para realizar diferentes tareas, como monitorear redes para detectar y analizar fallos, o para realizar ingeniería inversa en protocolos de red. Sin embargo, también es habitual su uso para fines maliciosos, como robar contraseñas, interceptar correos electrónicos, espiar conversaciones de chat, e interceptar y analizar paquetes de red. En este articulo utilizaremos WinArpAttacker versión 3.0.2.0.

b. Motor de captura de paquetes

Permite a las aplicaciones capturar y trasmitir los paquetes de red puenteando la pila de protocolos de un Host y tiene útiles características adicionales que incluyen el filtrado de paquetes, un motor de generación de estadísticas de red y soporte para captura de paquetes. El motor consiste en un controlador, compatible con el sistema operativo para proveer acceso de red a bajo nivel, y una biblioteca que se usa para acceder fácilmente a las capas de red de bajo nivel. Para acceder a la conexión entre capas de red en el entorno de prueba seleccionamos WinPcap versión 10.2-5002.

c. Escáner de seguridad

Este software posee varias funciones para sondear redes de computadoras, incluyendo detección de equipos, servicios, puertos e información de sistemas operativos. Estas funciones son extensibles mediante el uso de scripts para proveer servicios de detección avanzados, detección de vulnerabilidades y otras aplicaciones. Además, durante un escaneo, es capaz de adaptarse a las condiciones de la red incluyendo latencia y congestión de la misma. En este artículo utilizaremos Nmap para Windows versión 7.6.0.

d. Analizador de Protocolos

Este tipo de software permite realizar análisis y sirve para solucionar problemas en redes de comunicaciones, desarrollo de software y protocolos, y también sirve como una herramienta didáctica. Posee una interfaz gráfica y muchas opciones de organización y filtrado de información para facilitar la lectura y comprensión de todo el tráfico que pasa a través de una red. En este artículo utilizaremos Wireshark para Windows versión 2.4.1.

En resumen, la elección de las aplicaciones que componen la suite de herramientas de ataque de la PC atacante en el modelo propuesto, se detalla a continuación:

a. WinArpAttacker versión 3.0.2.0.
b. WinPcap versión 10.2-5002.
c. Nmap para Windows versión 7.6.0.
d. Wireshark para Windows versión 2.4.1.

4.4 Configuración Del Modelo

En el modelo propuesto, tenemos que establecer el mecanismo de defensa de un activo antes realizar el modelado y poder calcular la probabilidad de éxito de que se concrete un ataque. En el caso de un activo con mecanismo de defensa, en la ficha de propiedades de la herramienta EAAT, hay dos opciones llamadas *Funcionamiento_EvidenciaParaInyectar* y *Funcionamiento_InyectarEvidencia*, estableciendo el valor VERDADERO o FALSO en estas dos opciones, serán configuradas las propiedades del mecanismo de defensa seleccionada para los activos del modelo propuesto. La opción *Funcionamiento_EvidenciaParaInyectar* es la evidencia de que debe ser inyectada, y *Funcionamiento_InyectarEvidencia* indica si la evidencia se debe utilizar o no. En la siguiente tabla se muestran los valores de muestra cuando se configura la condición de un mecanismo de defensa.

Tabla 7 Valores de las opciones de configuración de un mecanismo de defensa.

Opción	Estado Verdadero	Estado Falso
Funcionamiento_EvidenciaParaInyectar	Verdadero	Falso
Funcionamiento_InyectarEvidencia	Verdadero	Verdadero

a. Los Mecanismos de defensa de HOST (Sistema Operativo) seleccionados

Tabla 8 Mecanismos de defensa de Host seleccionados.

Defensa	Funcionamiento_EvidenciaParaInyectar
ASLR	Verdadero
Antivirus Instalado	Verdadero
Memoria No Ejecutable	Verdadero
Firewall De Host	Verdadero
Actualizaciones Automáticas	Verdadero
Tabla Estática ARP	Falso
Deshabilitar Autoejecutable USB	Verdadero

La ASLR (Aleatorización de Disposición del Espacio de Direcciones de sus acrónimos en ingles Address Space Layout Randomization), cambia aleatoriamente la dirección de memoria de un software en particular. El sistema operativo Windows tiene esta característica desde Windows Vista, por lo tanto, en Windows 7 el valor del atributo ALSR se establece en VERDADERO.

Actualmente no existe Host sin Antivirus instalado, por lo tanto, Avast Free Antivirus se configura como VERDADERO. La protección de Memoria No Ejecutable impide que una aplicación pueda ejecutar código desde una memoria no ejecutable, está defensa se encuentra disponible en Windows 7, por lo tanto, esta propiedad se establece en VERDADERO. El firewall de Windows de forma predeterminada está activado, por lo tanto, el Firewall de Host se configura como VERDADERO.

Los Sistemas Operativos instalan los parches de seguridad más recientes por defecto, es decir la opción actualización automática de parches se establece en VERDADERO. La tabla ARP de Windows actúa de forma predeterminada como dinámica, y no ha sido alterada, es decir el atributo *TablaEstaticaARP* es FALSO. La ejecución automática USB está desactivado, entonces el valor del atributo *AutoejecutableUSBdeshabilitado* es VERDADERO en el modelo propuesto.

b. Los Mecanismos de Defensa de un Servidor de Aplicaciones seleccionados

Debido a que la información de la actualización del servidor de aplicaciones es confidencial y no pública, las opciones de actualización en todo el servidor de aplicaciones se establecen en VERDADERO.

c. Los Mecanismos de Defensa de una Aplicación Cliente seleccionados

Toda aplicación que se utiliza en este modelo se encuentra actualizada, por lo tanto, las opciones para parcheado se establecen en VERDADERO.

d. Los Mecanismos de Defensa de Zona de Red seleccionados

Tabla 9 Mecanismos de Defensa de Zona de Red seleccionados.

Zona De Red	Funcionamiento_EvidenciaParaInyectar
WLAN SystemAdmin	
• DNSSec	Verdadero
• PuertoDeSeguridad	Verdadero
INTERNET	
• DNSSec	Verdadero
• PuertoDeSeguridad	Falso

El atributo de defensa DNSSec de la Zona de Red se establece como VERDADERO ya que es muy común hoy en día, y ha llegado a expandirse en algunos dominios como *.com*, *.net* y *.org* [Broderick, J., 2006]. En la red establecida como una WLAN, las direcciones MAC de los ordenadores que componen la red adoptan una dirección IP dinámica, por lo tanto, en la configuración del router TP-LINK ARCHER C2, el *PuertoDeSeguridad* se establece con valor FALSO. Internet en la tabla anterior, representa Internet Publico, es decir es la nube fuera de la puerta de enlace del router TP-LINK ARCHER C2, y el valor del atributo *PuertoDeSeguridad* se establece en FALSO de forma predeterminada.

4.5 Validación

La validación, está compuesta de dos tipos de validaciones:

a. Validación de los ataques comprobables y sus probabilidades predichas:

Debido al límite de la capacidad del autor y del conocimiento profesional, algunos pasos de ataque son demasiado difíciles de implementar, como por ejemplo desarrollar una vulnerabilidad de día cero y encontrar una vulnerabilidad pública no detectable. Los pasos de ataque seleccionados y probados fueron ArpSpoof, Comprometer y Acceso a través de UI en el Sistema Operativo. El paso de ataque Comprometer en Aplicación Cliente, los pasos de ataque Acceso y DnsSpoof en Zona de Red y ArpSpoof en Interfaz de Red.

Los resultados previstos se calculan mediante el método de muestreo de inyección de evidencia directa.

b. Validación de probabilidad de éxito calculada mediante el experimento.

Se implementaron operaciones personalizadas para diferentes pasos de ataque. Más específicamente, el ataque de envenenamiento ARP en el sistema operativo se simula modificando maliciosamente la tabla ARP del sistema operativo y validando la probabilidad a través de la observación de los hechos en la máquina de prueba. Los ataques Comprometer y Acceso a través de UI en el Sistema Operativo y el ataque Comprometer en Aplicación Cliente son fáciles de imitar, ya que el permiso de usuario regular se puede simular. El ataque de envenenamiento DNS en la Zona de Red se simuló mediante la modificación intencional del archivo de host en el Sistema Operativo. El ataque de envenenamiento ARP se hizo cambiando la tabla ARP del enrutador a propósito.

La propuesta del modelo de evaluación de Ciberseguridad analizando los riesgos de los Sistemas de Ciberseguridad Convencionales desde el punto de vista de las vulnerabilidades asociadas, trajo aparejadas tareas y técnicas asociadas han sido validadas en entorno de pruebas que consiste en una WLAN con estaciones de trabajo (maquinas victimas) con el software seleccionado instalado en cada una de ellas, y por otro lado una estación de trabajo fuera de la WLAN (maquina atacante) con las herramientas de hacking seleccionadas instaladas y configuradas para realizar los ataques.

5. Conclusión

El papel de un profesional de seguridad informática es detectar vulnerabilidades y reportarlas a su cliente, pero cabe destacar que, si bien es la mejor alternativa, trae aparejado una serie de tareas para que la contratación se concrete con éxito. Podemos encontrar una serie de inconvenientes, que se mencionan a continuación:

a) Alcance de la contratación y planificación: uno de las dificultades habituales es que el cliente no sabe cuántas, ni cuáles son sus vulnerabilidades de seguridad, por lo tanto, queda sujeto a los reportes del profesional contratado.

b) Costos inesperados: de la mano con el punto anterior, el reporte de vulnerabilidades puede contener vulnerabilidades críticas y es muy común que se recomiende invertir en infraestructura de ciberseguridad.

c) Riesgos de seguridad: en algunos casos, contratar a un profesional de seguridad informática puede implicar el riesgo de compartir datos confidenciales de la empresa.

Por otro lado, un scanner de vulnerabilidades es una herramienta que permite realizar una verificación de seguridad en una red mediante el análisis de los puertos, protocolos, dispositivos en red. Trabaja con una base de datos de firmas de las vulnerabilidades actuales de software que permite la detección. La utilización de estas herramientas,

implican contar con personal capacitado para poder operarlas y la inversión correspondiente para adquirirlas.

En este artículo se ha propuesto utilizar un modelo para el análisis de vulnerabilidades y se ha corroborado que se puede aplicar una teoría de Ciberseguridad a la Arquitectura de Sistema de Ciberseguridad Convencional propuesta, a partir de esta teoría se genera un modelo propuesto utilizando plantillas existentes del Modelo CySeMoL para sistemas SCADA y la herramienta EAAT para el modelado gráfico y carga de parámetros. Luego se configura el modelo en la herramienta CySeMoL basados en el software seleccionado para ser evaluado, consecutivamente se ejecuta la herramienta EAAT que procesa los parámetros cargados y se obtienen resultados que resultan de mucha utilidad para un CISO (Chief Information Security Officer) o responsable de la seguridad informática en una organización, ya que obtiene indicadores de las vulnerabilidades de la arquitectura planteada, es decir nos da un indicio de lo que se debe mejorar, todo sin la necesidad imperiosa de consultar un experto en seguridad. Cabe aclarar que el aporte de este artículo fue validado para la arquitectura planteada exclusivamente, es decir en otras arquitecturas o situaciones de mayor complejidad, es conveniente consultar un experto en seguridad informática.

En este contexto, se han formulado los siguientes pasos para cumplir con el objetivo propuesto:

a. Identificar un modelo que permita aplicar una teoría de ciberseguridad basada en las especificaciones de la arquitectura de los Sistemas de Ciberseguridad Convencionales.
b. Desarrollar una teoría de Ciberseguridad aplicable a la arquitectura de los Sistemas de Ciberseguridad Convencionales.
c. Adaptar e integrar el modelo CySeMol a la teoría desarrollada en el punto anterior.

En conclusión, los resultados experimentales muestran que CySeMoL tiene un alto grado de precisión hacia la evaluación de las vulnerabilidades en el modelo propuesto, se espera un pequeño número de condiciones en las que los resultados previstos son diferentes del hecho experimental. Estos resultados demuestran que es factible acoplar conocimiento sobre pentesting y de seguridad informática en una herramienta que pueda automatizar las evaluaciones producidas. Los resultados obtenidos fueron notablemente coincidentes con los resultados predictivos con los que fueron comparados, por lo tanto, se puede afirmar que el modelo de evaluación propuesto en la tesis en desarrollo, es útil para realizar evaluaciones de ciberseguridad cuando no se cuenta en una empresa de mediana envergadura con un experto en seguridad informática.

6. Referencias

Pakalniškis, S. (2012). What factors explain why there is not a common and comprehensive global response to cyber threats?

Bayuk, J. L., Healey, J., Rohmeyer, P., Sachs, M. H., Schmidt, J., & Weiss, J. (2012). Cyber security policy guidebook. John Wiley & Sons.

Holm, H., Ekstedt, M., Sommestad, T., & Korman, M. (2013). A manual for the cyber security modeling language. Royal Institute of Technology (KTH), Tech. Rep.

Getoor, L., Friedman, N., Koller, D., & Pfeffer, A. (2001). Learning probabilistic relational models. In Relational data mining (pp. 307-335). Springer Berlin Heidelberg.

Breese, J. S., Heckerman, D., & Kadie, C. (1998, July). Empirical analysis of predictive algorithms for collaborative filtering. In Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence (pp. 43-52). Morgan Kaufmann Publishers Inc.

Sommestad, T., Ekstedt, M., & Johnson, P. (2010). A probabilistic relational model for security risk analysis. Computers & Security, 29(6), 659-679.

Liu, S., & Cheng, B. (2009). Cyberattacks: Why, what, who, and how. IT professional, 11(3).

Bishop, M. A. (2002). The art and science of computer security. Addison-Wesley Longman Publishing Co., Inc.

Broderick, J. S. (2006). ISMS, security standards and security regulations. information security technical report, 11(1), 26-31. K.

Lodderstedt, T., Basin, D., & Doser, J. (2002, September). SecureUML: A UML-based modeling language for model-driven security. In International Conference on the Unified Modeling Language (pp. 426-441). Springer, Berlin, Heidelberg.

Trevisani, K. M., & Garcia, R. E. (2008). SPML: A Visual Approach for Modeling Firewall Configurations 1.

Mouratidis, H. (2002). A natural extension of tropos methodology for modelling security.

Jürjens, J. (2005). Secure systems development with UML. Springer Science & Business Media.

Herrmann, D. S. (2002). Using the Common Criteria for IT security evaluation. Auerbach Publications.

Hogganvik, I. (2007). A graphical approach to security risk analysis.

Johnson, P., Ullberg, J., Buschle, M., Franke, U., & Shahzad, K. (2013, March). P 2 AMF: predictive, probabilistic architecture modeling framework. In International IFIP Working Conference on Enterprise Interoperability (pp. 104-117). Springer, Berlin, Heidelberg.

Yazar, Z. (2002). A qualitative risk analysis and management tool–CRAMM. SANS InfoSec Reading Room White Paper, 11, 12-32.

Karabacak, B., & Sogukpinar, I. (2005). ISRAM: information security risk analysis method. Computers & Security, 24(2), 147-159.

Liu, Y., & Man, H. (2005, March). Network vulnerability assessment using Bayesian networks. In Data Mining, Intrusion Detection, Information Assurance, and Data Networks Security 2005 (Vol. 5812, pp. 61-72). International Society for Optics and Photonics.

Statista Estadísticas Cuota de Mercado a Nivel Mundial de los Sistemas Operativos https://es.statista.com/estadisticas/576870/cuota-de-mercado-mundial-de-los-sistemas-operativos/ (sitio vigente al 31/01/2019).

Opswat Windows Anti-malware Market Share Report https://metadefender.opswat.com/reports/anti-malware-market-share#!/?date=2018-11-26 (sitio vigente al 31/01/2019).

NetMarketShare Browser Market Share https://netmarketshare.com/browser-market-share.aspx? (sitio vigente al 31/01/2019).

Whatpulse Statistics Applications https://whatpulse.org/stats/apps/?page=2&orderby=users (sitio vigente al 31/01/2019).

Opswat Microsoft Reigns Supreme in Instant Messaging Market https://www.opswat.com/blog/microsoft-reigns-supreme-instant-messaging-market (sitio vigente al 31/01/2019).