ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE CIENCIAS

EVALUACIÓN DEL RIESGO DE UNA RED DE IOT (INTERNET DE LAS COSAS) EN UNA CASA INTELIGENTE UTILIZANDO REDES BAYESIANAS

TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO MATEMÁTICO

PROYECTO DE INVESTIGACIÓN

DIEGO FERNANDO HEREDIA DÍAZ
diegoheredia31@gmail.com

Director: DR. MIGUEL ALFONSO FLORES SÁNCHEZ
miguel.flores@epn.edu.ec

Codirector: MSC. ROBERTO OMAR ANDRADE PAREDES
roberto.andrade@epn.edu.ec

QUITO, MARZO 2022

DECLARACIÓN

Yo DIEGO FERNANDO HEREDIA DÍAZ, declaro bajo juramento que el trabajo aquí escrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual, correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su reglamento y por la normatividad institucional vigente.

_____

Diego Fernando Heredia Díaz

CERTIFICACIÓN

Certificamos que el presente trabajo fue desarrollado por DIEGO FERNANDO HEREDIA DÍAZ, bajo nuestra supervisión.

Dr. Miguel Alfonso Flores Sánchez
Director del Proyecto

MSc. Roberto Omar Andrade Paredes
Codirector del Proyecto

AGRADECIMIENTOS

A la Escuela Politécnica Nacional y la Facultad de Ciencias por la formación académica a lo largo de la carrera.

A mis profesores por haber compartido sus conocimientos y experiencias. Especialmente agradezco a mi director de tesis Dr. Miguel Flores y codirector MSc. Roberto Andrade, por el tiempo y apoyo brindado en el desarrollo del proyecto de titulación.

A mis padres Ximena y Fernando por apoyarme incondicionalmente en todas las etapas de mi formación académica.

A mi hermana Saraí, tía Margarita y abuelito Ulpiano por sus consejos y palabras de apoyo cuando lo necesitaba.

A mi tío Rafael por sus enseñanzas y consejos.

A mis compañeros Alexander Constante, Gabriel Granda y Esteban Morillo por su amistad y ayuda dentro y fuera de las aulas.

DEDICATORIA

A mis padres Ximena y Fernando

A mi hermana Saraí

Índice general

Índice de figuras

Índice de tablas

Resumen

En el presente trabajo se implementa un modelo de evaluación del riesgo de una red del Internet de las Cosas (IoT) en una casa inteligente utilizando una red bayesiana. El grafo dirigido acíclico de la red bayesiana se obtiene a partir de un grafo de ataque que detalla la estructura de la red de IoT y los caminos por los que pueden ocurrir distintos ataques. Los parámetros de la red bayesiana se estiman con el método de máxima verosimilitud aplicado a un conjunto de datos obtenido a partir de la simulación de ataques a la red, en cinco diferentes escenarios. La evaluación del riesgo se enfoca en el análisis de los ataques de DoS, ataques de MitM y ambos simultáneamente a los dispositivos que permiten la automatización de la casa inteligente y que por lo general son los que individualmente tienen menores niveles de seguridad. Para esto se consideran inferencias en la red bayesiana y el impacto de los ataques utilizando los datos simulados.

Palabras clave: IoT, casa inteligente, red bayesiana, simulación, evaluación de riesgo

Abstract

In the present work, a risk assessment model of an Internet of Things (IoT) network is implemented in a smart home using a Bayesian network. The directed acyclic graph of the Bayesian network is obtained from an attack graph that details the structure of the IoT network and the paths through which different attacks can occur. The parameters of the Bayesian network are estimated with the maximum likelihood method applied to a data set obtained from the simulation of network attacks, in five different scenarios. The risk assessment focuses on the analysis of DoS attacks, MitM attacks and both simultaneously to the devices that allow the automation of the smart home and that are generally the ones that individually have lower levels of security. For this, inferences in the Bayesian network and the impact of the attacks using the simulated data are considered.

Keywords: IoT, smart home, bayesian network, simulation, risk assessment

Tabla de Definiciones

IoT	Internet de las Cosas
CVSS	Common Vulnerability Scoring System. Es un sistema de puntuación que mide el impacto o la gravedad de las vulnerabilidades.
AP	Punto de acceso
M	Nivel de Administración
CP	Celular que se encuentra en el Nivel de Administración
R	Router. Es un dispositivo que permite la interconexión de Internet a otros dispositivos.
F	Nivel de Dispositivos
SE	Ataque de ingeniería social. En este ataque se engaña a las víctimas para obtener información sensible y confidencial.
PH	Ataque de phishing. Es un tipo de ataque de ingeniería social en el que se emplean medios digitales.
MI	Ataque de malware. Este ataque consiste en la inserción de códigos maliciosos al sistema operativo del dispositivo objetivo.
DoS/DOS	Ataque de denegación de servicio. Este ataque altera la conexión de la red o de un dispositivo, con el fin de volverla inaccesible.
RTP	Ataque de routing table poisoning. Este ataque busca la alteración de los datos del router realizando un cambio malicioso.
PA	Ataque persistente. Este ataque inserta datos dañinos continuamente en la red Wi-Fi.
MitM/MITM	Ataque de man in the middle. En este ataque se toma el control de un canal de comunicación entre dos dispositivos con el fin de controlar el tráfico de comunicación entre las víctimas.
SECP	Variable aleatoria que indica si ocurre un ataque de ingeniería social desde el punto de acceso al nivel de administración
PHCP	Variable aleatoria que indica si ocurre un ataque de phishing desde el punto de acceso al nivel de administración
MICP	Variable aleatoria que indica si ocurre un ataque de malware desde el punto de acceso al nivel de administración
DOSR	Variable aleatoria que indica si ocurre un ataque de DoS desde el nivel de administración al router
RTPR	Variable aleatoria que indica si ocurre un ataque de RTP desde el nivel de administración al router
PAR	Variable aleatoria que indica si ocurre un ataque persistente desde el nivel de administración al router

DOSF	Variable aleatoria que indica si ocurre un ataque de DoS desde el router al nivel de dispositivos
MITMF	Variable aleatoria que indica si ocurre un ataque de MitM desde el router al nivel de dispositivos
S_{PHCP}	Variable aleatoria que indica si se selecciona un ataque de phishing al nivel de administración
S_{MICP}	Variable aleatoria que indica si se selecciona un ataque de malware al nivel de administración
S_{DOSR}	Variable aleatoria que indica si se selecciona un ataque de DoS al router
S_{RTPR}	Variable aleatoria que indica si se selecciona un ataque de RTP al router
S_{PAR}	Variable aleatoria que indica si se selecciona un ataque persistente al router
S_{DOSF}	Variable aleatoria que indica si se selecciona un ataque de DoS al nivel de dispositivos
S_{MITMF}	Variable aleatoria que indica si se selecciona un ataque de MitM al nivel de dispositivos
AV	Vector de ataque
AC	Complejidad de ataque
UI	Interacción de usuario
PR	Privilegios requeridos
S	Alcance
C	Impacto a la confidencialidad
I	Impacto a la integridad
A	Impacto a la disponibilidad
CR	Requerimiento de confidencialidad
IR	Requerimiento de integridad
AR	Requerimiento de disponibilidad
RL	Nivel de remediación
V_{SECP}	Variable aleatoria que indica si el atacante tiene éxito al vulnerar el nivel de administración con un ataque de ingeniería social
V_{PHCP}	Variable aleatoria que indica si el atacante tiene éxito al vulnerar el nivel de administración con un ataque de phishing
V_{MICP}	Variable aleatoria que indica si el atacante tiene éxito al vulnerar el nivel de administración con un ataque de malware
V_{DOSR}	Variable aleatoria que indica si el atacante tiene éxito al vulnerar el router con un ataque de DoS
V_{RTPR}	Variable aleatoria que indica si el atacante tiene éxito al vulnerar el router con un ataque de RTP
V_{PAR}	Variable aleatoria que indica si el atacante tiene éxito al vulnerar el router con un ataque persistente
V_{DOSF}	Variable aleatoria que indica si el atacante tiene éxito al vulnerar el nivel de dispositivos con un ataque de DoS
V_{MITMF}	Variable aleatoria que indica si el atacante tiene éxito al vulnerar el nivel de dispositivos con un ataque de MitM

Capítulo 1

Introducción

1.1. Antecedentes

El avance tecnológico a través de los años ha permitido facilitar las actividades de las personas mediante el uso de dispositivos electrónicos. Como parte de este desarrollo, se han creado redes de dispositivos interconectados entre sí a través de Internet capaces de compartir información entre ellos. Estas redes son conocidas como redes del Internet de las Cosas (IoT) y su utilización se encuentra en varios campos como la salud, transporte, educación, telecomunicaciones, agricultura, automatización de casas, ciudades inteligentes, entre otros. Cada fabricante establece seguridades específicas para sus dispositivos, sin embargo, estas seguridades pueden no ser del todo efectivas al estar los dispositivos interconectados entre sí en una red de IoT. Junto con el crecimiento y evolución de estas tecnologías surgen problemas que deben ser abordados y uno de los más importantes es la seguridad, específicamente en lo relacionado con los eventos de ciberataques a los dispositivos. Estos eventos representan un riesgo para las personas involucradas, tanto a nivel individual como colectivo, y pueden causar un gran impacto a nivel económico.

El riesgo se define a través de la probabilidad de un evento indeseable y su nivel de impacto [7]. Las metodologías de análisis de riesgo de seguridad informática tradicionalmente tienen un enfoque determinista y se basan en la priorización de la seguridad de los activos más críticos. Algunos modelos determinísticos que se han desarrollado para la evaluación de ciberataques son los árboles de ataque, redes Petri y grafos de ataque. Los modelos de grafos de ataque permiten una mejor descripción del proceso de ataque a la red, por lo que se han convertido en una de las herramientas más utilizadas para resolver problemas de seguridad en las redes [19].

Por otro lado, considerando modelos probabilísticos, se tiene que en 2005, Liu y Man [10] propusieron un modelo de grafos de ataque basado en redes bayesianas. En los siguientes años se implementaron mejoras al modelo de Liu y Man; por ejemplo, Frigault et.al [5] le añadieron el factor del tiempo estableciendo un modelo dinámico. En estos modelos, los resultados de las probabilidades de ataque se obtienen mediante inferencias sobre las redes bayesianas. Por esta razón, se han desarrollado algoritmos para que los cálculos sean eficientes. Liu y Man [10] utilizaron un algoritmo de eliminación de variables, sin embargo, este algoritmo tiene baja eficiencia computacional, por lo tanto, solo se puede utilizar en redes a pequeña escala [19]. Así, Muñoz-González et.al [12] utilizaron el método del árbol de uniones como una mejora del algoritmo de eliminación de variables. Los experimentos mostraron que este algoritmo es mejor que el algoritmo de eliminación de variables en términos de complejidad temporal y complejidad espacial y es más adecuado para situaciones reales [19].

1.2. Justificación

La evaluación de los riesgos en las redes es un proceso necesario para varias aplicaciones del mundo real. La evaluación de las vulnerabilidades a nivel de red es fundamental para que la administración de una empresa planifique sus estrategias de defensa y elabore el presupuesto adecuado para mantener la seguridad de su red [10]. Asimismo, a nivel general, comprender los riesgos que surgen de los dispositivos de IoT y administrarlos es una necesidad inminente de los profesionales de riesgos de seguridad de IoT [7]. Por lo tanto, el estudio del riesgo cibernético es indispensable para el diseño y utilización de una red de IoT.

El modelo de evaluación de riesgo que se implementa en este trabajo está basado en la teoría de redes bayesianas y la simulación de variables aleatorias discretas. Una red bayesiana es un modelo gráfico probabilístico que estudia la dependencia condicional de un conjunto de variables aleatorias mediante un grafo dirigido acíclico, en el cual los nodos representan las variables aleatorias y las aristas, las dependencias. Las redes bayesianas utilizan relaciones causales para estimar la probabilidad de un evento desconocido en función de los eventos que han ocurrido [19]. Así, se pueden obtener conclusiones acertadas sobre distintos eventos que involucran a las variables. Se utilizan comúnmente en el campo del análisis y razonamiento de la incertidumbre [19]. Los modelos de redes bayesianas se han utilizado en varios campos del conocimiento, por ejemplo, en la salud se han diseñado redes bayesianas

que permiten obtener diagnósticos de enfermedades. Con el propósito de mostrar la aplicación de las redes bayesianas en el área de la ciberseguridad, específicamente en la evaluación de riesgos, se propone diseñar un modelo basado en redes bayesianas para la evaluación del riesgo de una red de IoT en una casa inteligente.

Para este trabajo se plantea que el grafo de la red bayesiana esté basado en un grafo de ataque de una red de IoT de una casa inteligente, esto permite representar la información de las posibles rutas que puede tomar un atacante para cumplir su objetivo. La causalidad se refiere a la relación que existe entre el origen o causa y efecto o consecuencia. Existe una dependencia del estado del efecto con respecto al estado de la causa. Las redes bayesianas consideran la relación causal entre variables aleatorias y esto es similar a los grafos de ataque ya que, dada una arista en un grafo de ataque, el estado del nodo de destino depende del estado de su nodo de origen; es decir, el éxito de un ataque depende del éxito de un ataque al nodo anterior [10], como muestra la figura 1.1. De esta forma se pueden representar grafos de ataque mediante redes bayesianas y partir de esta información para realizar evaluaciones de riesgo. Una de las principales ventajas del uso de redes bayesianas para la evaluación del riesgo en este tipo de redes es la flexibilidad y la facilidad de entrenamiento de los parámetros, que son las distribuciones de probabilidad condicional de las variables; además con las redes bayesianas se puede analizar las vulnerabilidades, identificar los nodos de alto riesgo y predecir el comportamiento de los ataques [19].

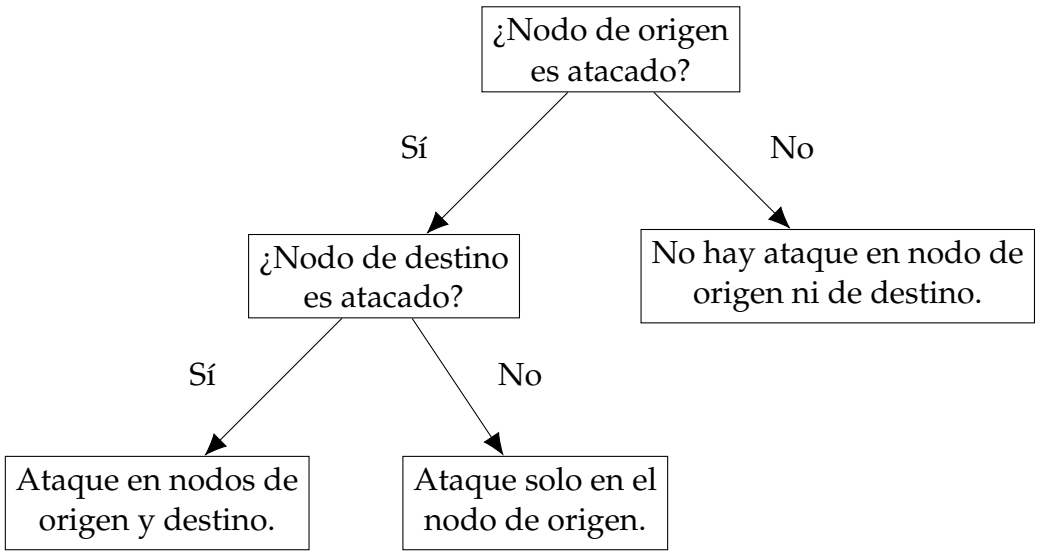


Figura 1.1: Dependencia del éxito en el ataque al nodo de destino con respecto al ataque al nodo de origen en una arista de un grafo de ataque

En el área de ciberseguridad, la falta de datos, especialmente datos históricos sobre brechas, incidentes y amenazas, dificulta el desarrollo de modelos realistas; sin embargo, las redes bayesianas poseen el potencial para abordar este desafío [3]. Específicamente, la capacidad de combinar diferentes fuentes de conocimiento permite que las redes bayesianas sean un modelo adecuado para el estudio de la ciberseguridad. La estimación de los parámetros (distribución de probabilidad en cada nodo) de las redes bayesianas, se puede llevar a cabo mediante la utilización de bases de datos o la opinión de expertos, siendo esta información escasa o de acceso limitado para el público en el caso de las redes de IoT. Frente a esta problemática, para determinar los parámetros se han utilizado los puntajes del Common Vulnerability Scoring System (CVSS) [10, 5, 12] que son valores que estiman el impacto de las vulnerabilidades de los dispositivos en las redes. Sin embargo, estos puntajes no consideran las relaciones de dependencia condicional en los nodos del grafo. Bajo este contexto, debido a la falta de información sobre los ataques a las redes de IoT o la limitación de acceso a la misma, se considera la necesidad de implementar algoritmos matemáticos de simulación, con el fin de obtener conjuntos de datos que permitan describir los ataques en la red de IoT de una casa inteligente y de esta forma aproximar los parámetros de la red bayesiana. También, mediante el cálculo de inferencias en la red bayesiana y la utilización de los datos simulados, se obtiene una evaluación adecuada de los riesgos en la red de IoT. Cabe recalcar que en los modelos de evaluación de riesgo mediante redes bayesianas en ciberseguridad, no se han realizado simulaciones de eventos de ataques para obtener un conjunto de datos y a partir de esto estimar los parámetros de la red bayesiana [3]. De esta forma, el presente trabajo representa un aporte para el estudio de la ciberseguridad mediante redes bayesianas.

1.3. Objetivos

1.3.1. General

Implementar un modelo de evaluación del riesgo en una red de IoT utilizando redes bayesianas cuyos parámetros sean definidos mediante escenarios de simulación, con la finalidad de identificar el nivel de riesgo en la red.

1.3.2. Específicos

- Definir la estructura de las redes de IoT enfocadas en las casas inteligentes, así como las puntuaciones de vulnerabilidad CVSS de sus dispositivos y los posibles ataques que se pueden presentar, con el fin de implementar una red bayesiana.
- Simular eventos de ataques a una red de IoT, basados en las puntaciones CVSS y en distribuciones de variables aleatorias, con el fin de estimar los parámetros de la red bayesiana y dar una evaluación del riesgo.
- Realizar inferencias en la red bayesiana y utilizar técnicas de simulación para definir el riesgo de la red de IoT.

Capítulo 2

Ciberseguridad y el Internet de las Cosas

2.1. Internet de las Cosas (IoT)

El Internet de las Cosas (IoT) es una infraestructura que incluye dispositivos físicos, vehículos modernos, edificios e incluso dispositivos eléctricos esenciales que usamos de manera constante, los cuales están interconectados entre sí a través de Internet para acumular e intercambiar datos entre ellos [15]. Se denomina red de IoT a un entorno de este tipo de dispositivos. En la actualidad, las redes de IoT se han popularizado y se mantienen en constante evolución. Su utilización se encuentra en diferentes verticales como la salud, transporte, educación, telecomunicaciones, agricultura, entre otros. El propósito principal de IoT es mejorar la calidad de vida de las personas reduciendo el estrés y disminuyendo la cantidad de trabajos repetitivos [2]. Una de las aplicaciones actuales de las redes de IoT y que se ha vuelto muy popular es la automatización de casas o casas inteligentes.

2.2. Casas inteligentes

Una casa inteligente se refiere a un sistema compuesto por dispositivos interconectados a través de una red de IoT en una casa. Algunas motivaciones para la utilización de redes de IoT en casas son [2]:

- **Seguridad:** La casa puede ser vigilada a través de cámaras de seguridad. Además, se pueden automatizar alertas y llamadas de emergencia.

- **Eficiencia energética y comodidad:** Los dispositivos de la casa inteligente pueden ser encendidos o apagados según las necesidades, por ejemplo, las luces según la luz del día o el aire acondicionado según la temperatura externa o en horas determinadas. Asimismo, se pueden programar los dispositivos para que se apaguen cuando no están en uso y ahorrar energía.
- **Salud y bienestar:** Se puede realizar seguimiento de los pasos, los latidos del corazón, los niveles de insulina y los patrones de sueño.
- **Ayuda a personas mayores o discapacitadas:** Se puede disponer de ayudas visuales o sonoras para proporcionar alertas de eventos. Las personas pueden hacer uso de un dispositivo portátil que permita alertar a la familia o a las autoridades en caso de emergencia.

2.3. Ciberseguridad en casas inteligentes

Junto con el crecimiento y evolución de estas tecnologías surgen problemas que deben ser abordados y uno de los más importantes es la seguridad. Se conoce que todos los dispositivos conectados a Internet están expuestos a ciberataques y los dispositivos de IoT no son una excepción [7]. En las redes de IoT algunos dispositivos presentan baja seguridad debido a sus condiciones de diseño; por ejemplo, una refrigeradora inteligente no tiene la misma calidad de seguridad que una computadora con protecciones como firewall o antivirus. Además, un ataque a cualquier dispositivo en la red puede generar un impacto severo ya que se puede producir un efecto cascada debido al grado de interconexión que tienen los dispositivos.

2.3.1. Estructura de la red de IoT de una casa inteligente

La red de IoT de una casa inteligente tiene la siguiente estructura [6]:

1. **Punto de Acceso (AP):** Es la puerta de ingreso a la red de IoT.
2. **Nivel de Administración (M):** En este nivel se encuentran los dispositivos que dan órdenes, como iPads, computadoras y celulares.
3. **Router (R):** El router o enrutador es un dispositivo que permite la interconexión de Internet a otros dispositivos.

4. **Nivel de Dispositivos (F):** En este nivel se ubican todos los dispositivos relacionados a la automatización de la casa inteligente, como luces, persianas, cámaras, entre otros.

Además, se tienen tres conexiones inalámbricas Wi-Fi:

1. **Red Wi-Fi 1:** Proporciona conexión entre el punto de acceso y el nivel de administración.
2. **Red Wi-Fi 2:** Proporciona conexión entre el nivel de administración y el router.
3. **Red Wi-Fi 3:** Proporciona conexión entre el router y el nivel de dispositivos.

La figura 2.1 muestra la estructura de la red de IoT de la casa inteligente.

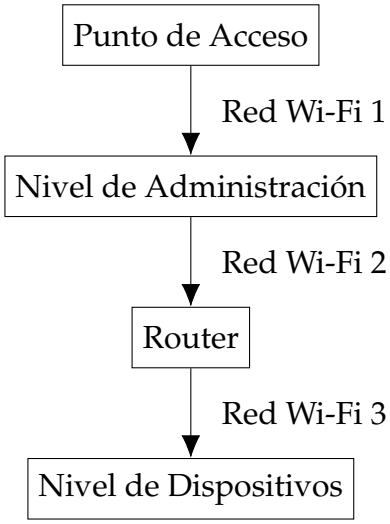


Figura 2.1: Estructura de la red de IoT de la casa inteligente

Los atacantes acceden al nivel de administración desde el punto de acceso a través de la red Wi-Fi 1. Desde ese punto pueden realizar otros ataques para llegar al router a través de la red Wi-Fi 2. Al comprometer el router, los atacantes puede acceder al nivel de dispositivos a través de la red Wi-Fi 3 y atacar a todos los dispositivos conectados.

2.3.2. Ataques a la red de IoT de las casas inteligentes

Los ataques que se pueden presentar en las redes de IoT de las casas inteligentes son los siguientes [6, 17, 9, 11]:

- **Ingeniería Social (SE):** En un ataque de ingeniería social los ingenieros sociales se aprovechan de las víctimas, generalmente engañándolas, para obtener información sensible y confidencial, como datos de tarjetas de crédito o contraseñas.

- **Phishing (PH):** Es un ataque en el que se emplean medios digitales como sitios web falsos, correos electrónicos o anuncios de premios u ofertas para obtener información de las víctimas. En la estructura de la red de IoT de la casa inteligente estos ataques están dirigidos a los dispositivos del nivel de administración con el fin de obtener datos importantes como credenciales de inicio de sesión.

- **Malware (MI):** Este ataque consiste en la inserción de códigos maliciosos al sistema operativo del dispositivo objetivo.

- **Denegación de Servicio (DoS):** Es un ataque que altera la conexión de la red o de un dispositivo, con el fin de volverla inaccesible para los usuarios.

- **Routing Table Poisoning (RTP):** Este ataque se dirige a la tabla de direccionamiento que se encuentra en el router. El ataque busca la alteración de los datos del router realizando un cambio malicioso y drástico en la rutina de esta tabla del router.

- **Ataque Persistente (PA):** Este ataque inserta datos dañinos continuamente en la red Wi-Fi.

- **Man in the Middle (MitM):** Es un ataque en el que se toma el control de un canal de comunicación entre dos dispositivos con el fin de controlar el tráfico de comunicación entre las víctimas sin dejar rastros. Por ejemplo, en una red de IoT, un atacante puede ingresar a un punto de acceso Wi-Fi entre dos dispositivos y obtener acceso de administrador en el dispositivo.

Capítulo 3

Marco Teórico

3.1. Conceptos preliminares

3.1.1. Teoría de grafos

Un grafo es un par $\mathcal{G} = (V, E)$ que consta de un conjunto de nodos V y un conjunto de aristas E que conectan dos elementos del conjunto V. Un par de nodos X_i, X_j pueden estar conectados por una arista dirigida o no dirigida. La diferencia entre ambos tipos de aristas es que las aristas dirigidas son pares ordenados de nodos, por lo que tenemos que $(X_i, X_j) \neq (X_j, X_i)$, a diferencia de las aristas no dirigidas donde $\{X_i, X_j\} = \{X_j, X_i\}$. Decimos que un grafo es dirigido si todas las aristas son dirigidas y no dirigido si todas las aristas son no dirigidas. En la figura 3.1 se muestra un ejemplo de un grafo dirigido y un grafo no dirigido.

(a) Grafo dirigido **(b)** Grafo no dirigido

Figura 3.1: Ejemplo de tipos de grafos

Las redes bayesianas se basan en grafos dirigidos, por lo que los siguientes conceptos se explican únicamente para este tipo de grafos. En un grafo dirigido $\mathcal{G}$, para la arista (X_i, X_j), decimos que X_j es el hijo de X_i en $\mathcal{G}$ y X_i es el padre de X_j en $\mathcal{G}$. Usamos Pa_{X_i} para denotar todos los nodos padres de X_i en $\mathcal{G}$. En la figura 3.2 se muestra un ejemplo de esta notación.

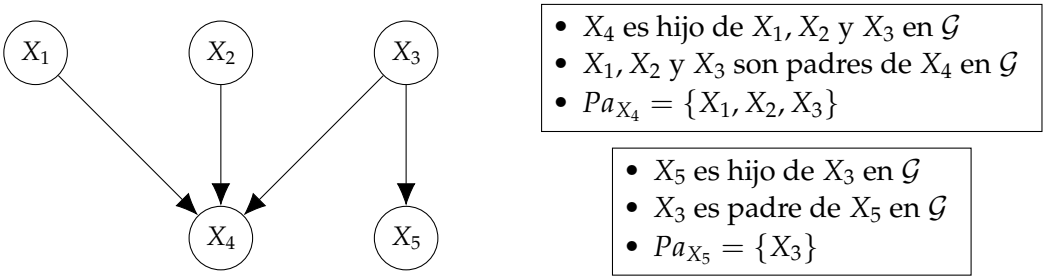


Figura 3.2: Ejemplo de notación de los nodos en grafos dirigidos

Los nodos $X_1, \ldots, X_k$ forman un camino en el grafo dirigido $\mathcal{G} = (V, E)$ si para todo $i = 1, \ldots, k-1$ existe una arista $(X_i, X_{i+1}) \in E$. Un ciclo en $\mathcal{G}$ es un camino $X_1, \ldots, X_k$ donde $X_1 = X_k$. Se muestra un ejemplo de estos conceptos en la figura 3.3. Un grafo es acíclico si no contiene ciclos. Los tipos de grafos que se utilizan para el estudio de redes bayesianas son los grafos dirigidos acíclicos.

(a) Camino **(b)** Ciclo

Figura 3.3: Ejemplo de caminos y ciclos en grafos dirigidos

Se denomina $Descendientes(X_i)$ al conjunto de nodos N tal que existe un camino de X_i a cualquier nodo de N. Por otro lado, $No\_Descendientes(X_i)$ son todos los nodos que no son X_i, Pa_{X_i} o $Descendientes(X_i)$ [4]. En la figura 3.4 se muestra un ejemplo de esta notación.

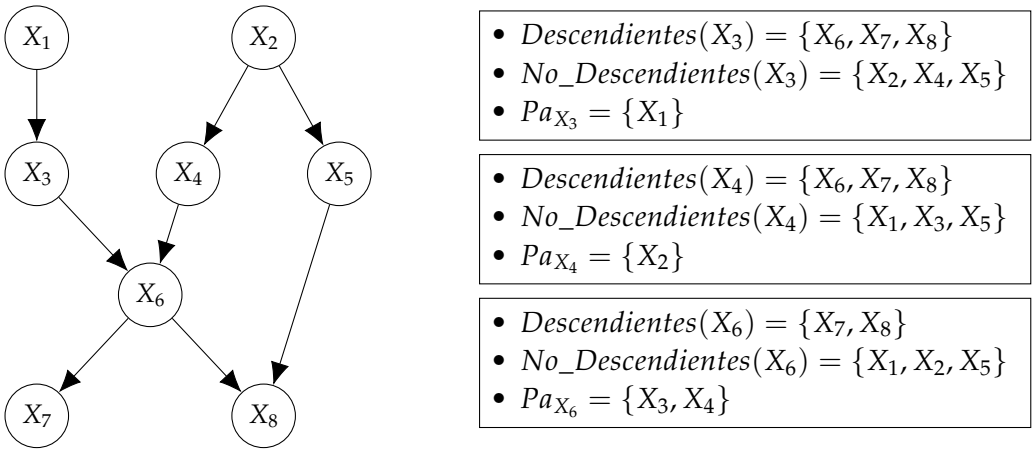


Figura 3.4: Ejemplo de notación de los nodos en grafos dirigidos

Para la representación de los grafos dirigidos acíclicos, los nodos pueden estar en cualquier orden; siempre que se especifiquen todas las aristas, se puede entender la estructura del grafo. Sin embargo, existe un ordenamiento para los nodos que se utiliza en algunos algoritmos y en la teoría de las redes bayesianas y se denomina ordenamiento topológico. Sea $\mathcal{G} = (V, E)$ un grafo dirigido acíclico con $V = \{X_1, \ldots, X_n\}$, un ordenamiento de los nodos $X_{(1)}, \ldots, X_{(n)}$ es un ordenamiento topológico relativo a $\mathcal{G}$ si para toda arista $(X_{(i)}, X_{(j)}) \in E$, se tiene que $i < j$ [8]. En la figura 3.5 se muestra un ejemplo del ordenamiento topológico de un grafo dirigido acíclico.

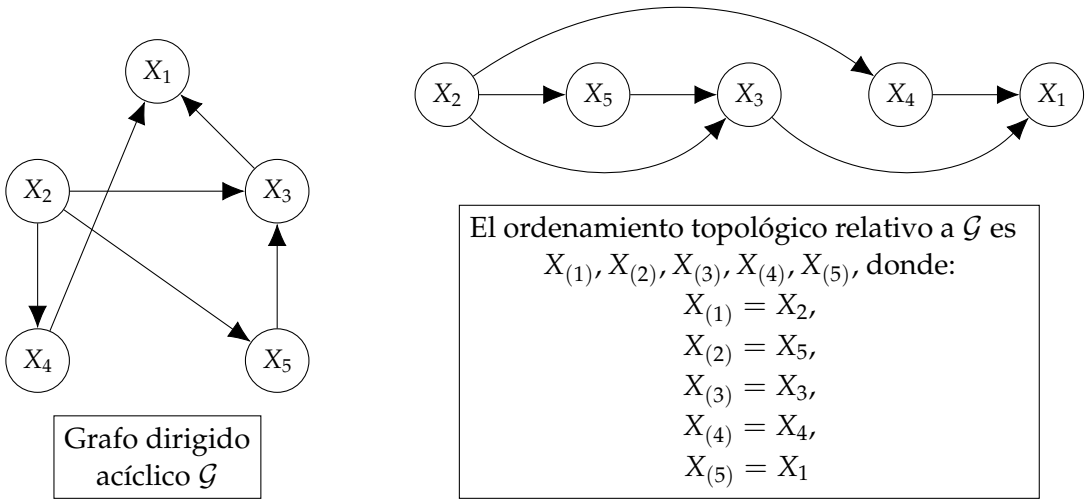


Figura 3.5: Ejemplo de ordenamiento topológico de un grafo dirigido acíclico

3.1.2. Variables aleatorias discretas

Un espacio muestral Ω es el conjunto de todos los posibles resultados de un experimento aleatorio. Una variable aleatoria X es una función de un espacio muestral Ω en $\mathbb{R}$. Usamos $Val(X)$ para denotar el conjunto de valores que puede tomar la variable X. Existen dos tipos de variables aleatorias, discretas y continuas. Una variable aleatoria discreta es aquella que toma un número contable de valores, por ejemplo, $Val(X) = \{0, 1, 2\}$ si X es el número de caras al lanzar una moneda dos veces; mientras que una variable aleatoria continua es aquella que toma valores en un intervalo real, por ejemplo, $Val(X) = [0, 3]$ si X es la cantidad de líquido en litros que hay en una botella. Para el desarrollo de este trabajo se utilizan variables aleatorias discretas para representar el estado de un dispositivo en la red y los posibles estados forman un conjunto contable de valores, específicamente no atacado y atacado, lo que se denota por 0 y 1, respectivamente.

Se denomina función de probabilidad de una variable aleatoria discreta a las probabilidades de los valores que puede tomar la variable. Considerando que la variable X pueda tomar los valores $x_1, \ldots, x_r$, entonces la función de probabilidad es $P(X = x_i) = p_i$ para todo $i \in \{1, \ldots, r\}$; donde debe cumplirse que $p_i \geq 0$ para todo $i \in \{1, \ldots, r\}$ y $\sum_{i=1}^{r} p_i = 1$.

Un vector aleatorio es un vector formado por variables aleatorias. Sea $\mathcal{X} = (X_1, \ldots, X_n)$ un vector aleatorio, $Val(\mathcal{X}) = Val(X_1) \times \cdots \times Val(X_n)$. La función de probabilidad conjunta de un vector aleatorio discreto $\mathcal{X}$ son las probabilidades $P(X_1 = x_1, \ldots, X_n = x_n) = p_{x_1,\ldots,x_n}$, para todos los valores $x_i \in Val(X_i)$, con $i \in \{1, \ldots, n\}$; donde, al igual que en el caso de una variable, debe cumplirse que las probabilidades sean mayores o iguales a 0 y que la suma de estas probabilidades sea 1. La función de probabilidad marginal de una variable X_k del vector $\mathcal{X}$ es su función de probabilidad y puede ser calculada con la conjunta de la siguiente manera:

$$P(X_k = x_k) = \sum_{x_1,\ldots,x_{k-1},x_{k+1},\ldots,x_n} P(X_1 = x_1, \ldots, X_k = x_k, \ldots, X_n = x_n)$$

es decir, para cada valor x_k de X_k, se suman las probabilidades sobre todos los valores de $X_1, \ldots, X_{k-1}, X_{k+1}, \ldots, X_n$.

La función de probabilidad condicional de $X_1, \ldots, X_{k-1}, X_{k+1}, \ldots, X_n$ dado que $X_k = x_k$ son las probabilidades

$$P(X_1 = x_1, \ldots, X_{k-1} = x_{k-1}, X_{k+1} = x_{k+1}, \ldots, X_n = x_n \mid X_k = x_k) \tag{3.1}$$

para todos los valores $x_i \in Val(X_i)$, con $i \in \{1, \ldots, k-1, k+1, \ldots, n\}$. Mediante las propiedades de probabilidad condicional se tiene que (3.1) es igual a:

$$\frac{P(X_1 = x_1, \ldots, X_k = x_k, \ldots, X_n = x_n)}{P(X_k = x_k)}$$

donde el numerador es la función de probabilidad conjunta de $\mathcal{X}$ y el denominador es la función de probabilidad marginal de X_k evaluada en x_k.

Dos conceptos importantes para el estudio de redes bayesianas son la independencia e independencia condicional de variables aleatorias. Sean X,Y dos variables aleatorias, se dice que X es independiente de Y si

$$P(X = x, Y = y) = P(X = x) \cdot P(Y = y)$$

o equivalentemente,

$$P(X = x \mid Y = y) = P(X = x)$$

para todos los valores $x \in Val(X)$, $y \in Val(Y)$ y se denota por $(X \perp Y)$. Sea Z otra variable aleatoria, se dice que X es condicionalmente independiente de Y dado Z si

$$P(X = x, Y = y \mid Z = z) = P(X = x \mid Z = z) \cdot P(Y = y \mid Z = z)$$

o equivalentemente,

$$P(X = x \mid Y = y, Z = z) = P(X = x \mid Z = z)$$

para todos los valores $x \in Val(X)$, $y \in Val(Y)$ y $z \in Val(Z)$ y se denota por $(X \perp Y \mid Z)$.

Sean X, Y, Z y W variables aleatorias, algunas de las principales propiedades de la independencia condicional son las siguientes:

- **Simetría:** $(X \perp Y \mid Z) \Rightarrow (Y \perp X \mid Z)$
- **Descomposición:** $(X \perp Y, W \mid Z) \Rightarrow (X \perp Y \mid Z)$
- **Unión Débil:** $(X \perp Y, W \mid Z) \Rightarrow (X \perp Y \mid Z, W)$
- **Contracción:** $(X \perp W \mid Z, Y)$ y $(X \perp Y \mid Z) \Rightarrow (X \perp Y, W \mid Z)$

3.2. Redes bayesianas

Las redes bayesianas son un tipo de modelo gráfico probabilístico que permiten razonar sobre eventos de variables aleatorias mediante una especificación gráfica de las mismas. Para un conjunto de variables aleatorias, una red bayesiana representa su distribución de probabilidad conjunta de forma compacta aprovechando las independencias condicionales presentes [4, 8]. Formalmente las redes bayesianas se definen de la siguiente manera:

DEFINICIÓN 3.1. *(Red Bayesiana) Una red bayesiana para un vector aleatorio* $\mathcal{X} = (X_1, \ldots, X_n)$ *con* $n \in \mathbb{N}$*, es un par* $(\mathcal{G}, \theta)$*, donde:*

- $\mathcal{G}$ *es un grafo dirigido acíclico denominado estructura de red. En este grafo, cada nodo representa una variable de* $\mathcal{X}$*.*

- $\theta = \{\theta_{X_1|Pa_{X_1}}, \ldots, \theta_{X_n|Pa_{X_n}}\}$ *es un conjunto de funciones de probabilidad condicional, denominado parametrización de la red.*

Se utiliza la notación $\theta_{X_i|Pa_{X_i}}$ para representar las funciones de probabilidad condicional de X_i dado que $Pa_{X_i} = pa_{X_i}$, para todo $pa_{X_i} \in Val(Pa_{X_i})$.

OBSERVACIÓN. Los nodos del grafo $\mathcal{G}$ en una red bayesiana representan cada una de las variables de $\mathcal{X}$ y las aristas representan las dependencias condicionales entre dichas variables.

Según el tipo de variable aleatoria, se distinguen tres tipos de redes bayesianas: discretas si las variables son discretas, continuas si las variables son continuas e híbridas si se tienen variables tanto discretas como continuas. Para este trabajo se consideran redes bayesianas discretas.

OBSERVACIÓN. En una red bayesiana discreta, para toda variable $X_i \in \mathcal{X}$ y para todos los valores pa_{X_i} de sus padres Pa_{X_i}, se cumple que:

$$P(X_i = x_i \mid pa_{X_i}) \geq 0, \forall x_i \in Val(X_i) \quad y \sum_{x_i \in Val(X_i)} P(X_i = x_i \mid pa_{X_i}) = 1$$

debido a que por cada pa_{X_i} se tiene una función de probabilidad condicional.

DEFINICIÓN 3.2. *(Independencias Locales) En una red bayesiana, el grafo dirigido acíclico $\mathcal{G}$ codifica el siguiente conjunto de supuestos de independencia condicional local llamado independencias locales:*

$$\forall X_i \in \mathcal{X} : (X_i \perp No\_Descendientes(X_i) \mid Pa_{X_i})$$

En otras palabras, las independencias locales establecen que cada nodo X_i es condicionalmente independiente de sus no descendientes dados sus padres en el grafo $\mathcal{G}$.

TEOREMA 3.1. *(Regla de la Cadena para Redes Bayesianas) Dada una red bayesiana para un vector aleatorio $\mathcal{X} = (X_1, \ldots, X_n)$, se tiene que:*

$$P(X_1, ..., X_n) = \prod_{i=1}^{n} P(X_i \mid Pa_{X_i})$$

Demostración. Sea $\mathcal{X} = (X_1, \ldots, X_n)$ un vector aleatorio y $\mathcal{B} = (\mathcal{G}, \theta)$ una red bayesiana sobre este vector. Sea $X_{(1)}, \ldots, X_{(n)}$ el ordenamiento topológico de las variables

de $\mathcal{X}$ en relación al grafo $\mathcal{G}$, se tiene que:

$$\begin{aligned} P(X_1, \ldots, X_n) =& P(X_{(1)}, \ldots, X_{(n)}) \\ =& P(X_{(n)} \mid X_{(n-1)}, \ldots, X_{(1)}) \cdot P(X_{(n-1)} \mid X_{(n-2)}, \ldots, X_{(1)}) \cdots \\ & P(X_{(2)} \mid X_{(1)}) \cdot P(X_{(1)}) \\ =& \prod_{i=1}^{n} P(X_{(i)} \mid X_{(1)}, \ldots, X_{(i-1)}) \end{aligned} \tag{3.2}$$

Considerando uno de los factores $P(X_{(i)} \mid X_{(1)}, \ldots, X_{(i-1)})$ de (3.2), debido al orden topológico en $\mathcal{G}$ se tiene que $Pa_{X_{(i)}} \subseteq \{X_{(1)}, \ldots, X_{(i-1)}\}$; además, ninguno de los descendientes de $X_{(i)}$ está en el conjunto $\{X_{(1)}, \ldots, X_{(i-1)}\}$. Por lo tanto:

$$\{X_{(1)}, \ldots, X_{(i-1)}\} = Z \cup Pa_{X_{(i)}} \tag{3.3}$$

donde $Z \subseteq No\_Descendientes(X_{(i)})$. Por las independencias locales y por la propiedad de independencia condicional de descomposición, se tiene que:

$$(X_{(i)} \perp Z \mid Pa_{X_{(i)}}) \tag{3.4}$$

Por lo tanto, de (3.3) y (3.4):

$$P(X_{(i)} \mid X_{(1)}, \ldots, X_{(i-1)}) = P(X_{(i)} \mid Z \cup Pa_{X_{(i)}}) = P(X_{(i)} \mid Pa_{X_{(i)}})$$

Aplicando este razonamiento a todos los factores de (3.2) se sigue que:

$$P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i \mid Pa_{X_i})$$

que es lo que queríamos demostrar. □

3.2.1. Aprendizaje de parámetros

En la teoría de redes bayesianas, la selección y estimación de modelos se conocen como aprendizaje; y esto se lleva a cabo en dos pasos: el aprendizaje de la estructura de la red bayesiana, es decir, el grafo dirigido acíclico; y el aprendizaje de los parámetros, que son las distribuciones de probabilidad condicional que corresponden a la estructura del grafo [18]. Para el aprendizaje de parámetros, utilizamos el método de máxima verosimilitud para un conjunto de datos completo.

El método de máxima verosimilitud consiste en encontrar los argumentos que maximizan la función de verosimilitud, es decir, dado un conjunto de datos $\mathcal{D}$, los

parámetros $\hat{\theta}$ tales que:

$$L(\hat{\theta} : \mathcal{D}) = \max_{\theta \in \Theta} L(\theta : \mathcal{D})$$

En otras palabras, este método permite encontrar estimadores de los parámetros de un modelo, de modo que estos maximicen la probabilidad de observar el conjunto de datos dado [4].

A continuación, se presenta un ejemplo de la estimación por máxima verosimilitud de los parámetros de una variable aleatoria discreta. Sea X una variable aleatoria discreta que puede tomar valores $x_1, \ldots, x_K$ y sea $\theta = (\theta_1, \ldots, \theta_K) \in [0,1]^K$ tal que $P(X = x_i) = \theta_i$ para todo $i \in \{1, \ldots, K\}$ y $\sum_{i=1}^{K} \theta_i = 1$. El espacio de parámetros de este modelo es $\Theta = \left\{\theta \in [0,1]^K : \sum_{i=i}^{K} \theta_i = 1\right\}$. Si $M[i]$ es el número de veces que el valor x_i aparece en el conjunto de datos, entonces la función de verosimilitud se puede escribir de la siguiente forma:

$$L(\theta : \mathcal{D}) = \prod_{i=1}^{K} \theta_i^{M[i]}$$

Debido a que la función logaritmo es monótonamente creciente, maximizar la función de verosimilitud es equivalente a maximizar la función de log-verosimilitud, la cual es:

$$l(\theta : \mathcal{D}) = \sum_{i=1}^{K} M[i] \cdot log(\theta_i)$$

Así, se debe maximizar la función $l(\theta : \mathcal{D})$, sujeto a la restricción $\sum_{i=1}^{K} \theta_i = 1$. Utilizando el método de multiplicadores de Lagrange, se define:

$$F(\theta_1, \ldots, \theta_K, \lambda) = \sum_{i=1}^{K} M[i] \cdot log(\theta_i) - \lambda \left(\sum_{i=1}^{K} \theta_i - 1 \right)$$

El gradiente de la función es:

$$\begin{aligned} \nabla F(\theta_1, \ldots, \theta_K, \lambda) &= \left(\frac{\partial F}{\partial \theta_1}, \frac{\partial F}{\partial \theta_2}, \ldots, \frac{\partial F}{\partial \theta_K}, \frac{\partial F}{\partial \lambda} \right) \\ &= \left(\frac{M[1]}{\theta_1} - \lambda, \frac{M[2]}{\theta_2} - \lambda, \ldots, \frac{M[K]}{\theta_K} - \lambda, 1 - \sum_{i=1}^{K} \theta_i \right) \end{aligned}$$

Igualando el gradiente al vector 0, se obtiene el sistema de ecuaciones:

$$\begin{cases} \theta_1 &= \frac{M[1]}{\lambda} \\ \theta_2 &= \frac{M[2]}{\lambda} \\ &\vdots \\ \theta_K &= \frac{M[K]}{\lambda} \\ \sum_{i=1}^{K} \theta_i &= 1 \end{cases}$$

Reemplazando las K primeras ecuaciones en la ecuación $K+1$, se tiene que:

$$\lambda = \sum_{i=1}^{K} M[i]$$

Finalmente, reemplazando λ en las K primeras ecuaciones del sistema:

$$\hat{\theta}_j = \frac{M[j]}{\sum_{i=1}^{K} M[i]}, \qquad \forall j \in \{1, \ldots, K\}$$

De esta forma, $\hat{\theta}_j$ son los estimadores de máxima verosimilitud de los parámetros de la distribución de la variable X. En otras palabras, la probabilidad de cada valor de X corresponde a su frecuencia en el conjunto de datos [8].

Ahora el objetivo es estimar los parámetros de una red bayesiana de N variables $\{X_1, \ldots, X_N\}$ con estructura $\mathcal{G}$ y parámetros θ. La estructura de la red bayesiana nos permite reducir el problema de estimación de parámetros a un conjunto de problemas no relacionados [8]. Sea $\mathcal{D} = \{\xi[1], \ldots, \xi[M]\}$ un conjunto de datos completo de las variables $\{X_1, \ldots, X_N\}$; donde $\xi[i]$ es una N-tupla de valores de cada variable; es decir, $\xi[i] = (x_1[i], \ldots, x_N[i]) \in Val(X_1) \times \cdots \times Val(X_N)$, $\forall i \in \{1, \ldots, N\}$. La función de verosimilitud es:

$$\begin{aligned} L(\theta : \mathcal{D}) &= \prod_{m=1}^{M} P_{\mathcal{G}}(\xi[m] : \theta) \\ &= \prod_{m=1}^{M} \prod_{i=1}^{N} P(x_i[m] \mid pa_{X_i}[m] : \theta) \\ &= \prod_{i=1}^{N} \left[\prod_{m=1}^{M} P(x_i[m] \mid pa_{X_i}[m] : \theta) \right] \end{aligned} \tag{3.5}$$

donde la segunda igualdad se obtiene de la regla de la cadena para redes bayesianas. Cada uno de los términos entre corchetes de (3.5) se refiere a la función de verosimilitud condicional de una variable dados sus padres en el grafo. Usamos $\theta_{X_i|Pa_{X_i}}$ para denotar el subconjunto de parámetros que determina $P(X_i \mid Pa_{X_i})$ en nuestro

modelo. De esta forma se puede escribir:

$$L(\theta : \mathcal{D}) = \prod_{i=1}^{N} L_i(\theta_{X_i|Pa_{X_i}} : \mathcal{D}),$$

donde la función de verosimilitud local para cada X_i es:

$$L_i(\theta_{X_i|Pa_{X_i}} : \mathcal{D}) = \prod_{m=1}^{M} P(x_i[m] \mid pa_{X_i}[m] : \theta_{X_i|Pa_{X_i}})$$

Tomando en cuenta que los parámetros son disjuntos, es decir, que cada distribución está parametrizada por un conjunto de parámetros separado que no se superpone; este análisis muestra que la verosimilitud se descompone como el producto de términos independientes. Se tiene la siguiente proposición [8].

PROPOSICIÓN 3.2. *Sea $\mathcal{D}$ un conjunto de datos completo para $X_1, \ldots, X_N$, sea $\mathcal{G}$ la estructura de la red sobre estas variables y suponiendo que los parámetros $\theta_{X_i|Pa_{X_i}}$ son disjuntos de $\theta_{X_j|Pa_{X_j}}$ para todo $j \neq i$. Sean $\hat{\theta}_{X_i|Pa_{X_i}}$ los parámetros que maximizan a $L_i(\theta_{X_i|Pa_{X_i}} : \mathcal{D})$. Entonces $\hat{\theta} = \{\hat{\theta}_{X_1|Pa_{X_1}}, \ldots, \hat{\theta}_{X_N|Pa_{X_N}}\}$ maximiza a $L(\theta : \mathcal{D})$.*

En otras palabras, podemos maximizar cada función de verosimilitud local independientemente del resto de la red y luego combinar las soluciones para obtener los estimadores de máxima verosimilitud del conjunto de parámetros de la red bayesiana.

Ahora, considerando una red bayesiana discreta, que es el caso de este trabajo; supongamos que en una red bayesiana se tiene la variable X y sus padres U, entonces se tiene un parámetro $\theta_{x|u}$ para cada combinación de los posbiles valores de las variables, es decir, $x \in Val(X)$ y $u \in Val(U)$. En este caso, se puede escribir la función de verosimilitud local como:

$$\begin{aligned} L_X(\theta_{X|U} : D) &= \prod_{m=1}^{M} \theta_{x[m]|u[m]} \\ &= \prod_{u \in Val(U)} \left[\prod_{x \in Val(X)} \theta_{x|u}^{M[u,x]} \right] \end{aligned} \tag{3.6}$$

donde $M[u, x]$ es el número de veces que $x[m] = x$ y $u[m] = u$ en $\mathcal{D}$. Debido a que la elección de los parámetros dados diferentes valores u de U son independientes entre sí, entonces podemos maximizar los términos entre corchetes de (3.6), independientemente; sujeto a que $\sum_{x \in Val(X)} \theta_{x|u} = 1$ para cada u, debido a que son funciones de probabilidad condicional. Notemos que si se fija un $u \in U$, el término entre corche-

tes de (3.6) es la función de verosimilitud de una variable aleatoria discreta, donde el conteo en los datos de los diferentes valores de x es $\{M[u,x] : x \in Val(X)\}$. Utilizando el razonamiento del ejemplo desarrollado, se puede afirmar que los estimadores de máxima verosmilitud son:

$$\hat{\theta}_{x|u} = \frac{M[u,x]}{\sum_{x \in Val(X)} M[u,x]} = \frac{M[u,x]}{M[u]}$$

Por lo tanto, los estimadores de máxima verosimilitud de los parámetros de una red bayesiana con variables aleatorias discretas $\{X_1, \ldots, X_N\}$, son:

$$\hat{\theta}_{x_i|pa_{X_i}} = \frac{M[x_i, pa_{X_i}]}{M[pa_{X_i}]}$$

para todo $x_i \in Val(X_i)$ y $pa_{X_i} \in Val(Pa_{X_i})$ con $i \in \{1, \ldots, N\}$.

3.2.2. Inferencia

Las redes bayesianas, al igual que otros modelos estadísticos, pueden utilizarse para responder preguntas sobre la naturaleza de los datos que van más allá de la descripción del comportamiento de la muestra observada y las técnicas utilizadas para obtener esas respuestas se conocen en general como inferencia [18]. El tipo de inferencia más común es el cálculo de probabilidades condicionales dada una evidencia, es decir, sea Y una variable de la red bayesiana, $E \subset \mathcal{X} - Y$ un subconjunto de variables y $e \in Val(E)$, se busca calcular $P(Y \mid E = e)$, que es la distribución de probabilidad posterior sobre los valores y de Y, condicionada a que $E = e$ [8]. Otras consultas que se realizan en el ámbito de las redes bayesianas son las probabilidades de una variable o conjunto de variables, por ejemplo, $P(Y = y)$ para todos los valores y de Y; y la explicación más probable, que consiste en encontrar la asignación más probable de todas las variables que no son evidencia, por ejemplo, si $W = \mathcal{X} - E$, el objetivo es encontrar la asignación w' de las variables de W tal que $P(W = w' \mid E = e)$ es máximo.

En general, los cálculos de las probabilidades mencionadas pueden ser desarrollados mediante una lectura completa y correcta de la información probabilística codificada por la red bayesiana seguida de la aplicación repetida de suficientes leyes de la teoría de la probabilidad, como el teorema de Bayes. Sin embargo, el número de posibles aplicaciones de estas leyes puede llegar a ser un problema incluso en redes de pequeña escala. Por lo tanto, el objetivo de los algoritmos de inferencia es automatizar el proceso de razonamiento probabilístico mientras tratan de usar

la menor cantidad de recursos computacionales. Esta automatización no solo sirve para facilitar el proceso de razonamiento para el usuario, sino que en ciertos casos, como redes bayesianas de miles de variables, el razonamiento automatizado puede ser el único método factible para resolver los problemas de inferencia [4].

Los algoritmos para el cálculo de inferencias se clasifican en exactos y aproximados. Los algoritmos exactos se basan en la aplicación repetida del teorema de Bayes junto con cálculos locales y debido a su naturaleza son factibles para redes pequeñas [18]. Algunos ejemplos de algoritmos exactos son:

- **Eliminación de variables:** Este algoritmo utiliza la estructura de la red bayesiana directamente, especificando la secuencia óptima de operaciones en las distribuciones locales y cómo reutilizar los resultados intermedios para evitar cálculos innecesarios.
- **Árbol de uniones:** Este algoritmo transforma el grafo dirigido acíclico en una estructura llamada árbol de uniones o en inglés, junction tree. De esta forma se pueden calcular las inferencias de manera eficiente mediante otros algoritmos especializados para estas estructuras. Muñoz, et al. muestran en [12] que este algoritmo presenta mejoras en cuanto al tiempo empleado y la memoria que ocupa, con respecto al algoritmo de eliminación de variables.

Por otro lado, los algoritmos aproximados se basan en la utilización de simulaciones Monte Carlo para tomar muestras de la distribución global de la red y así estimar inferencias [18]. Algunos ejemplos de algoritmos aproximados son:

- **Logic Sampling:** Este algoritmo utiliza el ordenamiento topológico de las variables de la red para simular de forma eficiente valores de las distribuciones condicionales. Luego aproxima la probabilidad condicional, mediante el conteo del número de casos donde se encuentran las variables y la evidencia dividido para el número casos donde solo se encuentra la evidencia.
- **Likelihood Weighting:** Es una mejora del algoritmo logic sampling debido a que añade un peso que permite utilizar la evidencia en cada paso de la simulación.

Se garantiza que los algoritmos exactos arrojan respuestas correctas y tienden a ser más exigentes desde el punto de vista computacional. Por otro lado, los algoritmos aproximados relajan la insistencia en las respuestas exactas para aliviar las demandas computacionales [4].

3.3. Simulación

El componente básico de un estudio de simulación es la capacidad de generar números aleatorios. Se pueden generar números aleatorios mediante experimentos como el lanzamiento de un dado o una ruleta. Un ejemplo más complejo es el de RANDOM.ORG que genera números cuya aleatoriedad proviene del ruido atmosférico. La desventaja de estos métodos es que no son prácticos para estudios donde se requieren miles de números aleatorios. En este trabajo, para la simulación, un número aleatorio representa el valor de una variable aleatoria uniformemente distribuida en el intervalo $(0,1)$. Para simular valores de esta distribución, el método más conveniente es utilizar una computadora para generar una sucesión de números, los cuales, aunque se generan de manera determinista, tienen toda la apariencia de ser realizaciones de variables aleatorias independientes uniformemente distribuidas en el intervalo $(0,1)$ [16]. Estos números son obtenidos mediante algoritmos llamados generadores de números pseudoaleatorios los cuales parten de un valor inicial llamado semilla y mediante funciones transforman esta semilla en los valores deseados. Los métodos se diferencian por la elección de la semilla y las funciones que se utilizan. Las principales ventajas de los algoritmos que se han desarrollado para generar números pseudoaleatorios son la rapidez, comodidad, reproducibilidad y portabilidad. Además, estos algoritmos han sido implementados en los diferentes lenguajes de programación para facilitar la operatividad al usuario.

En base a la generación de números aleatorios con distribución uniforme se han desarrollado métodos para simular valores de variables con diferentes distribuciones de probabilidad y uno de ellos es el método de inversión.

A continuación se describe el proceso para simular valores de una variable aleatoria discreta utilizando el método de inversión [16]. Sea X una variable aleatoria discreta que puede tomar los valores $x_0, x_1, \ldots, x_M$ y tiene una función de probabilidad:

$$P(X = x_j) = p_j, \quad j = 0, 1, \ldots, M; \qquad \sum_{j=0}^{M} p_j = 1$$

Para simular un valor de dicha variable aleatoria, se genera un número aleatorio U que esté uniformemente distribuido en el intervalo $(0,1)$ y se fija:

$$X = \begin{cases} x_0 & \text{Si } U < p_0 \\ x_1 & \text{Si } p_0 \leq U < p_0 + p_1 \\ \vdots & \\ x_j & \text{Si } \sum_{i=0}^{j-1} p_i \leq U < \sum_{i=0}^{j} p_i \\ \vdots & \\ x_M & \text{Si } U \geq \sum_{i=0}^{M-1} p_i \end{cases}$$

Demostración. Notemos que, para $0 < a < b < 1$, $P(a \leq U < b) = b - a$, entonces, si $j \in \{1, ..., M-1\}$ se tiene que:

$$P(X = x_j) = P\left(\sum_{i=0}^{j-1} p_i \leq U < \sum_{i=0}^{j} p_i\right) = \sum_{i=0}^{j} p_i - \sum_{i=0}^{j-1} p_i = p_j$$

Además, para x_0 y x_M, se tiene que:

$$P(X = x_0) = P(U < p_0) = p_0$$
$$P(X = x_M) = P\left(U \geq \sum_{i=0}^{M-1} p_i\right) = 1 - P\left(U < \sum_{i=0}^{M-1} p_i\right) = 1 - \sum_{i=0}^{M-1} p_i = p_M$$

Por lo tanto, X tiene la distribución deseada. □

En este trabajo, el enfoque de la simulación se encuentra en las variables aleatorias discretas con distribución Bernoulli. Para simular un valor de una variable aleatoria discreta X que sigue una distribución Bernoulli de parámetro p, es decir, tal que para $0 < p < 1$:

$$P(X = x) = \begin{cases} 1 - p & \text{Si } x = 0 \\ p & \text{Si } x = 1 \end{cases}$$

se genera un número aleatorio U que esté uniformemente distribuido en el intervalo $(0, 1)$ y se fija:

$$X = \begin{cases} 0 & \text{Si } U < 1 - p \\ 1 & \text{Si } U \geq 1 - p \end{cases}$$

Para el proceso de simulación de ataques también se simulan vectores aleatorios de dos y tres variables con distribución Bernoulli. Se presenta el procedimiento para generar un valor de un vector con dos variables Bernoulli. Sean X, Y variables aleatorias discretas tales que:

$$P(X=i,Y=j)=p_{ij},\quad i,j\in\{0,1\};\qquad \sum_{i=0}^{1}\sum_{j=0}^{1}p_{ij}=1$$

Para generar un valor de dicho vector aleatorio, se genera un número aleatorio U que esté uniformemente distribuido en el intervalo $(0,1)$ y se fija:

$$(X,Y)=\begin{cases}(0,0) & \text{Si } U<p_{00}\\ (0,1) & \text{Si } p_{00}\leq U<p_{00}+p_{01}\\ (1,0) & \text{Si } p_{00}+p_{01}\leq U<p_{00}+p_{01}+p_{10}\\ (1,1) & \text{Si } U\geq p_{00}+p_{01}+p_{10}\end{cases}$$

Ahora se presenta el procedimiento para generar un valor de un vector con tres variables Bernoulli. Sean X,Y,Z variables aleatorias discretas tales que:

$$P(X=i,Y=j,Z=k)=p_{ijk},\quad i,j,k\in\{0,1\};\qquad \sum_{i=0}^{1}\sum_{j=0}^{1}\sum_{k=0}^{1}p_{ijk}=1$$

Para generar un valor de dicho vector aleatorio, se genera un número aleatorio U que esté uniformemente distribuido en el intervalo $(0,1)$ y se fija:

$$(X,Y,Z)=\begin{cases}(0,0,0) & \text{Si } U<p_{000}\\ (0,0,1) & \text{Si } p_{000}\leq U<p_{000}+p_{001}\\ (0,1,0) & \text{Si } p_{000}+p_{001}\leq U<p_{000}+p_{001}+p_{010}\\ (0,1,1) & \text{Si } p_{000}+p_{001}+p_{010}\leq U<\\ & \quad p_{000}+p_{001}+p_{010}+p_{011}\\ (1,0,0) & \text{Si } p_{000}+p_{001}+p_{010}+p_{011}\leq U<\\ & \quad p_{000}+p_{001}+p_{010}+p_{011}+p_{100}\\ (1,0,1) & \text{Si } p_{000}+p_{001}+p_{010}+p_{011}+p_{100}\leq U<\\ & \quad p_{000}+p_{001}+p_{010}+p_{011}+p_{100}+p_{101}\\ (1,1,0) & \text{Si } p_{000}+p_{001}+p_{010}+p_{011}+p_{100}+p_{101}\leq U<\\ & \quad p_{000}+p_{001}+p_{010}+p_{011}+p_{100}+p_{101}+p_{110}\\ (1,1,1) & \text{Si } U\geq p_{000}+p_{001}+p_{010}+p_{011}+p_{100}+p_{101}+p_{110}\end{cases}$$

Capítulo 4

Implementación del modelo

En este capítulo se presenta el modelo para evaluar el riesgo de la red de IoT de la casa inteligente. Se explica cómo se obtiene la estructura de la red bayesiana, el algoritmo de simulación de ataques, los parámetros de la red bayesiana y finalmente el cálculo del impacto y el riesgo.

4.1. Estructura de la red bayesiana

4.1.1. Grafos de ataque

Un grafo de ataque es un modelo gráfico que representa el conocimiento sobre las vulnerabilidades en una red y sus interacciones, mostrando los diferentes caminos que un atacante puede seguir para alcanzar un objetivo determinado. En la red de IoT de una casa inteligente, un grafo de ataque describe el orden en que deben ocurrir los distintos ataques en los dispositivos hasta llegar a atacar a otro dispositivo que se considera objetivo.

El grafo de ataque referido a la red de IoT de una casa inteligente que se utiliza para el desarrollo del modelo fue presentado por Ibrahim y Nabulsi [6] y se muestra en la figura 4.1. Este grafo está basado en la estructura de la red de IoT de la casa inteligente, es decir, el punto de acceso, nivel de administración, router y nivel de dispositivos; y los tipos de ataques presentados anteriormente. Se considera que el objetivo en el grafo de ataque es el nivel de dispositivos, por lo que los caminos describen las formas en que se puede llegar a atacar el nivel de dispositivos.

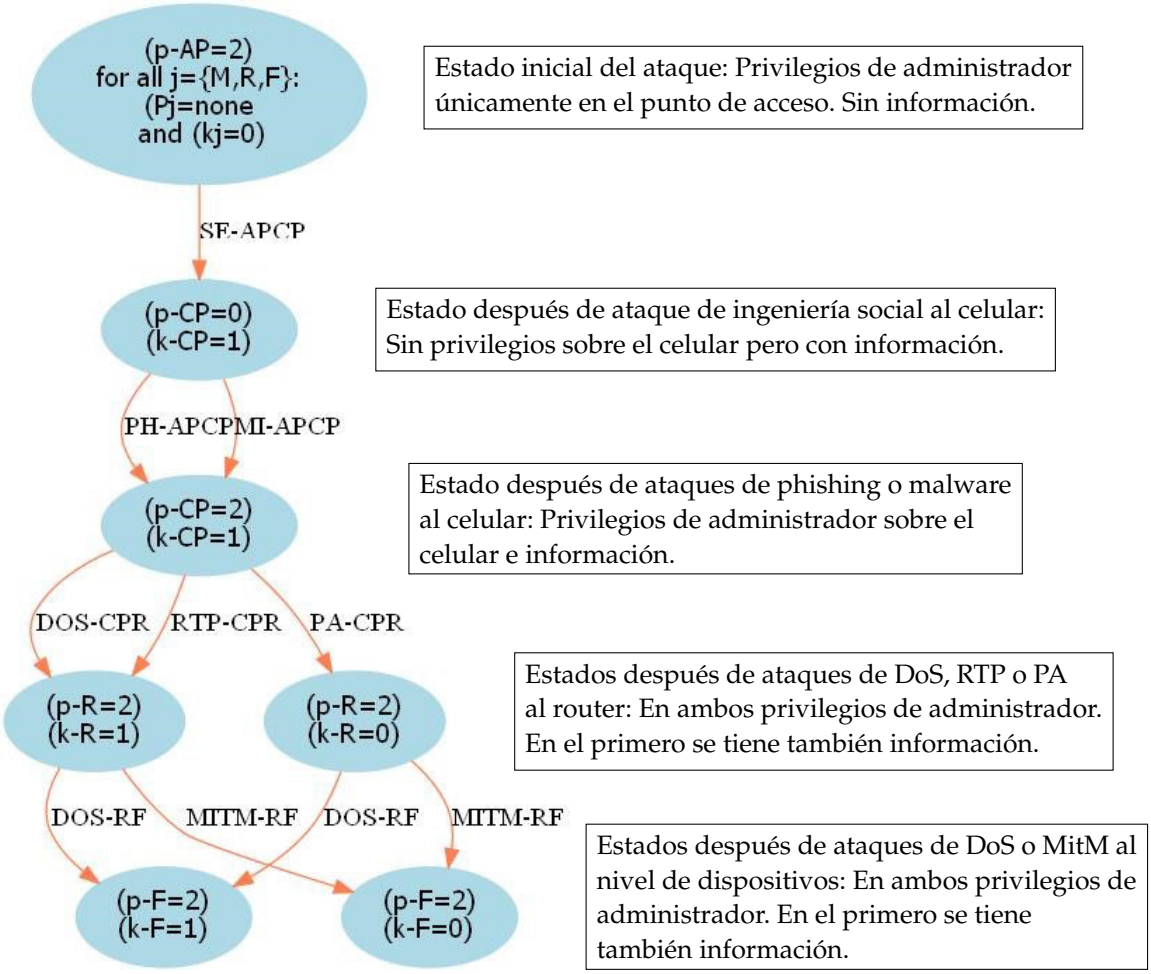


Figura 4.1: Grafo de ataque de la red de IoT de una casa inteligente
Fuente: Elaborado por Ibrahim y Nabulsi [6]

En el grafo de ataque, los nodos representan el estado de cada nivel de la estructura de la red de IoT en cuanto a la información (k), donde 0 significa que el atacante no tiene información y 1 significa que sí; y el tipo de privilegios que el atacante posee (p), donde 0 significa ningún privilegio, 1 privilegios de usuario y 2 privilegios de administrador. Las aristas representan el tipo de ataque y los dispositivos involucrados. De esta forma, en el grafo se pueden distinguir 12 posibles caminos en los que el atacante logra obtener privilegios de administrador en el nivel de dispositivos. Los niveles de la estructura de la red se denotan de la siguiente manera: punto de acceso (AP), nivel de administración (M), sin embargo, se considera que solo un celular (CP) se encuentra en este nivel, router (R) y el nivel de dispositivos (F).

El grafo de ataque se interpreta de la siguiente manera: al principio, el atacante tiene privilegios de administrador en el punto de acceso (p-AP=2), sin embargo, no tiene privilegios ni información sobre ninguno de los otros dispositivos (for all j={M,R,F}: Pj=none and kj=0). Se puede realizar un ataque de ingeniería social

desde el punto de acceso al celular que se encuentra en el nivel de administración (SE-APCP), con el fin de obtener información sobre el celular, pero no privilegios (k-CP=1 y p-CP=0). Una vez obtenida la información, se pueden realizar ataques de phishing y malware desde el punto de acceso al celular (PH-APCP y MI-APCP) con el fin de obtener privilegios de administrador en el celular (p-CP=2), tomando en cuenta que también se tiene la información (k-CP=1). Una vez obtenidos los privilegios de administrador, se pueden realizar ataques de DoS, RTP y persistente desde el celular hasta el router (DOS-CPR, RTP-CPR y PA-CPR). Todos estos ataques permiten obtener privilegios de administrador sobre el router (p-R=2), pero solo los de DoS y RTP permiten obtener información (k-R=1), a diferencia del ataque persistente donde esto no ocurre (k-R=0). Finalmente, debido a que los tres ataques permiten tener privilegios, entonces se pueden realizar ataques de DoS y MitM desde el router hasta el nivel de dispositivos (DOS-RF y MITM-RF) y de esta forma se obtiene acceso de administrador al nivel de dispositivos (p-F=2); sin embargo, solo los ataques de DoS permiten obtener información del nivel de dispositivos (k-F=1), mientras que los de MitM no lo permiten (k-F=0).

Este modelo de grafo de ataque se limita a analizar las formas en las que pueden ocurrir los distintos ataques en la red de IoT de la casa inteligente. Por otro lado, el modelo que se desarrolla en este trabajo parte del grafo de ataque para diseñar el grafo dirigido acíclico que conforma la estructura de una red bayesiana. La red bayesiana luego se utiliza para determinar las probabilidades de ocurrencia de los ataques, el impacto y el riesgo de los mismos, complementando el estudio de la ciberseguridad en las casas inteligentes.

4.1.2. Grafo dirigido acíclico

La construcción de una red bayesiana se lleva a cabo en tres pasos principales. Primero, se selecciona el conjunto de variables relevantes y sus posibles valores. Luego se construye la estructura de la red conectando las variables en un grafo dirigido acíclico. Finalmente, se define la distribución de probabilidad condicional para cada variable de la red [4]. Basado en el grafo de ataque presentado por Ibrahim y Nabulsi se determina el conjunto de variables y se construye el grafo dirigido acíclico de la red bayesiana. Para el modelo consideramos variables aleatorias discretas que siguen una distribución Bernoulli ya que se utilizan para determinar el estado de los ataques en la red, es decir, las variables toman el valor de 1 si ocurre el tipo de ataque especificado y 0 caso contrario. Así, para el presente trabajo se definen las

siguientes variables:

- **SECP:** Ocurre un ataque de ingeniería social desde el punto de acceso al nivel de administración.
- **PHCP:** Ocurre un ataque de phishing desde el punto de acceso al nivel de administración.
- **MICP:** Ocurre un ataque de malware desde el punto de acceso al nivel de administración.
- **DOSR:** Ocurre un ataque de DoS desde el nivel de administración al router.
- **RTPR:** Ocurre un ataque de RTP desde el nivel de administración al router.
- **PAR:** Ocurre un ataque persistente desde el nivel de administración al router.
- **DOSF:** Ocurre un ataque de DoS desde el router al nivel de dispositivos.
- **MITMF:** Ocurre un ataque de Man In The Middle desde el router al nivel de dispositivos.

Las variables se conectan en un grafo dirigido acíclico de tal forma que se mantengan los caminos del grafo de ataque de la figura 4.1. De esta forma, en la figura 4.2 se presenta el grafo dirigido acíclico de la red bayesiana.

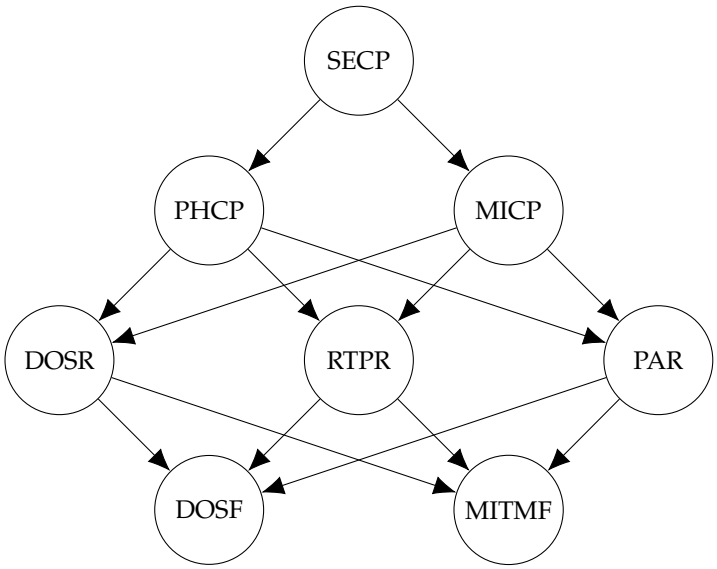


Figura 4.2: Grafo dirigido acíclico de la red bayesiana

4.2. Simulación de ataques

El objetivo de la simulación es describir cómo se realizan los ciberataques a través de los nodos de la red. Para cada nivel de la estructura de la red de IoT de la casa inteligente, se seleccionan los tipos de ataques que se llevan a cabo y se determina si el atacante logró vulnerar los dispositivos. Como resultado de la simulación se obtiene un conjunto de datos donde cada columna representa un nodo de la red y cada fila representa un evento de ataque completo en la red, indicando los nodos por los cuales se realizaron los ataques. Las consideraciones generales para la simulación son que el atacante puede realizar múltiples ataques desde un nodo y no se permiten los ataques que no sigan las rutas descritas por el grafo.

4.2.1. Selección de ataques

Los ciberataques en la casa inteligente siguen los caminos descritos en el grafo de la red bayesiana de la figura 4.2. Por ejemplo, uno de los posibles caminos que logra vulnerar el nivel de dispositivos es $SECP \rightarrow PHCP \rightarrow RTPR \rightarrow DOSF$.

Para cada nivel de la estructura de la red de IoT de la casa inteligente, el primer paso es decidir el tipo de ataque o los tipos de ataques que se ejecutan y esto se realiza simulando vectores aleatorios de 2 o 3 variables con distribución Bernoulli. Estas variables toman el valor de 1 si se selecciona el nodo para el ataque y 0 caso contrario. Se considera que el atacante siempre escoge por lo menos uno de los ataques disponibles en cada nivel, por lo tanto, el primer nodo del grafo $(SECP)$ siempre será escogido y no hace falta simular la selección de esta variable. Los vectores que se simulan son:

- **Selección de ataques en el nivel de administración:** (S_{PHCP}, S_{MICP}) con S_{PHCP}: se selecciona un ataque de phishing y S_{MICP}: se selecciona un ataque de malware.
- **Selección de ataques en el nivel de router:** $(S_{DOSR}, S_{RTPR}, S_{PAR})$ con S_{DOSR}: se selecciona un ataque de DoS, S_{RTPR}: se selecciona un ataque RTP y S_{PAR}: se selecciona un ataque persistente.
- **Selección de ataques en el nivel de dispositivos:** (S_{DOSF}, S_{MITMF}) con S_{DOSF}: se selecciona un ataque de DoS y S_{MITMF}: se selecciona un ataque de MitM.

4.2.2. Escenarios de simulación

En la estructura de la red de IoT de la casa inteligente, el nivel de dispositivos conforma los dispositivos que permiten la automatización de la casa y son aquellos que tienen menores niveles de seguridad. En este trabajo, la evaluación del riesgo está enfocada en los ataques de DoS y MitM a este nivel de dispositivos. El objetivo de plantear los escenarios de simulación es comparar distintos comportamientos de los atacantes en relación a los ataques que son seleccionados en cada nivel de la red de IoT. Esta selección depende de la prioridad de protección que el propietario de la casa inteligente o el evaluador de riesgo proporcione para los distintos ataques a los dispositivos en cada nivel de la red de IoT. Así, en el primer escenario se considera que los ataques tienen la misma prioridad; en el segundo y cuarto escenario, se da una mayor prioridad a los ataques de DoS cuyo objetivo es el bloqueo de los dispositivos; y en el tercer y quinto escenario, se da una mayor prioridad a los ataques de MitM cuyo objetivo es interceptar los canales de comunicación entre los dispositivos. Los valores para los distintos escenarios de simulación fueron determinados a manera de ejemplo para la aplicación de la metodología. Sin embargo, dependiendo de las necesidades o del requerimiento del propietario de la red se puede implementar cualquier escenario, dándole las prioridades sugeridas por dicho dueño. Los parámetros de los vectores aleatorios propuestos para la selección de ataques son las probabilidades de la función de probabilidad conjunta, y se denominan probabilidades de selección; estas definen los escenarios de simulación y para cada nivel se muestran en las tablas 4.1, 4.2 y 4.3.

Variables		**Escenarios**				
S_{PHCP}	S_{MICP}	1	2	3	4	5
0	0	0,00	0,00	0,00	0,00	0,00
0	1	0,45	0,60	0,30	0,75	0,15
1	0	0,45	0,30	0,60	0,15	0,75
1	1	0,10	0,10	0,10	0,10	0,10

Tabla 4.1: Probabilidades de selección para el nivel de administración

Variables			Escenarios				
S_{DOSR}	S_{RTPR}	S_{PAR}	1	2	3	4	5
0	0	0	0,00	0,00	0,00	0,00	0,00
0	0	1	0,25	0,20	0,25	0,15	0,25
0	1	0	0,25	0,25	0,30	0,25	0,35
0	1	1	0,07	0,06	0,08	0,05	0,09
1	0	0	0,25	0,30	0,20	0,35	0,15
1	0	1	0,07	0,07	0,06	0,07	0,05
1	1	0	0,07	0,08	0,07	0,09	0,07
1	1	1	0,04	0,04	0,04	0,04	0,04

Tabla 4.2: Probabilidades de selección para el nivel de router

Variables		Escenarios				
S_{DOSF}	S_{MITMF}	1	2	3	4	5
0	0	0,00	0,00	0,00	0,00	0,00
0	1	0,45	0,30	0,60	0,15	0,75
1	0	0,45	0,60	0,30	0,75	0,15
1	1	0,10	0,10	0,10	0,10	0,10

Tabla 4.3: Probabilidades de selección para el nivel de dispositivos

Donde los valores de (S_{PHCP}, S_{MICP}), $(S_{DOSR}, S_{RTPR}, S_{PAR})$ y (S_{DOSF}, S_{MITMF}) determinan las posibles combinaciones de selecciones de ataques en cada nivel y los escenarios se definen mediante las probabilidades de selección de ataques.

Cuando los vectores tienen los valores $(0,0)$, $(0,0,0)$ y $(0,0)$, esto indica que no se seleccionaron ataques y en ese caso se asigna la probabilidad de 0 debido a que se considera que el atacante siempre escoge por lo menos uno de los ataques disponibles en cada nivel de la estructura de la red de IoT de la casa inteligente.

La descripción de las probabilidades cuando se escogen todos los posibles ataques en cada nivel es la siguiente:

- En el nivel de administración que se presenta en la tabla 4.1, cuando el vector es $(1,1)$ se considera una probabilidad de 0,10 para todos los escenarios porque generalmente los atacantes eligen solo un tipo de ataque, aunque no se descarta la posibilidad de un ataque doble y por eso se considera una probabilidad baja de 0,10.

- En el nivel de router que se presenta en la tabla 4.2, cuando el vector es $(1,1,1)$ se considera una probabilidad de 0,04 para todos los escenarios, debido a que no es común que un atacante elija tres tipos distintos de ataques simultáneamente para un solo dispositivo, sin embargo, esta posibilidad no es descartada.

- En el nivel de dispositivos que se presenta en la tabla 4.3, cuando el vector es $(1,1)$ se considera el mismo caso que el nivel de administración, es decir, una probabilidad del 0,10.

Para los vectores intermedios entre $(0,0)$ y $(1,1)$ en los niveles de administración y de dispositivos y para los vectores entre $(0,0,0)$ y $(1,1,1)$ en el nivel de router, las probabilidades varían en cada escenario ya que se consideran diferentes tendencias en la selección de ataques simples y dobles. La descripción de los escenarios en cuanto a estos vectores es la siguiente:

- **Escenario 1:** En el nivel de administración y de dispositivos se considera igual probabilidad de 0,45 para los ataques simples, es decir, cuando los vectores son $(0,1)$ y $(1,0)$. En el nivel de router se considera una probabilidad de 0,25 para los ataques simples, es decir, $(0,0,1)$, $(0,1,0)$ y $(1,0,0)$; y una probabilidad de 0,07 para los ataques dobles, es decir, $(0,1,1)$, $(1,0,1)$ y $(1,1,0)$.

- **Escenarios 2 y 4:** Para ambos escenarios, en el nivel de administración se analiza el efecto cuando se asigna mayor probabilidad a S_{MICP} que a S_{PHCP}. En el nivel de router, en cuanto a los ataques simples, se considera mayor probabilidad a S_{DOSR} que a S_{RTPR}, y a este mayor que a S_{PAR}. En cuanto a los ataques dobles el orden de mayor a menor probabilidad se considera de la siguiente manera: S_{DOSR} y S_{RTPR}, luego S_{DOSR} y S_{PAR} y finalmente S_{RTPR} y S_{PAR}. Para el nivel de dispositivos se considera una mayor probabilidad a S_{DOSF} que S_{MITMF}. La diferencia entre el escenario 4 y el escenario 2 es que en el segundo se asigna una incidencia media y en el cuarto una incidencia alta a los ataques con mayor probabilidad en cada nivel.

- **Escenarios 3 y 5:** Para ambos escenarios, en el nivel de administración se analiza el efecto cuando se asigna mayor probabilidad a S_{PHCP} que a S_{MICP}. En cuanto al nivel de router, para los ataques simples, el orden de mayor a menor probabilidad es: S_{RTPR}, S_{PAR} y S_{DOSR}; y en cuanto a los ataques dobles: S_{RTPR} y S_{PAR}, luego S_{DOSR} y S_{RTPR} y finalmente S_{DOSR} y S_{PAR}. Para el nivel de dispositivos se considera una mayor probabilidad a S_{MITMF} que S_{DOSF}. La diferencia entre el escenario 3 y el escenario 5 es que en el tercero se asigna una incidencia media y en el quinto una incidencia alta a los ataques con mayor probabilidad en cada nivel.

4.2.3. Vulnerabilidades y puntuaciones CVSS

El Common Vulnerability Scoring System (CVSS) [14] es un sistema de puntuación que mide el impacto o la gravedad de las vulnerabilidades en la seguridad de la Tecnología de la Información. Las puntuaciones CVSS se basan en las principales características técnicas de las vulnerabilidades de software, hardware y firmware. El CVSS fue desarrollado para generar puntajes cualitativos y cuantitativos de vulnerabilidad de dispositivos. En general, para calcular estos puntajes se utilizan tres métricas: base, temporal y ambiental. La puntuación base refleja la gravedad de una vulnerabilidad de acuerdo con sus características intrínsecas que son constantes en el tiempo y se estima en base al peor de los casos en diferentes escenarios. Las métricas temporales se describen en función de factores que cambian con el tiempo y las métricas ambientales se describen de acuerdo con un entorno específico.

Las puntuaciones CVSS se han utilizado en trabajos anteriores para estimar los parámetros de las redes bayesianas discretas en modelos de evaluación de riesgos en sistemas informáticos [10, 5, 12, 19]. En [12, 5] se considera que la probabilidad de explotar una vulnerabilidad es la puntuación CVSS dividida por el tamaño del dominio, que es 10. Sin embargo, este enfoque no considera las relaciones causales entre los ataques presentes en el grafo. En este trabajo, las puntuaciones CVSS no se utilizan como parámetros de la red bayesiana, sino como uno de los criterios para la simulación de ataques. Las métricas base de cada nodo de la red se transforman en probabilidad siguiendo el enfoque de [5], es decir, dividiendo por 10 la puntuación CVSS, y este valor se utiliza como parámetro de una variable aleatoria con distribución Bernoulli. Estas variables se denominan variables de vulnerabilidad y su simulación indica si el atacante tuvo éxito al llevar a cabo el ataque. De esta forma, las variables de vulnerabilidad en la red son las siguientes: para el nivel de administración V_{SECP}, V_{PHCP} y V_{MICP}, para el nivel de router V_{DOSR}, V_{RTPR} y V_{PAR}, y para el nivel de dispositivos V_{DOSF} y V_{MITMF}. Las métricas base de CVSS que se utilizan para el cálculo de los parámetros de estas variables son:

- **Vector de ataque (AV):** Indica el contexto en el que es posible la explotación de vulnerabilidades. La puntuación base es mayor cuanto más remoto es el ataque. Los valores que puede tomar son: red (N), adyacente (A), local (L) y físico (P).
- **Complejidad de ataque (AC):** Describe las condiciones más allá del control del atacante que deben existir para llevar a cabo el ataque. Tales condiciones

pueden requerir la recopilación de más información sobre el objetivo o excepciones computacionales. La puntuación base es mayor para los ataques menos complejos. Los valores que puede tomar son: bajo (L) y alto (H).

- **Interacción de usuario (UI):** Determina si la vulnerabilidad puede explotarse únicamente a voluntad del atacante, o si un usuario debe participar de alguna manera. La puntuación base es mayor cuando no se requiere la interacción del usuario. Los valores que puede tomar son: ninguno (N) y requerido (R).

- **Privilegios requeridos (PR):** Describe el nivel de privilegios que debe tener un atacante antes de explotar con éxito la vulnerabilidad. La puntuación base es mayor si no se requieren privilegios. Los valores que puede tomar son: ninguno (N), bajo (L) y alto (H).

- **Alcance (S):** Determina si una vulnerabilidad afecta los recursos en componentes más allá de su alcance de seguridad. La puntuación base es mayor si el alcance es modificado. Los valores que puede tomar son: modificado (C) y no modificado (U).

- **Impacto en la confidencialidad (C):** Mide el impacto en la confidencialidad de la información debido a una vulnerabilidad explotada con éxito. La confidencialidad se refiere a limitar el acceso y la divulgación de la información solo a los usuarios autorizados, así como a prevenir el acceso o la divulgación a personas no autorizadas. La puntuación base es mayor cuando la pérdida del componente afectado es mayor. Los valores que puede tomar son: ninguno (N), bajo (L) y alto (H).

- **Impacto en la integridad (I):** Mide el impacto en la integridad de una vulnerabilidad explotada con éxito. La integridad se refiere a la confiabilidad y veracidad de la información. Se refiere principalmente a la modificación de datos y el tipo de control que posee el atacante en la modificación de la información. La puntuación base es mayor cuando la consecuencia para el componente afectado es mayor. Los valores que puede tomar son: ninguno (N), bajo (L) y alto (H).

- **Impacto en la disponibilidad (A):** Mide el impacto en la disponibilidad del componente afectado como resultado de una vulnerabilidad explotada con éxito. Esta métrica se refiere a la pérdida de disponibilidad del componente afectado en sí, como un servicio en red (web, base de datos, correo electrónico). La puntuación base es mayor cuando la consecuencia para el componente

afectado es mayor. Los valores que puede tomar son: ninguno (N), bajo (L) y alto (H).

CVSS pone a disposición del usuario una calculadora oficial [13] que entrega las puntuaciones CVSS en base a los valores de las métricas descritas para cada ataque en un dispositivo. Estas puntuaciones son transformadas a probabilidades y se utilizan como parámetros de cada variable V_{Nodo}, para todos los nodos del grafo de la red bayesiana. Para el caso de este trabajo, los valores de las métricas, la puntuación CVSS y los parámetros de las variables V_{Nodo} denotados por Par. V_{Nodo} se encuentran tabulados en la tabla 4.4.

Nodo	**AV**	**AC**	**UI**	**PR**	**S**	**C**	**I**	**A**	**CVSS**	**Par. V_{Nodo}**
SECP	N	L	N	N	C	H	N	N	8,6	0,86
PHCP	N	L	R	N	C	H	L	N	8,2	0,82
MICP	N	L	R	N	C	H	L	L	8,8	0,88
DOSR	N	L	N	H	C	H	L	H	9,0	0,90
RTPR	N	L	N	H	C	H	L	H	9,0	0,90
PAR	N	L	N	H	C	H	L	H	9,0	0,90
DOSF	N	H	N	N	C	H	H	H	9,0	0,90
MITMF	A	H	R	L	C	H	H	H	7,6	0,76

Tabla 4.4: Métricas base CVSS y parámetros de las variables V_{Nodo}

4.2.4. Algoritmo de simulación de ataques

Considerando las variables de selección y vulnerabilidad definidas en las secciones 4.2.1 y 4.2.3, se desarrolla el algoritmo 1 para simular ataques a la red.

Algoritmo 1: Simulación de Ataques

Entrada: N, S_{PHCP}, S_{MICP}, S_{DOSR}, S_{RTPR}, S_{PAR}, S_{DOSF}, S_{MITMF}, V_{SECP}, V_{PHCP}, V_{MICP}, V_{DOSR}, V_{RTPR}, V_{PAR}, V_{DOSF}, V_{MITMF}
Salida: Conjunto de datos de ataque X

Para $n = 1$ **a** $n = N$ **hacer**
- Simular V_{SECP}
- **Si** $V_{SECP} = 1$, **entonces**
  - Simular (S_{PHCP}, S_{MICP})
  - Para los $i \in \{PHCP, MICP\}$ tales que $S_i = 1$ en el paso anterior, simular V_i
  - $X_{n,1} = \{i \in \{PHCP, MICP\} \mid V_i = 1$ en el paso anterior$\}$
  - **Si** $|X_{n,1}| \neq 0$, **entonces**
    - **Para** $j \in X_{n,1}$, **hacer**
      - Simular $(S_{DOSR}, S_{RTPR}, S_{PAR})$
      - Para los $i \in \{DOSR, RTPR, PAR\}$ tales que $S_i = 1$ en el paso anterior, simular V_i
      - $X_{n,2,j} = \{i \in \{DOSR, RTPR, PAR\} \mid V_i =$ 1 en el paso anterior$\}$
    - $X_{n,2} = \bigcup_{j \in X_{n,1}} X_{n,2,j}$
    - Eliminar los elementos repetidos de $X_{n,2}$
    - **Si** $|X_{n,2}| \neq 0$, **entonces**
      - **Para** $k \in X_{n,2}$ **hacer**
        - Simular (S_{DOSF}, S_{MITMF})
        - Para los $i \in \{DOSF, MITMF\}$ tales que $S_i = 1$ en el paso anterior, simular V_i
        - $X_{n,3,k} = \{i \in \{DOSF, MITMF\} \mid V_i =$ 1 en el paso anterior$\}$
      - $X_{n,3} = \bigcup_{k \in X_{n,2}} X_{n,3,k}$
      - Eliminar los elementos repetidos de $X_{n,3}$
      - $X_n = \{SECP\} \cup X_{n,1} \cup X_{n,2} \cup X_{n,3}$
    - **En otro caso**
      - $X_n = \{SECP\} \cup X_{n,1}$
  - **En otro caso**
    - $X_n = \{SECP\}$
- **En otro caso**
  - $X_n = \emptyset$
- $X = \bigcup_{n=1}^{N} X_n$

Los datos de entrada son el número de eventos de ataque a la red (N), las variables de selección y las variables de vulnerabilidad. Para que un nodo se considere atacado debe ser seleccionado y vulnerado. En este algoritmo, para cada evento de ataque a la red se guardan en cada X_n con $n = 1, \ldots, N$, los nombres de los nodos

que fueron tanto seleccionados como vulnerados. Finalmente se guardan en X todos los datos de ataques.

4.2.5. Datos de ataques

Para la implementación del modelo de evaluación de riesgo en la red de IoT de la casa inteligente se simulan 10000 datos de eventos de ataque para cada escenario de simulación, utilizando el algoritmo 1.

Para la implementación de este algoritmo en lenguaje R, se utilizan los parámetros de las variables de selección y vulnerabilidad, y como resultado se obtiene una tabla donde cada columna representa un nodo de la red y cada fila un evento de ataque, colocando los valores 1 y 0 en cada celda si el nodo representado por la columna es atacado o no atacado, respectivamente. En la tabla 4.5 se presenta un ejemplo de los resultados de la simulación de cinco eventos de ataque a la red.

n	SECP	PHCP	MICP	DOSR	RTPR	PAR	DOSF	MITMF
1	1	0	1	1	0	0	1	0
2	1	1	0	0	1	1	0	1
3	0	0	0	0	0	0	0	0
4	1	1	1	1	0	0	0	0
5	1	0	1	1	0	0	1	1

Tabla 4.5: Ejemplo de simulación de datos de ataques

En este ejemplo se puede observar que solo el primer, segundo y quinto evento llegan al nivel de dispositivos. En el tercer evento no se logró concretar el ataque de ingeniería social al celular y por lo tanto no ocurrieron ninguno de los otros ataques. Por otro lado, en el cuarto evento se realizó un ataque doble de phishing y malware en el nivel de administración y los ataques llegaron hasta el nivel de router, sin embargo, no lograron vulnerar el nivel de dispositivos.

4.3. Parametrización de la red bayesiana

Una vez diseñada la estructura de la red bayesiana y definidos los conjuntos de datos de ataques mediante simulación, se procede a estimar los parámetros de la red bayesiana utilizando el método de máxima verosimilitud. Estos parámetros, denotados por θ, son todas las posibles funciones de probabilidad condicional de

las variables dados sus padres en el grafo. Para la red bayesiana de este trabajo, los nodos del grafo y sus padres se detallan en la tabla 4.6.

Nodo	Nodos Padres
SECP	-
PHCP	SECP
MICP	SECP
DOSR	PHCP, MICP
RTPR	PHCP, MICP
PAR	PHCP, MICP
DOSF	DOSR, RTPR, PAR
MITMF	DOSR, RTPR, PAR

Tabla 4.6: Nodos y sus padres en el grafo de la red bayesiana del modelo

Por tanto, los parámetros son: θ_{SECP}, $\theta_{PHCP|SECP}$, $\theta_{MICP|SECP}$, $\theta_{DOSR|PHCP,MICP}$, $\theta_{RTPR|PHCP,MICP}$, $\theta_{PAR|PHCP,MICP}$, $\theta_{DOSF|DOSR,RTPR,PAR}$ y $\theta_{MITMF|DOSR,RTPR,PAR}$. Dado que la variable SECP no tiene padres en el grafo entonces el parámetro θ_{SECP} representa una función de probabilidad marginal. En este trabajo, debido a que las variables de la red bayesiana son discretas, entonces cada función de probabilidad de los parámetros es un conjunto de probabilidades. En la tabla 4.7 se muestran los parámetros de la red bayesiana propuesta y el conjunto de probabilidades que se estiman por cada parámetro.

Parámetros	Probabilidades
θ_{SECP}	$P(SECP)$
$\theta_{PHCP\|SECP}$	$P(PHCP \mid SECP)$
$\theta_{MICP\|SECP}$	$P(MICP \mid SECP)$
$\theta_{DOSR\|PHCP,MICP}$	$P(DOSR \mid PHCP, MICP)$
$\theta_{RTPR\|PHCP,MICP}$	$P(RTPR \mid PHCP, MICP)$
$\theta_{PAR\|PHCP,MICP}$	$P(PAR \mid PHCP, MICP)$
$\theta_{DOSF\|DOSR,RTPR,PAR}$	$P(DOSF \mid DOSR, RTPR, PAR)$
$\theta_{MITMF\|DOSR,RTPR,PAR}$	$P(MITMF \mid DOSR, RTPR, PAR)$

Tabla 4.7: Parámetros de la red bayesiana

En esta tabla, la notación en cuanto al conjunto de probabilidades implica a todos los posibles valores de las variables descritas. Así, para la probabilidad $P(SECP)$ los valores a estimar son $P(SECP = 1)$ y $P(SECP = 0)$. Para las probabilidades condicionales, por ejemplo, $P(PHCP \mid SECP)$ implica las dos funciones de probabilidad condicional $P(PHCP \mid SECP = 1)$ y $P(PHCP \mid SECP = 0)$; por lo tanto, se deben calcular cuatro probabilidades $P(PHCP = 1 \mid SECP = 1)$,

$P(PHCP = 0 \mid SECP = 1)$, $P(PHCP = 1 \mid SECP = 0)$ y $P(PHCP = 0 \mid SECP = 0)$. Para los demás parámetros el razonamiento es similar.

4.4. Inferencias para la evaluación del riesgo

Uno de los objetivos principales al implementar un modelo de redes bayesianas es realizar inferencias para actualizar el conocimiento que se posee sobre una variable utilizando evidencias en otras variables. Considerando que el objetivo del grafo de ataque es evitar los ataques al nivel de dispositivos [6], la evaluación del riesgo se basa en este principio. Así, se calculan las siguientes inferencias en la red bayesiana.

- **Probabilidades de variables:** Se refiere al cálculo de las probabilidades de que ocurra cierto tipo de ataque a los dispositivos de la red, sin evidencias. En este caso se calculan: $P(DOSF = 1)$, $P(MITMF = 1)$ y $P(DOSF = 1, MITMF = 1)$.

- **Densidades marginales a posteriori:** Se considera como evidencia que $Y = 1$, con $Y \in \{SECP, PHCP, MICP, DOSR, RTPR, PAR\}$ y se calcula $P(DOSF = 1 \mid Y = 1)$, $P(MITMF = 1 \mid Y = 1)$ y $P(DOSF = 1, MITMF = 1 \mid Y = 1)$.
  También se calcula $P(DOSF = 1 \mid PHCP, MICP)$, $P(MITMF = 1 \mid PHCP, MICP)$, $P(DOSF = 1, MITMF = 1 \mid PHCP, MICP)$ para todos los valores de $PHCP$ y $MICP$.

- **Explicaciones más probables:** Para el ataque DoS, se determinan los valores de $SECP$, $PHCP$, $MICP$, $DOSR$, $RTPR$ y PAR, tales que: $P(SECP, PHCP, MICP, DOSR, RTPR, PAR \mid DOSF = 1, MITMF = 0)$ es máximo. Para el ataque MitM, se determinan los valores de $SECP$, $PHCP$, $MICP$, $DOSR$, $RTPR$ y PAR, tales que: $P(SECP, PHCP, MICP, DOSR, RTPR, PAR \mid DOSF = 0, MITMF = 1)$ es máximo.

Para el cálculo de las inferencias se utiliza un algoritmo exacto debido a la precisión de los resultados y a que es factible porque el grafo de la red bayesiana tiene pocos nodos. Específicamente, se utiliza el algoritmo de árbol de uniones, debido a que presenta mejoras en relación al algoritmo de eliminación de variables [12].

4.5. Impacto de los ataques

En este trabajo, el impacto es definido como la pérdida de confidencialidad, integridad y disponibilidad en los dispositivos si son vulnerados con éxito mediante cualquier tipo de ataque. Así, el método propuesto para el cálculo del impacto está basado en las puntuaciones CVSS e incluye algunas métricas base, temporales y ambientales [1]. Las métricas base utilizadas son el impacto a la confidencialidad, integridad y disponibilidad que se explicaron anteriormente para las vulnerabilidades. Las métricas ambientales son el requerimiento de confidencialidad, integridad y disponibilidad; y permiten personalizar la puntuación CVSS según la importancia del activo afectado, medido en términos de confidencialidad, integridad y disponibilidad. Estas métricas pueden tomar los valores: alto, cuando es probable que la pérdida de confidencialidad, integridad o disponibilidad tenga un efecto adverso catastrófico en la organización o en las personas asociadas con la organización (por ejemplo, empleados, clientes); medio, cuando el efecto adverso es grave; bajo, cuando el efecto adverso es limitado y no definido cuando no hay información suficiente para elegir uno de los otros valores. La métrica temporal utilizada es el nivel de remediación y este se refiere al tipo de soluciones que se proponen para las vulnerabilidades. Normalmente las vulnerabilidades de los dispositivos no son corregidas en cuanto son descubiertas y se proporcionan soluciones provisionales hasta que se emita una oficial. Cada una las etapas hasta la solución oficial reduce la puntuación temporal CVSS y en este caso, el valor del impacto. Cuanto menos oficial y permanente sea una solución, mayor será la puntuación de impacto [14]. En la tabla 4.8, se describen los valores de las distintas métricas que se utilizan para el cálculo del impacto. Los valores numéricos de las métricas y sus escalas están dados por CVSS.

Métrica	CVSS	Valor	Valor
Impacto a la confidencialidad (C)	Base	Alto (H)	1
Impacto a la integridad (I)		Bajo (L)	0,5
Impacto a la disponibilidad (A)		Ninguno (N)	0
Req. de confidencialidad (CR)	Ambiental	Bajo (L)	30
Req. de integridad (IR)		Medio (M)	60
Req. de disponibilidad (AR)		Alto (H)	100
		No definido (N)	100
Nivel de remediación (RL)	Temporal	Arreglo oficial (OF)	0,15
		Arreglo temporal (TF)	0,1
		Solución alterna (W)	0,05
		No disponible (U)	0
		No definido (N)	0

Tabla 4.8: Métricas para el cálculo del impacto

El valor del impacto se calcula mediante la fórmula [1]:

$$Impacto = \frac{CR \times C + IR \times I + AR \times A}{3} \times (1 - RL)$$

Las métricas CVSS para el cálculo del impacto tienen tres áreas: base, ambiental y temporal. A cada tipo de métrica, CVSS le asigna un valor numérico con diferentes escalas. Para el caso de las métricas base sus valores están entre 0 y 1 porque representan la fracción que se va a tomar de las métricas ambientales, cuyos valores están entre 0 y 100. Los valores de la métrica temporal se hallan entre 0 y 1 porque representan la fracción del impacto que se va a restar del puntaje al considerar el tipo de solución disponible para el ataque.

Para su aplicación en la fórmula, primero se calculan los denominados puntajes de impacto no remediados multiplicando los valores de impacto a la confidencialidad, integridad y disponibilidad en las métricas base por su respectivo requerimiento en las métricas ambientales, cuyo valor numérico está entre 0 y 100. Las métricas base de impacto a la confidencialidad, integridad y disponibilidad representan la fracción de su respectivo requerimiento en las métricas ambientales y por esta razón toman valores entre 0 y 1, como se indicó anteriormente. Finalmente, el valor del impacto se calcula con el promedio de estos puntajes de impacto no remediados multiplicado por 1 menos el nivel de remediación. Dado que el valor del nivel de remediación se encuentra entre 0 y 1, este reduce el valor del impacto no remediado considerando la existencia de soluciones para el tipo de ataque que se analiza.

De esta forma el impacto se representa como un puntaje de 0 a 100, donde 0 significa que no tiene impacto y 100 es el impacto máximo. Por lo tanto, en la tabla

4.9 se presentan los valores de las métricas que se consideran para cada nodo, junto con el valor del impacto. Aquí se puede ver que los ataques al nivel de dispositivos tienen los puntajes más altos, mientras que los ataques al nivel de administración tienen los puntajes más bajos.

Nodo	C	I	A	CR	IR	AR	RL	Impacto
SECP	H	N	N	M	L	L	U	20,00
PHCP	H	L	N	M	L	L	OF	21,25
MICP	H	L	L	M	M	M	OF	34,00
DOSR	H	L	H	H	M	H	W	72,83
RTPR	H	L	H	H	M	M	W	60,17
PAR	H	L	H	H	L	M	W	55,42
DOSF	H	H	H	H	M	H	U	86,67
MITMF	H	H	H	M	M	H	U	73,33

Tabla 4.9: Métricas para el cálculo del impacto en el modelo

En las bases de datos obtenidas por simulación, para determinar el impacto de un ataque completo a la red, que es una fila de la base, se suman los valores del impacto de los nodos en los que ocurrieron ataques. Para calcular el impacto de un evento que incluya un ataque en específico, se filtran los registros donde ocurrió el ataque al nodo que se tiene de objetivo y se obtiene la media de la suma de impactos de estos registros. Para el caso de probabilidades condicionales, se filtran los registros donde ocurrieron tanto lo que se quiere evaluar como la evidencia y de igual forma se obtiene la media.

4.6. Evaluación del riesgo

En este trabajo, el riesgo se lo considera como una función de la probabilidad de un ataque en un dispositivo y el impacto que este ataque genera. El riesgo de los eventos se calcula con la siguiente fórmula [1].

$$Riesgo = Probabilidad \cdot Impacto \tag{4.1}$$

Las probabilidades de los eventos se obtienen de las inferencias en la red bayesiana que fueron explicadas en la sección 4.4 y el impacto del método explicado en la sección 4.5.

Individualmente, los ataques en cada dispositivo, es decir cada nodo, pueden tener un valor de impacto entre 0 y 100. Por otro lado, para calcular el impacto de un evento de ataque a la red se suman todos los valores de impacto de los nodos

por los que ocurrió el ataque. De esta forma, dado que el grafo tiene ocho nodos, el impacto de un evento puede tomar valores entre 0 y 800. Ahora, considerando que el impacto en cada nodo es fijo, entonces el mayor valor de impacto que puede tener un evento es 423,67; que es la suma de los valores de la columna Impacto, de la tabla 4.9. Tomando en cuenta que las probabilidades pueden tomar valores entre 0 y 1 y que el riesgo se calcula con la fórmula 4.1, entonces, el riesgo solo puede tomar valores entre 0 y 423,67.

El valor de riesgo que se calcula para diferentes eventos de ataque es utilizado con fines comparativos para determinar el tipo ataque que representa mayor o menor riesgo en el nivel de dispositivos; y su comparación con el ataque doble de DoS y MitM.

Los resultados que se obtienen del cálculo de la probabilidad, impacto y riesgo en los cinco escenarios de simulación descritos en la sección 4.2.2 permiten que la persona encargada de implementar los sistemas de seguridad en la casa inteligente pueda determinar las mejores condiciones de seguridad individual de los dispositivos, con el propósito de disminuir los efectos de un posible ataque hacia la red de IoT y de esta forma disminuir el impacto económico que podría ocurrir a los intereses del propietario de la casa inteligente. Toda esta información se tabula en un informe que recoge tanto los resultados de la simulación como la descripción de los riesgos que se pueden presentar en la red de IoT de la casa inteligente. Un ejemplo de modelo de este informe se presenta en el Apéndice A.

Capítulo 5

Resultados

En este capítulo se presentan los resultados de la evaluación del riesgo en la red de IoT de la casa inteligente. Se simularon 10000 datos de ataques para cada escenario de simulación propuesto. Para los resultados solo se contempla el caso de los ataques al nivel de dispositivos debido a que la construcción de la red bayesiana se realizó con ese objetivo.

5.1. Resultados sin considerar evidencia de ataques

Las figuras 5.1, 5.2 y 5.3 muestran la probabilidad, impacto y riesgo de ataques de DoS, MitM y ataque doble de DoS y MitM al nivel de dispositivos, sin considerar evidencia de ataques a los demás nodos. Se diferencian los 5 escenarios de simulación propuestos.

En la figura 5.1 se puede observar que en el nivel de dispositivos, para el primer, segundo y cuarto escenario, el ataque de DoS tiene mayor probabilidad que el de MitM, mientras que para el tercer y quinto escenario ocurre lo contrario. Por otro lado, para todos los escenarios, la probabilidad de cualquiera de los dos ataques simples es mayor que la del ataque doble.

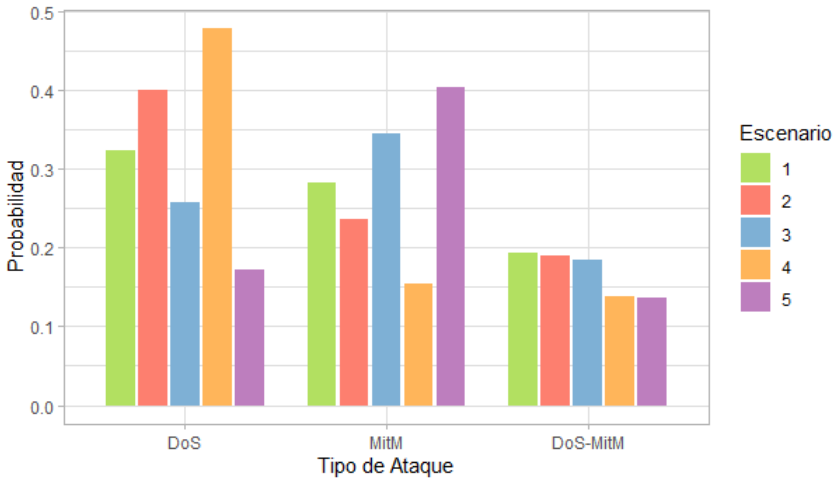


Figura 5.1: Probabilidad de ataque al nivel de dispositivos por escenarios

En la figura 5.2 se puede observar que el valor del impacto para los ataques de DoS y MitM son similares en todos los escenarios. Por otro lado, el ataque doble tiene mayor impacto que los simples en todos los escenarios.

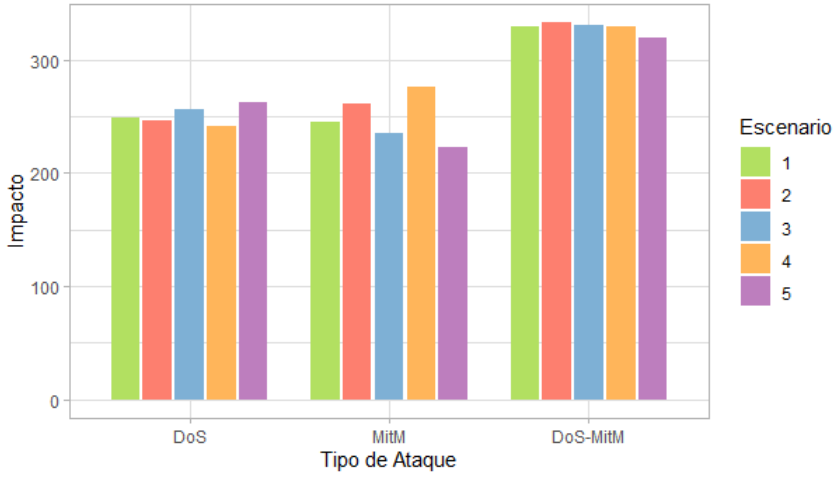


Figura 5.2: Impacto de ataque al nivel de dispositivos por escenarios

En la figura 5.3, se puede observar que predomina el ataque de DoS en el primer, segundo y cuarto escenario y el de MitM en los escenarios restantes, siguiendo la tendencia de las probabilidades de la figura 5.1. A pesar de que el impacto del ataque doble es mayor que el de los ataques simples, como el riesgo se define como el producto de la probabilidad y el impacto, el riesgo del ataque doble es menor que el de los ataques simples en la mayoría de escenarios.

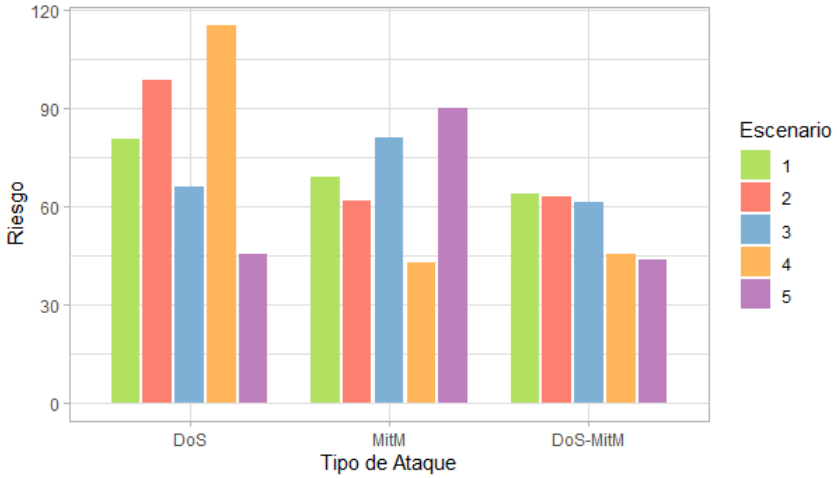


Figura 5.3: Riesgo de ataque al nivel de dispositivos por escenarios

5.2. Resultados considerando que ocurre un ataque en los niveles de administración y de router

Las figuras 5.4, 5.5, 5.6 y 5.7 muestran la probabilidad y el riesgo de los ataques simples en el nivel de dispositivos considerando la evidencia de ataque en los niveles de administración y de router.

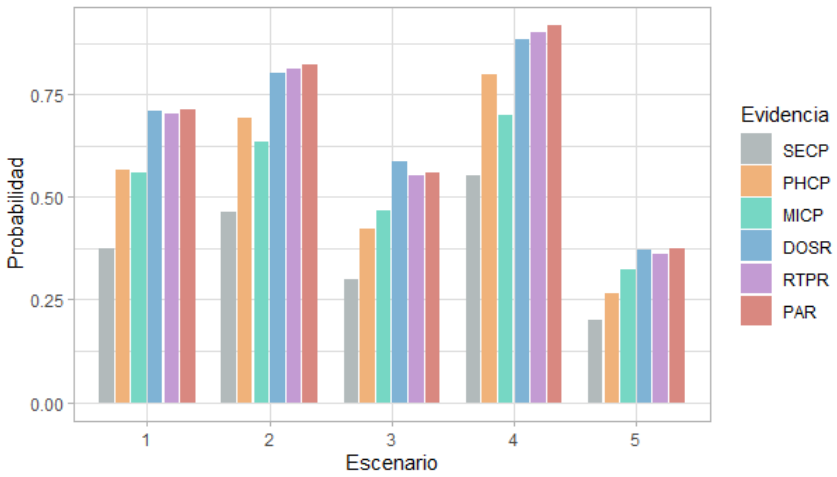


Figura 5.4: Probabilidad de ataque DoS al nivel de dispositivos en cada escenario considerando la evidencia de ataque en los demás nodos

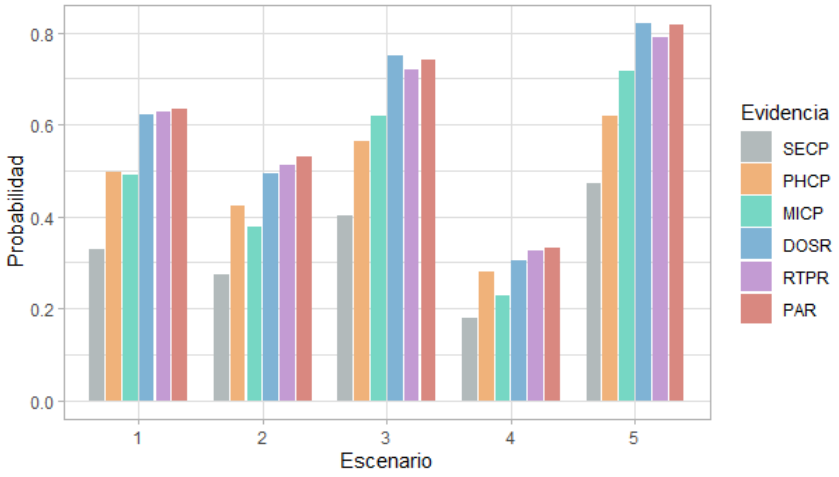


Figura 5.5: Probabilidad de ataque MitM al nivel de dispositivos en cada escenario considerando la evidencia de ataque en los demás nodos

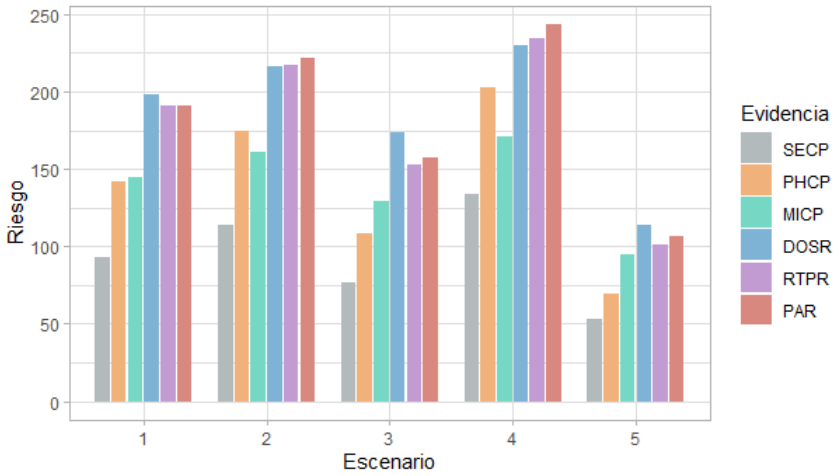


Figura 5.6: Riesgo de ataque DoS al nivel de dispositivos en cada escenario considerando la evidencia de ataque en los demás nodos

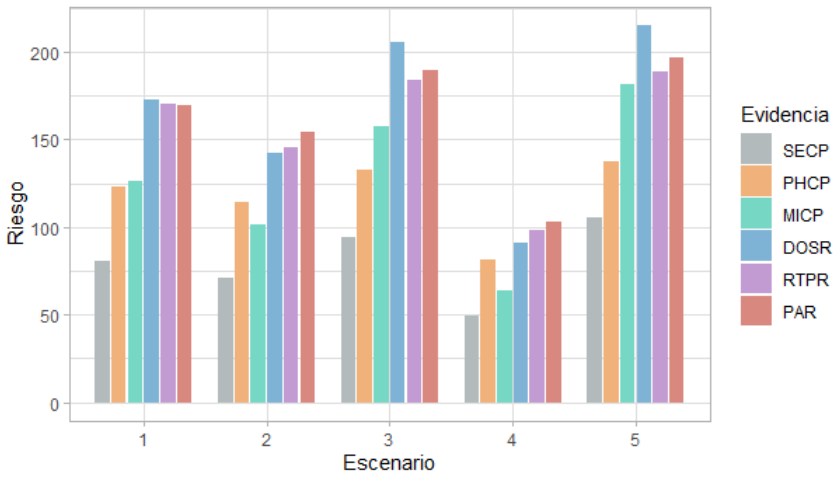


Figura 5.7: Riesgo de ataque MitM al nivel de dispositivos en cada escenario considerando la evidencia de ataque en los demás nodos

En las figuras 5.4 y 5.5 se puede observar que en el primer escenario, a pesar de que las probabilidades de selección son las mismas, el ataque de DoS al nivel de dispositivos tiene mayor probabilidad que el ataque de MitM, con cualquiera de las evidencias. Este comportamiento también ocurre en el caso del riesgo en las figuras 5.6 y 5.7.

En el segundo y cuarto escenario, tanto la probabilidad como el riesgo son mayores para el ataque de DoS al nivel de dispositivos, en comparación con el ataque de MitM; y para el tercer y quinto escenario son mayores para el ataque de MitM. Este comportamiento es esperado debido a la asignación de las probabilidades de selección para la definición de los escenarios de simulación. Sin embargo, la ventaja de este análisis es que se pueden cuantificar los valores. De esta forma, en la figura 5.6 se puede observar que cuando se da evidencia de ataques al nivel de router ($DOSR$, $RTPR$ y PAR), el ataque de DoS al nivel de dispositivos presenta riesgos superiores a 200 en el segundo y cuarto escenario; mientras que en la figura 5.7 se puede observar que para el ataque de MitM, el riesgo solo supera el valor de 200 cuando se da la evidencia de ataque $DOSR$ en el tercer y quinto escenario.

En el apéndice B se presentan gráficos complementarios correspondientes al impacto de los ataques simples y a la probabilidad, impacto y riesgo del ataque doble, considerando que ocurre un ataque en los niveles de administración y de router.

5.3. Resultados considerando que ocurren dos ataques en el nivel de administración

Con el propósito de analizar el efecto que representa aumentar evidencias en la red bayesiana, en este caso se considera que dos nodos en el grafo son atacados. El comportamiento que se desea evidenciar es que los valores de probabilidad, impacto y riesgo aumentan para todos los escenarios, con respecto al caso cuando solo se considera que ocurre un tipo de ataque y cuando no se consideran ataques.

Las figuras 5.8, 5.9 y 5.10 muestran la probabilidad, impacto y riesgo de los ataques de DoS, MitM y doble de DoS y MitM al nivel de dispositivos considerando como evidencia que ocurrieron los ataques de phishing y malware al nivel de administración ($PHCP$ y $MICP$).

En la figura 5.8 se puede observar que para el primer, segundo y cuarto escenario, el ataque de DoS tiene mayor probabilidad que el de MitM, mientras que en el resto

de escenarios ocurre lo contrario. Para todos los escenarios, la probabilidad de los ataques simples es mayor que la del ataque doble. Este comportamiento es el mismo que se observó en la figura 5.1, sin embargo, en este caso, las probabilidades son mayores para todos los escenarios debido a que se cuenta con la evidencia de que ocurrieron dos tipos de ataques.

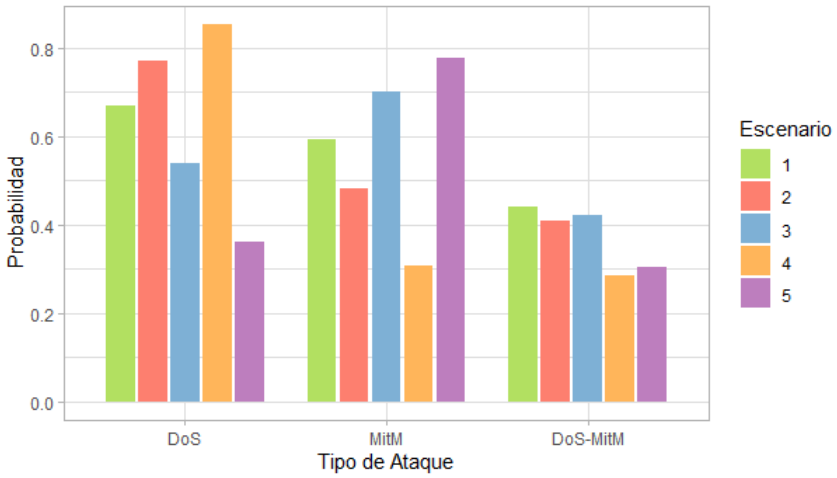


Figura 5.8: Probabilidad de ataque al nivel de dispositivos considerando la evidencia de ataques de phishing y malware en el nivel de administración, por escenarios

En la figura 5.9 se puede observar que los valores del impacto para los ataques simples son similares y el del ataque doble es mayor que el de los simples, para todos los escenarios. Además, únicamente en el quinto escenario para el ataque de MitM, el valor del impacto es inferior a 300, lo que demuestra un aumento de los valores con respecto al caso en que no se dan evidencias, de la figura 5.2, donde ninguno de los ataques simples tiene valor superior a 300.

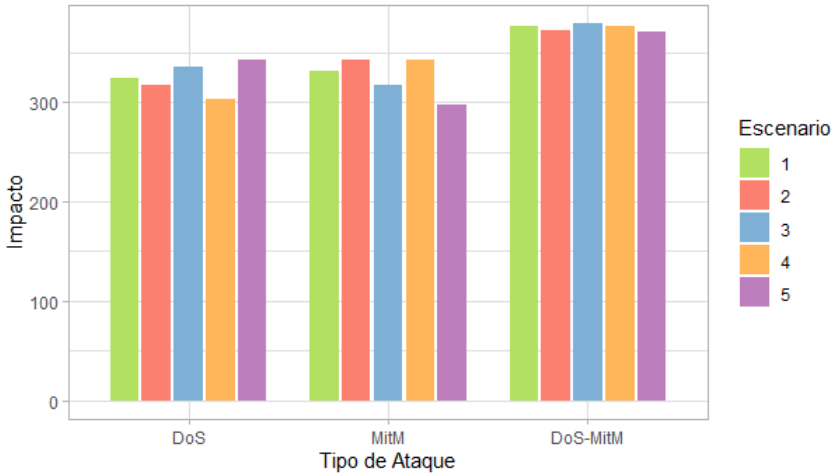


Figura 5.9: Impacto de ataque al nivel de dispositivos considerando la evidencia de ataques de phishing y malware en el nivel de administración, por escenarios

En la figura 5.10 se puede observar que predomina el ataque de DoS en el primer, segundo y cuarto escenario; y el ataque de MitM para los escenarios restantes. Además, el mayor riesgo por cada escenario es superior a 200 lo que demuestra un aumento en comparación a cuando se tenía evidencia de un solo ataque de phishing o malware al nivel de administración, que se puede ver en las figuras 5.6 y 5.7.

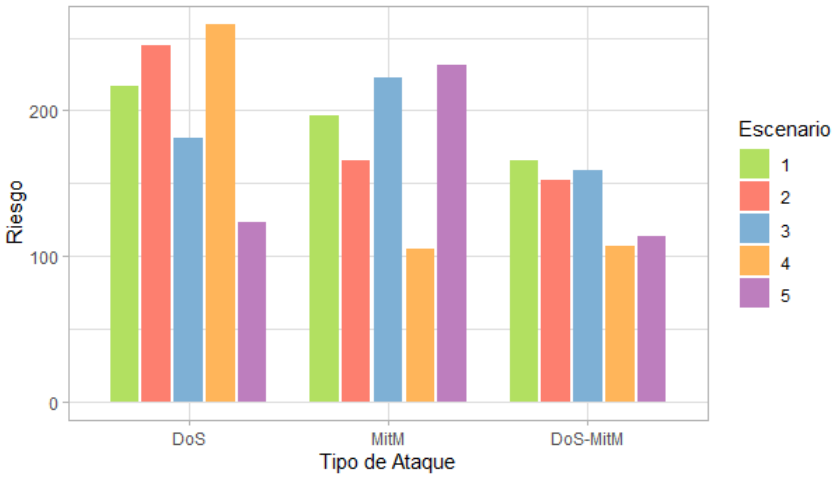
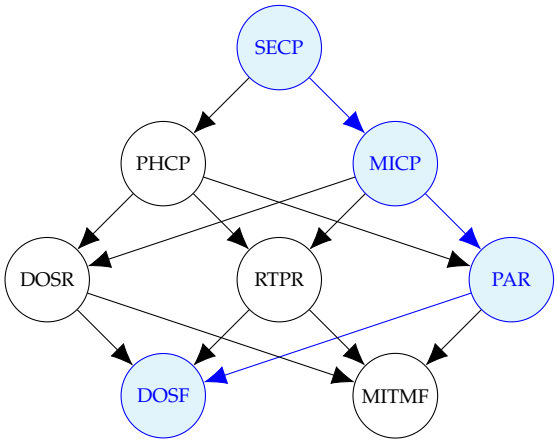


Figura 5.10: Riesgo de ataque al nivel de dispositivos considerando la evidencia de ataques de phishing y malware en el nivel de administración, por escenarios

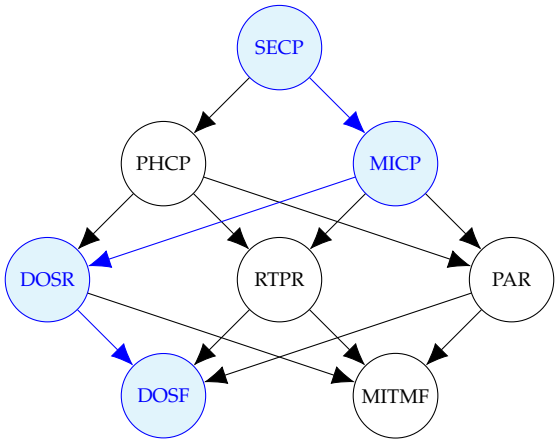
5.4. Diagramas de explicaciones más probables considerando que ocurre solo un ataque en el nivel de dispositivos

Hasta el momento se ha realizado un análisis predictivo en la red bayesiana, es decir, que se han utilizado evidencias en los nodos de los niveles de administración y de router para analizar los efectos en la probabilidad, impacto y riesgo en los nodos del nivel de dispositivos ($DOSF$ y $MITMF$). Para este resultado se realiza un análisis diagnóstico, es decir, que se considera la evidencia de que ocurre un ataque en los nodos del nivel de dispositivos y se analizan los nodos del nivel de administración y de router. Específicamente, se buscan los nodos del nivel de administración y de router que son atacados de tal forma que la probabilidad dada la evidencia sea máxima, lo que es conocido como explicación más probable y cuyas fórmulas se explicaron en la sección 4.4.

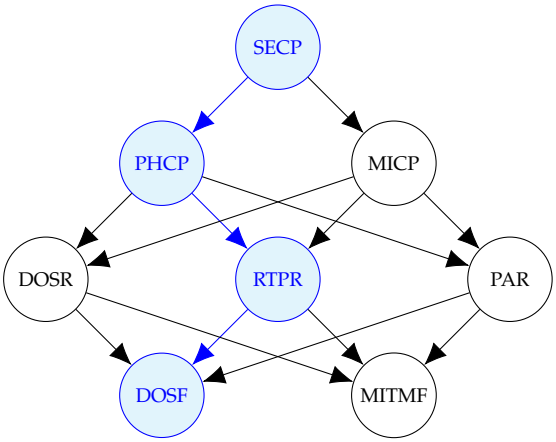
En la figura 5.11 se presentan los caminos más probables cuando se da evidencia de que ocurrió un ataque de DoS en el nivel de dispositivos y no ocurrió un ataque de MitM, considerando los 5 escenarios de simulación. Los nodos de color azul significan que son son atacados y los demás, que no lo son.



(a) Explicación más probable para DOSF en el escenario 1



(b) Explicación más probable para DOSF en los escenarios 2 y 4

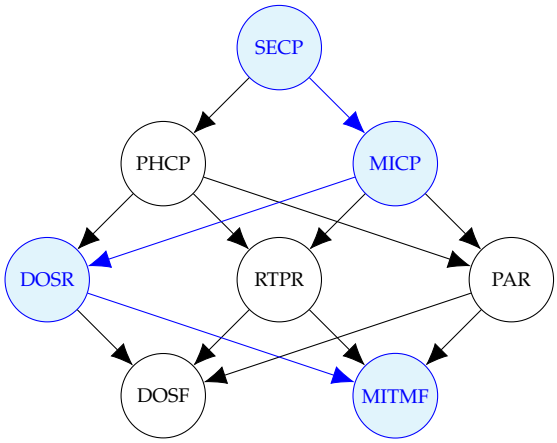


(c) Explicación más probable para DOSF en los escenarios 3 y 5

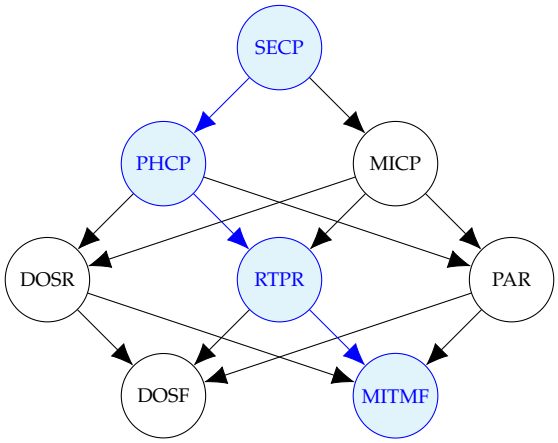
Figura 5.11: Explicaciones más probables para el nodo DOSF

Se puede observar que en este análisis, el camino que maximiza la probabilidad cuando se da evidencia de ataque DoS, en el segundo y cuarto escenario, incluye el ataque de malware al nivel de administración y de DoS al nivel de router; y en el tercer y quinto escenario incluye un ataque de phishing y RTP. Este comportamiento era esperado debido a la asignación de las probabilidades de selección, sin embargo en el primer escenario, en el que estas probabilidades son iguales para los ataques simples, se puede observar que el camino incluye un ataque persistente a nivel de router.

En la figura 5.12 se presentan los caminos más probables cuando se da evidencia de que ocurrió un ataque de MitM en el nivel de dispositivos y no ocurrió uno de DoS.



(a) Explicación más probable para MITMF en los escenarios 1, 2 y 4



(b) Explicación más probable para MITMF en los escenarios 3 y 5

Figura 5.12: Explicaciones más probables para el nodo MITMF

Se puede observar que el camino que maximiza la probabilidad cuando se da evidencia de ataque MitM, en el primer, segundo y cuarto escenario, incluye el ataque de malware al nivel de administración y de DoS al nivel de router; y en el tercer y quinto escenario incluye un ataque de phishing y RTP. En este caso, el primer escenario es igual al segundo y cuarto; sin embargo el comportamiento era esperado para los demás escenarios.

En general se puede afirmar que los nodos que conforman los caminos más probables para cada caso, siguen la tendencia que se asigna en las probabilidades de selección al definir los escenarios de simulación. En el primer escenario, al no haber una preferencia por ninguno de los ataques, cualquiera puede conformar el camino de la explicación más probable. A pesar de esto, el ataque de malware coincidió en ambos casos.

Capítulo 6

Conclusiones y recomendaciones

En el presente trabajo se implementó un modelo de evaluación de riesgo de una red de IoT en una casa inteligente utilizando una red bayesiana. El grafo dirigido acíclico de la red bayesiana fue desarrollado a partir del grafo de ataque presentado por Ibrahim y Nabulsi [6] y los parámetros de la red bayesiana fueron estimados mediante el método de máxima verosimilitud aplicado a un conjunto de datos de ataques obtenidos por simulación, en cinco diferentes escenarios. Para la simulación de eventos de ataques se considera tanto la selección de ataques como la capacidad de vulnerar los dispositivos de la red con distintos ataques. La construcción del grafo de la red bayesiana se realiza con el objetivo de evaluar el riesgo para los ataques de DoS, MitM y doble de DoS y MitM en el nivel de dispositivos, que es donde se encuentran los dispositivos que permiten la automatización de la casa inteligente y por lo general son los que individualmente tienen menores niveles de seguridad. Para la evaluación del riesgo se consideran inferencias en la red bayesiana y el impacto de los ataques en los dispositivos de la red de IoT.

De los resultados obtenidos de la implementación del modelo se obtienen las siguientes conclusiones:

- De acuerdo al análisis realizado sin considerar evidencia, el ataque de DoS es el que representa un mayor riesgo para el nivel de dispositivos; debido a que predomina en la mayor cantidad de escenarios. De esta forma, el mayor riesgo es que el atacante impida el acceso a la red por parte de los dispositivos.

- De acuerdo al análisis realizado cuando se considera un ataque como evidencia y cuando se consideran dos ataques como evidencia, los resultados en cuanto a las tendencias son similares, es decir, el ataque de DoS predomina

en la mayoría de los casos, representando de esta manera un mayor riesgo de bloqueo de los dispositivos de la red de IoT.

- El comportamiento de los ataques en cuanto a mayor probabilidad y riesgo sigue lo asignado en las probabilidades de selección para la definición de los escenarios de simulación, para el segundo, tercer, cuarto y quinto escenario; sin embargo, en el primer escenario al no haber una preferencia por ningún ataque, predomina el ataque de DoS en todos los resultados, es decir, en este caso predomina el riesgo de bloqueo de la red.

- A pesar de que el ataque doble presenta mayores valores de impacto, en relación al riesgo siempre resulta ser menor que los ataques simples; esto ocurre debido a que el riesgo es una función tanto de la probabilidad como del impacto de los ataques.

- La ventaja de este tipo de análisis es la cuantificación de los valores de riesgo de los diferentes ataques. Esto permite un análisis más detallado, lo que puede representar una mejor toma de decisiones en cuanto al ataque al que se debe dar mayor prioridad para la implementación de seguridades en la red.

- Desde el punto de vista del riesgo, para el diseño de la red de IoT en una casa inteligente se debe tomar mayor atención a los ataques de DoS que a los de MitM en el nivel de dispositivos, sin dejar de lado los ataques a los niveles de administración y de router ya que cuando se da evidencia de ataques en estos niveles, se observa un grado de incidencia en el riesgo de los ataques al nivel de dispositivos.

Finalmente, si bien los resultados evidencian que el mayor riesgo se encuentra en el ataque de DoS cuyo objetivo es bloquear la red para el usuario, sobre el de MitM cuyo objetivo es el robo de información, las necesidades del propietario de la red también influyen para priorizar el riesgo. Para dicha evaluación del riesgo se pueden establecer nuevos escenarios de simulación de tal forma que se pueda realizar un análisis específico de acuerdo a los requerimientos del propietario de la red de IoT de la casa inteligente.

Con el fin de ampliar los conocimientos en cuanto a este tipo de análisis y como trabajo a futuro, se consideran las siguientes recomendaciones:

- Es adecuado profundizar con nuevas investigaciones de tal forma que a futuro, las redes bayesianas se conviertan en una herramienta que apoyada por

simulación matemática, permita analizar riesgos de mejor manera para áreas más complejas, donde el riesgo y el costo pueden ser más altos.

- Los grafos de ataque no son utilizados únicamente para redes de IoT de casas inteligentes, sino que también tienen otras aplicaciones como ciudades inteligentes o sistemas informáticos en general; un trabajo a futuro puede ser desarrollar nuevos modelos de redes bayesianas a partir de estos grafos de ataque, basados en la metodología presentada en este trabajo. Cabe recalcar que para este tipo de análisis se requerirá mayor información proporcionada por expertos en cada área en estudio.

- En el presente trabajo se desarrolló un modelo para evaluar el riesgo de una red de IoT de una casa inteligente. La metodología expuesta tiene como propósito mostrar cómo se determina la probabilidad, impacto y riesgo de los diferentes ataques a la red. Dependiendo del ámbito de aplicación, el análisis del riesgo puede tener una mayor complejidad; como puede ser el caso de las áreas industriales donde las redes no solo involucran aspectos informáticos de datos sino también de sistemas electrónicos de control de procesos, sistemas de control de calidad, entre otros lo que incrementa el número de variables para los análisis. Otro ejemplo puede ser el área educativa donde existen riesgos por ataques hacia las bases de datos de calificaciones, archivos de registros de exámenes, perfiles de docentes, mallas curriculares, entre otros donde no solo están involucrados atacantes externos sino también internos; representando una mayor complejidad. Por este motivo se recomienda desarrollar evaluaciones del riesgo en este tipo de redes más complejas en base a la metodología presentada en este trabajo.

Bibliografía

[1] Aksu, M. U., Dilek, M. H., Tatli, E. I., Bicakci, K., Dirik, H. I., Demirezen, M. U., y Aykir, T. A quantitative CVSS-based cyber security risk assessment methodology for IT systems. En *2017 International Carnahan Conference on Security Technology (ICCST)*. IEEE, 2017.

[2] Bastos, D., Shackleton, M., y El-Moussa, F. Internet of things: A survey of technologies and security risks in smart home and city environments. En *Living in the Internet of Things: Cybersecurity of the IoT - 2018*. Institution of Engineering and Technology, 2018.

[3] Chockalingam, S., Pieters, W., Teixeira, A., y van Gelder, P. Bayesian network models in cyber security: A systematic review. En Lipmaa, H., Mitrokotsa, A., y Matulevičius, R., editors, *Secure IT Systems*, págs. 105–122, Cham, 2017. Springer International Publishing.

[4] Darwiche, A. *Modeling and Reasoning with Bayesian Networks*. Cambridge University Press, 2009.

[5] Frigault, M., Wang, L., Singhal, A., y Jajodia, S. Measuring network security using dynamic bayesian network. En *Proceedings of the 4th ACM Workshop on Quality of Protection*, QoP '08, pág. 23–30, New York, NY, USA, 2008. Association for Computing Machinery.

[6] Ibrahim, M. y Nabulsi, I. Security analysis of smart home systems applying attack graph. En *2021 Fifth World Conference on Smart Trends in Systems Security and Sustainability (WorldS4)*. IEEE, 2021.

[7] Kandasamy, K., Srinivas, S., Achuthan, K., y Rangan, V. P. IoT cyber risk: a holistic analysis of cyber risk assessment frameworks, risk vectors, and risk ranking process. *EURASIP Journal on Information Security*, 2020(1):8, May 2020.

[8] Koller, D. y Friedman, N. *Probabilistic Graphical Models: Principles and Techniques*. Adaptive Computation and Machine Learning series. MIT Press, 2009.

[9] Liang, L., Zheng, K., Sheng, Q., y Huang, X. A denial of service attack method for an IoT system. En *2016 8th International Conference on Information Technology in Medicine and Education (ITME)*. IEEE, December 2016.

[10] Liu, Y. y Man, H. Network vulnerability assessment using Bayesian networks. En Dasarathy, B. V., editor, *Data Mining, Intrusion Detection, Information Assurance, and Data Networks Security 2005*, volume 5812, págs. 61 – 71. International Society for Optics and Photonics, SPIE, 2005.

[11] Mallik, A., Ahsan, A., Shahadat, M., y Tsou, J. Man-in-the-middle-attack: Understanding in simple words. *International Journal of Data and Network Science*, págs. 77–92, 2019.

[12] Muñoz-González, L., Sgandurra, D., Barrère, M., y Lupu, E. C. Exact inference techniques for the analysis of bayesian attack graphs. *IEEE Transactions on Dependable and Secure Computing*, 16(2):231–244, 2019.

[13] NIST. Common vulnerability scoring system version 3.1 calculator. https://www.first.org/cvss/calculator/3.1.

[14] NIST. Common vulnerability scoring system version 3.1 specification document. https://www.first.org/cvss/v3.1/specification-document.

[15] Rajab, H. y Cinkelr, T. IoT based smart cities. En *2018 International Symposium on Networks, Computers and Communications (ISNCC)*. IEEE, 2018.

[16] Ross, S. *Simulation*. Elsevier, 2013.

[17] Salahdine, F. y Kaabouch, N. Social engineering attacks: A survey. *Future Internet*, 11(4), 2019.

[18] Scutari, M. y Denis, J.-B. *Bayesian Networks: With Examples in R*. Chapman and Hall/CRC, 2021.

[19] Zeng, J., Wu, S., Chen, Y., Zeng, R., y Wu, C. Survey of attack graph analysis methods from the perspective of data and knowledge processing. *Security and Communication Networks*, 2019:1–16, 2019.

Apéndice A

Informe de evaluación del riesgo

Se presenta un modelo del informe de evaluación de riesgo en la red de IoT de la casa inteligente, utilizando la metodología del presente trabajo.

Informe de Riesgo

Datos Generales	
Tipo de Aplicación:	Casa Inteligente
Tipo de Red:	Red del Internet de las Cosas (IoT)
Técnica Utilizada:	Redes bayesianas y simulación de variables aleatorias
Propietario:	Dr. Juan Pérez
Punto de Acceso:	Wi-Fi de ingreso del proveedor de servicio
Router:	Enrutador de señal a dispositivos (genérico)
Dispositivos:	Cámaras de seguridad Sistema de alarmas automatizadas Control de luces Cerraduras de puertas Sistema de seguridad con control móvil externo Asistente personal Alexa Computadora para trámites bancarios Celulares conectados a la red
Ataques considerados:	Ataques en el nivel de dispositivos: **DoS:** Bloquea el sistema y lo deja inaccesible para el usuario **MitM**: Obtiene el control de los dispositivos

Resultados de la Simulación	
Escenario 1	**Sin evidencia:** Predomina el ataque DoS para la probabilidad (32,32 %) y el riesgo (80,34); y el ataque doble para el impacto (329,23). **Una evidencia:** Predomina el ataque DoS para la probabilidad (71,25 %) y el riesgo (198,33); y el ataque doble para el impacto (348,60). **Dos evidencias:** Predomina el ataque DoS para la probabilidad (66,94 %) y el riesgo (217,16); y el ataque doble para el impacto (377,43).

Resultados de la Simulación	
Escenario 2	**Sin evidencia:** Predomina el ataque DoS para la probabilidad (39,98 %) y el riesgo (98,39); y el ataque doble para el impacto (332,68). **Una evidencia:** Predomina el ataque DoS para la probabilidad (82,23 %) y el riesgo (221,86); y el ataque doble para el impacto (347,00). **Dos evidencias:** Predomina el ataque DoS para la probabilidad (76,95 %) y el riesgo (244,49); y el ataque doble para el impacto (372,39).
Escenario 3	**Sin evidencia:** Predomina el ataque MitM para la probabilidad (34,47 %) y el riesgo (80,98); y el ataque doble para el impacto (330,90). **Una evidencia:** Predomina el ataque MitM para la probabilidad (74,98 %) y el riesgo (205,91); y el ataque doble para el impacto (354,56). **Dos evidencias:** Predomina el ataque MitM para la probabilidad (70,07 %) y el riesgo (222,83); y el ataque doble para el impacto (379,51).
Escenario 4	**Sin evidencia:** Predomina el ataque DoS para la probabilidad (47,74 %) y el riesgo (115,22); y el ataque doble para el impacto (329,79). **Una evidencia:** Predomina el ataque DoS para la probabilidad (91,79 %) y el riesgo (243,28); y el ataque doble para el impacto (346,45). **Dos evidencias:** Predomina el ataque DoS para la probabilidad (85,24 %) y el riesgo (259,07); y el ataque doble para el impacto (377,41).
Escenario 5	**Sin evidencia:** Predomina el ataque MitM para la probabilidad (40,33 %) y el riesgo (89,96); y el ataque doble para el impacto (319,17). **Una evidencia:** Predomina el ataque MitM para la probabilidad (81,99 %) y el riesgo (214,97); y el ataque doble para el impacto (350,42). **Dos evidencias:** Predomina el ataque MitM para la probabilidad (77,59 %) y el riesgo (230,99); y el ataque doble para el impacto (371,84).
Rangos:	Probabilidad entre 0 % y 100 %. Impacto entre 0 y 423,67. Riesgo entre 0 y 423,67.

Riesgo	
De acuerdo con la simulación, los tipos de ataques y dispositivos del análisis:	
1	El valor del riesgo de un ataque doble de DoS y MitM es menor que el de los ataques simples, por lo tanto, es necesario reforzar la seguridad en los dispositivos en función del ataque simple que represente un mayor riesgo.
2	El mayor riesgo se halla en el ataque de DoS que es utilizado para bloquear el acceso del usuario al sistema y a los dispositivos. Considerando la ubicación y el tipo de casa, se debe implementar seguridades para proteger las cámaras y las alarmas automatizadas de un posible bloqueo, ya que esto proporcionaría las mejores condiciones para el robo de la casa, cuyas pérdidas pueden ser altas.
3	No se puede despreciar el ataque de MitM, pero se requiere que el atacante tenga una experticia en hackeo y la prioridad es la protección de los ataques de DoS.
4	El dispositivo Alexa y las computadoras cuentan con la información bancaria del propietario y una posible intromisión con ataques de MitM sería perjudicial. Por esta razón no se debe descuidar el asegurar a estos dispositivos de ataques de MitM; sin embargo, la prioridad es el ataque DoS.

Recomendaciones	
1	Invertir en sistemas de cámaras de seguridad y alarmas más sofisticados que permitan configurar mayores controles.
2	No guardar información delicada como cuentas bancarias en dispositivos que no tienen seguridades nativas, como el asistente personal Alexa.
3	Instalar software para seguridad como antivirus y antimalware en dispositivos móviles y computadoras personales.
4	Cambiar periódicamente las contraseñas de los diferentes dispositivos de la red de IoT y que estas sean de alta seguridad.
5	Dedicar la mayor parte de la inversión en seguridad a los sistemas de cámaras y alarmas antes que al software de seguridad para los dispositivos móviles, porque el estudio muestra que el ataque de DoS representa mayor riesgo que el ataque de MitM.

El presente informe de riesgo puede ser utilizado por el propietario para que la persona responsable de automatizar la casa inteligente pueda establecer los requisitos de seguridad que deba cumplir cada dispositivo que conforma la red de IoT de la casa inteligente del Dr. Juan Pérez.

Informe del resultado de la evaluación del riesgo realizado por:

Ing. Diego Heredia

Ingeniero Matemático

Apéndice B

Gráficos complementarios

Se presentan los gráficos complementarios para los resultados considerando que ocurre un ataque en los niveles de administración y de router, es decir, los resultados de la sección 5.2. Estos gráficos están relacionados al impacto de los ataques DoS y MitM al nivel de dispositivos cuando se considera una evidencia; y a la probabilidad impacto y riesgo del ataque doble DoS-MitM cuando se considera una evidencia.

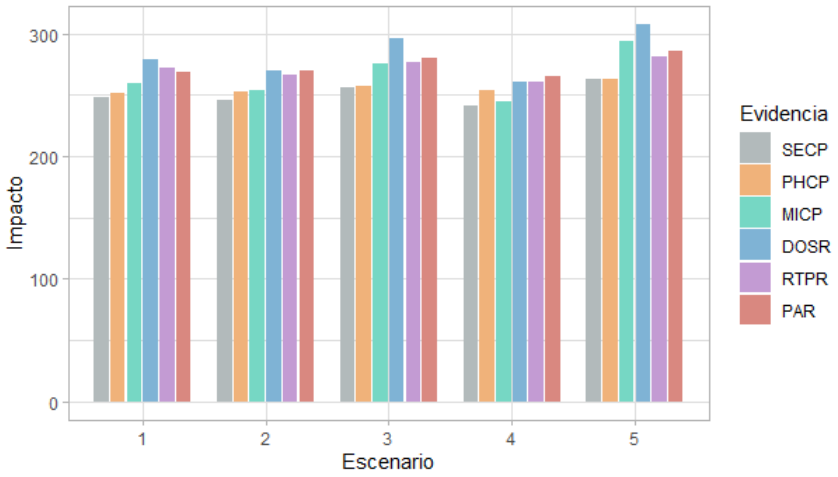


Figura B.1: Impacto de ataque DoS al nivel de dispositivos en cada escenario considerando la evidencia de ataque en los demás nodos

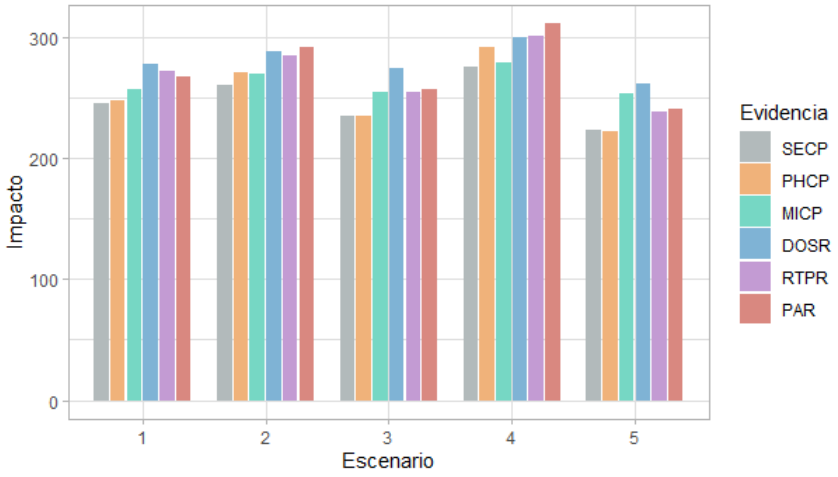


Figura B.2: Impacto de ataque MitM al nivel de dispositivos en cada escenario considerando la evidencia de ataque en los demás nodos

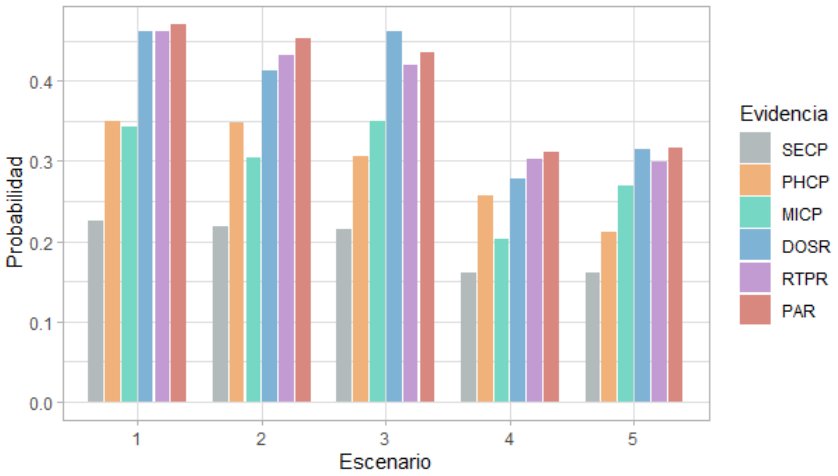


Figura B.3: Probabilidad de ataque doble DoS-MitM al nivel de dispositivos en cada escenario considerando la evidencia de ataque en los demás nodos

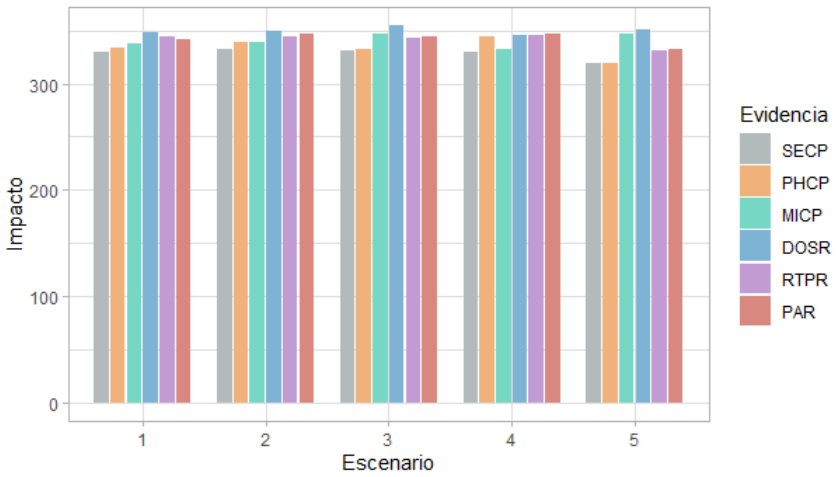


Figura B.4: Impacto de ataque doble DoS-MitM al nivel de dispositivos en cada escenario considerando la evidencia de ataque en los demás nodos

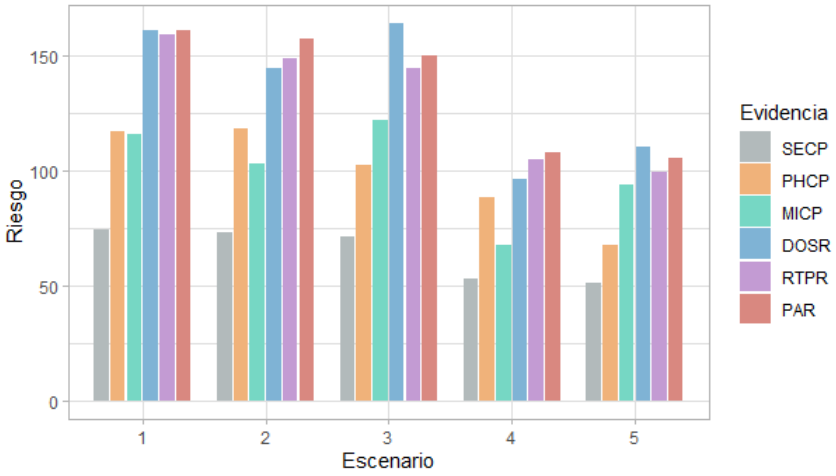


Figura B.5: Riesgo de ataque doble DoS-MitM al nivel de dispositivos en cada escenario considerando la evidencia de ataque en los demás nodos

Apéndice C

Implementación en lenguaje R

C.1. Librerías

```
library(Rgraphviz)
library(tidyverse)
library(bnlearn)
library(gRain)
```

C.2. Funciones

```
set.seed(2022)
##############################
#    Simulacion de Ataques   #
##############################
# N: Numero de ataques
# grafo: Grafo de la red bayesiana. Objeto de tipo bn
# prob_vuln: Probabilidades de vulnerabilidad (CVSS)
# y1: Probabilidades de seleccion de los nodos PHCP y MICP
# y2: Probabilidades de seleccion de los nodos DOSR, RTPR y PAR
# y3: Probabilidades de seleccion de los nodos DOSF y MITMF
datos <- function(N,grafo,prob_vuln,y1,y2,y3){
  ataques <- data.frame(matrix(0,ncol = length(grafo$nodes), nrow = 1))
  names(ataques) <- names(grafo$nodes)
  for(k in 1:N){
    res <- c()
    x1 <- bnlearn::root.nodes(grafo) #SECP
    x3 <- prob_vuln[which(prob_vuln$nodos==x1),2] #vulnerabilidad SECP
    x4 <- rbinom(n=1,size=1,prob=x3)
```

```
19      if(x4==1){
20        res <- c(res,x1) #SECP
21        sy1 <- slice_sample(.data = y1, n = 1, weight_by = y1$probs)
22        x2 <- colnames(sy1)[which(sy1==1)] #nodos seleccionados
23        x5 <- c()
24        for(i in x2){
25          x3 <- prob_vuln[which(prob_vuln$nodos==i),2] #vulnerabilidad PHCP
    -MICP
26          x4 <- rbinom(n=1,size=1,prob=x3)
27          if(x4==1){
28            res <- c(res,i)
29            x5 <- c(x5,i) #nodos seleccionados y vulnerados
30          }
31        }
32        if(length(x5)!=0){
33          x6 <- c() #Seleccion nivel router
34          x7 <- c() #Seleccion y vulnerabilidad nivel router
35          for(i in x5){
36            sy2 <- slice_sample(.data = y2, n = 1, weight_by = y2$probs) #
    seleccion de cada nodo en x5
37            x6 <- colnames(sy2)[which(sy2==1)]
38            for(i in x6){ #Vulnerabilidad x6
39              x3 <- prob_vuln[which(prob_vuln$nodos==i),2] #vulnerabilidad
    DOSR-RTPR-PAR
40              x4 <- rbinom(n=1,size=1,prob=x3)
41              if(x4==1){
42                res <- c(res,i)
43                x7 <- c(x7,i) #nodos seleccionados y vulnerados
44              }
45            }
46          }
47          x7 <- unique(x7)
48          if(length(x7)!=0){ #nivel Dispositivos
49            x8 <- c() #Seleccion nivel dispositivos
50            x9 <- c() #Seleccion y vulnerabilidad nivel dispositivos
51            for(i in x7){
52              sy3 <- slice_sample(.data = y3, n = 1, weight_by = y3$probs)
53              x8 <- colnames(sy3)[which(sy3==1)]
54              for(i in x8){ #Vulnerabilidad x8
55                x3 <- prob_vuln[which(prob_vuln$nodos==i),2] #
    vulnerabilidad DOSF-MITMF
56                x4 <- rbinom(n=1,size=1,prob=x3)
57                if(x4==1){
58                  res <- c(res,i)
```

```
                x9 <- c(x9,i) #nodos seleccionados y vulnerados
              }
            }
          }
          x9 <- unique(x9)
          res <- unique(res)
        }
      }
    }
    fila <- nodes(grafo) %in% unique(res) %>% as.numeric()
    ataques <- ataques %>% rbind(fila)
  }
  # Se elimina la primera fila
  ataques <- ataques[-c(1),]
  row.names(ataques) <- 1:nrow(ataques)
  # Transformar a factor las observaciones
  ataques <- ataques %>% mutate_if(is.numeric,as.factor)
  return(ataques)
}

############################
#   Evaluacion del Riesgo  #
############################
# Datos de entrada iguales a los de la funcion de simulacion de datos
# datos_impacto: Valor de impacto de cada nodo
evaluacion_riesgo <- function(N,grafo,prob_vuln,y1,y2,y3,datos_impacto){
  # Simulacion de N eventos de ataque a la red
  datos_ataques <- datos(N,grafo,prob_vuln,y1,y2,y3)
  # Aprendizaje de parametros por maxima verosimilitud
  bn.mle <- bn.fit(grafo, data = datos_ataques, method = "mle")
  # Impacto de ataques
  datos_ataques2 <- datos_ataques %>% mutate_if(is.factor,as.numeric)
  datos_ataques2 <- datos_ataques2-1
  datos_ataques2 <- t(t(datos_ataques2)*datos_impacto[,2])
  datos_ataques2 <- cbind(datos_ataques2, suma=rowSums(datos_ataques2))
  datos_ataques2 <- as.data.frame(datos_ataques2)
  # Evaluacion de riesgo
  # Inferencia exacta - Metodo junction tree
  junction <- compile(as.grain(bn.mle))
  ### Resultado 1
  # P(DOSF=1)
  i1_a1 <- querygrain(junction, nodes = "DOSF")[[1]][2] %>% as.numeric
  # Media de la suma de impactos
```

```
i1_a2 <- datos_ataques2 %>% filter(DOSF!=0) %>% summarise(media = mean(
  suma)) %>% as.numeric
# Riesgo DOSF
i1_a <- i1_a1*i1_a2
# P(MITMF=1)
i1_b1 <- querygrain(junction, nodes = "MITMF")[[1]][2] %>% as.numeric
# Media de la suma de impactos
i1_b2 <- datos_ataques2 %>% filter(MITMF!=0) %>% summarise(media = mean
  (suma)) %>% as.numeric
# Riesgo MITMF
i1_b <- i1_b1*i1_b2
# P(DOSF=1,MITMF01)
i1_c1 <- querygrain(junction, nodes = c("DOSF","MITMF"), type = "joint"
  )[2,2]
# Media de la suma de impactos
i1_c2 <- datos_ataques2 %>% filter(DOSF!=0 & MITMF!=0) %>% summarise(
  media = mean(suma)) %>% as.numeric
# Riesgo DOSF
i1_c <- i1_c1*i1_c2
tabla0 <- data.frame(Nodo=c("DOSF", "MITMF", "DOSF-MITMF"),
                     Probabilidad = c(i1_a1,i1_b1,i1_c1),
                     Impacto = c(i1_a2,i1_b2,i1_c2),
                     Riesgo = c(i1_a, i1_b, i1_c))
### Resultado 2
# Probabilidades
tabla_nodo <- c()
tabla_inf1 <- c()
tabla_inf2 <- c()
tabla_inf3 <- c()
for(i in nodes(grafo)[1:6]){
  ev <- setEvidence(junction, nodes = i, states = "1")
  inferencia1 <- querygrain(ev, nodes = "DOSF", type = "marginal")
  [[1]][2] %>% as.numeric
  inferencia2 <- querygrain(ev, nodes = "MITMF", type = "marginal")
  [[1]][2] %>% as.numeric
  inferencia3 <- querygrain(ev, nodes = c("DOSF","MITMF"), type = "
  joint")[2,2]
  tabla_nodo <- c(tabla_nodo, i)
  tabla_inf1 <- c(tabla_inf1, inferencia1)
  tabla_inf2 <- c(tabla_inf2, inferencia2)
  tabla_inf3 <- c(tabla_inf3, inferencia3)
}
tabla1 <- data.frame(Nodo=tabla_nodo, DOSF=tabla_inf1,
                     MITMF=tabla_inf2, DOSF_MITMF=tabla_inf3)
```

```
# Impacto
tabla_s_nodo <- c()
tabla_s_inf1 <- c()
tabla_s_inf2 <- c()
tabla_s_inf3 <- c()
for(i in nodes(grafo)[1:6]){
  score2_1 <- datos_ataques2 %>% filter((!!as.name(i))!=0 & DOSF!=0)
 %>%
    summarise(media = mean(suma)) %>% as.numeric
  score2_2 <- datos_ataques2 %>% filter((!!as.name(i))!=0 & MITMF!=0)
 %>%
    summarise(media = mean(suma)) %>% as.numeric
  score2_3 <- datos_ataques2 %>% filter((!!as.name(i))!=0 & DOSF!=0 &
 MITMF!=0) %>%
    summarise(media = mean(suma)) %>% as.numeric
  tabla_s_nodo <- c(tabla_s_nodo, i)
  tabla_s_inf1 <- c(tabla_s_inf1, score2_1)
  tabla_s_inf2 <- c(tabla_s_inf2, score2_2)
  tabla_s_inf3 <- c(tabla_s_inf3, score2_3)
}
tabla1_s <- data.frame(Nodo=tabla_s_nodo, DOSF=tabla_s_inf1,
                       MITMF=tabla_s_inf2, DOSF_MITMF=tabla_s_inf3)
# Tabla de Riesgos: Multiplicar por elementos los dos data frames
riesgo2 <- tabla1[-1]*tabla1_s[-1]
riesgo2 <- riesgo2 %>% mutate(Nodo=tabla1$Nodo)
riesgo2 <- riesgo2[c(4,1,2,3)]
Res2 <- list(tabla1,tabla1_s,riesgo2)
names(Res2) <- c("Probabilidad","Impacto","Riesgo")
### Resultado 3
# Evidencias
ev2_a <- setEvidence(junction, nodes = c("PHCP","MICP"), states = c("0"
 ,"0"))
ev2_b <- setEvidence(junction, nodes = c("PHCP","MICP"), states = c("0"
 ,"1"))
ev2_c <- setEvidence(junction, nodes = c("PHCP","MICP"), states = c("1"
 ,"0"))
ev2_d <- setEvidence(junction, nodes = c("PHCP","MICP"), states = c("1"
 ,"1"))
# Probabilidades
# Fila 1
in2_a1 <- querygrain(ev2_a, nodes = "DOSF", type = "marginal")[[1]][2]
in2_a2 <- querygrain(ev2_a, nodes = "MITMF", type = "marginal")[[1]][2]
in2_a3 <- querygrain(ev2_a, nodes = c("DOSF","MITMF"), type = "joint")
 [2,2]
```

```
# Fila 2
in2_b1 <- querygrain(ev2_b, nodes = "DOSF", type = "marginal")[[1]][2]
in2_b2 <- querygrain(ev2_b, nodes = "MITMF", type = "marginal")[[1]][2]
in2_b3 <- querygrain(ev2_b, nodes = c("DOSF","MITMF"), type = "joint")
  [2,2]
# Fila 3
in2_c1 <- querygrain(ev2_c, nodes = "DOSF", type = "marginal")[[1]][2]
in2_c2 <- querygrain(ev2_c, nodes = "MITMF", type = "marginal")[[1]][2]
in2_c3 <- querygrain(ev2_c, nodes = c("DOSF","MITMF"), type = "joint")
  [2,2]
# Fila 4
in2_d1 <- querygrain(ev2_d, nodes = "DOSF", type = "marginal")[[1]][2]
in2_d2 <- querygrain(ev2_d, nodes = "MITMF", type = "marginal")[[1]][2]
in2_d3 <- querygrain(ev2_d, nodes = c("DOSF","MITMF"), type = "joint")
  [2,2]
# Tabla Resultado
tabla2 <- data.frame(PHCP=c(0,0,1,1), MICP=c(0,1,0,1),
                     DOSF=c(in2_a1,in2_b1,in2_c1,in2_d1),
                     MITMF=c(in2_a2,in2_b2,in2_c2,in2_d2),
                     DOSF_MITMF=c(in2_a3,in2_b3,in2_c3,in2_d3))
# Impacto
# Fila 1
in2_s_a1 <- datos_ataques2 %>% filter(PHCP==0 & MICP==0 & DOSF!=0) %>%
  count() %>% as.numeric()
in2_s_a2 <- datos_ataques2 %>% filter(PHCP==0 & MICP==0 & MITMF!=0) %>%
  count() %>% as.numeric()
in2_s_a3 <- datos_ataques2 %>% filter(PHCP==0 & MICP==0 & DOSF!=0 &
 MITMF!=0) %>%
  count() %>% as.numeric()
# Fila 2
in2_s_b1 <- datos_ataques2 %>% filter(PHCP==0 & MICP!=0 & DOSF!=0) %>%
  summarise(media = mean(suma)) %>% as.numeric
in2_s_b2 <- datos_ataques2 %>% filter(PHCP==0 & MICP!=0 & MITMF!=0) %>%
  summarise(media = mean(suma)) %>% as.numeric
in2_s_b3 <- datos_ataques2 %>% filter(PHCP==0 & MICP!=0 & DOSF!=0 &
 MITMF!=0) %>%
  summarise(media = mean(suma)) %>% as.numeric
# Fila 3
in2_s_c1 <- datos_ataques2 %>% filter(PHCP!=0 & MICP==0 & DOSF!=0) %>%
  summarise(media = mean(suma)) %>% as.numeric
in2_s_c2 <- datos_ataques2 %>% filter(PHCP!=0 & MICP==0 & MITMF!=0) %>%
  summarise(media = mean(suma)) %>% as.numeric
in2_s_c3 <- datos_ataques2 %>% filter(PHCP!=0 & MICP==0 & DOSF!=0 &
 MITMF!=0) %>%
```

```
213     summarise(media = mean(suma)) %>% as.numeric
214   # Fila 4
215   in2_s_d1 <- datos_ataques2 %>% filter(PHCP!=0 & MICP!=0 & DOSF!=0) %>%
216     summarise(media = mean(suma)) %>% as.numeric
217   in2_s_d2 <- datos_ataques2 %>% filter(PHCP!=0 & MICP!=0 & MITMF!=0) %>%
218     summarise(media = mean(suma)) %>% as.numeric
219   in2_s_d3 <- datos_ataques2 %>% filter(PHCP!=0 & MICP!=0 & DOSF!=0 &
      MITMF!=0) %>%
220     summarise(media = mean(suma)) %>% as.numeric
221   # Tabla Impacto
222   tabla2_s <- data.frame(PHCP=c(0,0,1,1), MICP=c(0,1,0,1),
223                          DOSF=c(in2_s_a1,in2_s_b1,in2_s_c1,in2_s_d1),
224                          MITMF=c(in2_s_a2,in2_s_b2,in2_s_c2,in2_s_d2),
225                          DOSF_MITMF=c(in2_s_a3,in2_s_b3,in2_s_c3,in2_s_d3
      ))
226   # Tabla de Riesgos: Multiplicar por elementos los dos data frames
227   riesgo3 <- tabla2[-c(1,2)]*tabla2_s[-c(1,2)]
228   riesgo3 <- riesgo3 %>% mutate(PHCP=tabla2$PHCP, MICP=tabla2$MICP)
229   riesgo3 <- riesgo3[c(4,5,1,2,3)]
230   Res3 <- list(tabla2,tabla2_s,riesgo3)
231   names(Res3) <- c("Probabilidad","Impacto","Riesgo")
232   ### Resultado 4
233   # Nodo: DOSF y no MITMF
234   # Inferencia
235   ev3_a <- setEvidence(junction, nodes = c("DOSF","MITMF"), states = c("1
      ","0"))
236   in3_a <- querygrain(ev3_a,
237                       nodes=c("SECP","PHCP","MICP","DOSR","RTPR","PAR"),
238                       type="joint") %>% ftable %>% as.data.frame
239   #Rutas cuya probabilidad es distinta de 0
240   in3_a <- in3_a %>% filter(Freq!=0)
241   # Incluir impacto de la ruta
242   in3_a1 <- in3_a[,1:6] %>% mutate_if(is.factor,as.numeric)
243   in3_a1 <- in3_a1-1
244   in3_a1 <- t(t(in3_a1)*datos_impacto[1:6,2])
245   in3_a1 <- cbind(in3_a1, suma=rowSums(in3_a1))
246   in3_a1 <- as.data.frame(in3_a1)
247   in3_a <- in3_a %>% mutate(Score=in3_a1$suma)
248   # Incluir el riesgo
249   in3_a <- in3_a %>% mutate(Riesgo=Freq*Score)
250   # Ruta con mayor probabilidad
251   r1d1 <- in3_a[which.max(in3_a$Freq),]
252   # Ruta con mayor riesgo
253   r1d2 <- in3_a[which.max(in3_a$Riesgo),]
```

```
254   # Nodo MITMF y no DOSF
255   # Inferencia
256   ev3_b <- setEvidence(junction, nodes = c("DOSF","MITMF"), states = c("0
        ","1"))
257   in3_b <- querygrain(ev3_b,
258                       nodes=c("SECP","PHCP","MICP","DOSR","RTPR","PAR"),
259                       type="joint") %>% ftable %>% as.data.frame
260   #Rutas cuya probabilidad es distinta de 0
261   in3_b <- in3_b %>% filter(Freq!=0)
262   # Incluir score de la ruta
263   in3_b1 <- in3_b[,1:7] %>% mutate_if(is.factor,as.numeric)
264   in3_b1 <- in3_b1-1
265   in3_b1 <- t(t(in3_b1)*datos_impacto[1:7,2])
266   in3_b1 <- cbind(in3_b1, suma=rowSums(in3_b1))
267   in3_b1 <- as.data.frame(in3_b1)
268   in3_b <- in3_b %>% mutate(Score=in3_b1$suma)
269   # Incluir el riesgo
270   in3_b <- in3_b %>% mutate(Riesgo=Freq*Score)
271   # Ruta con mayor probabilidad
272   r2m1 <- in3_b[which.max(in3_b$Freq),]
273   # Ruta con mayor riesgo
274   r2m2 <- in3_b[which.max(in3_b$Riesgo),]
275   Res4 <- list(r1d1,r1d2,r2m1,r2m2)
276   names(Res4) <- c("RutaProbDoS","RutaRiesgoDoS","RutaProbMitM","
        RutaRiesgoMitM")
277   RedBayesianaDatos <- list(datos_ataques, bn.mle, junction)
278   names(RedBayesianaDatos) <- c("BaseDatos","BN.MLE","JUNCTION")
279   resultados <- list(RedBayesianaDatos, tabla0, Res2, Res3, Res4)
280   names(resultados) <- c("RedBayesiana","Resultado1", "Resultado2", "
        Resultado3", "Resultado4")
281   return(resultados)
282 }
283
284 ##################################################
285 #    Funcion para determinar el impacto (CVSS)    #
286 ##################################################
287 impacto_cvss <- function(CRd,Cd,IRd,Id,ARd,Ad,RLd){
288   CR <- case_when(
289     CRd == "L" ~ 30,
290     CRd == "M" ~ 60,
291     CRd == "H" | CRd == "N" ~ 100
292   )
293   IR <- case_when(
294     IRd == "L" ~ 30,
```

```
    IRd == "M" ~ 60,
    IRd == "H" | IRd == "N" ~ 100
  )
  AR <- case_when(
    ARd == "L" ~ 30,
    ARd == "M" ~ 60,
    ARd == "H" | ARd == "N" ~ 100
  )
  C <- case_when(
    Cd == "C" ~ 1,
    Cd == "P" ~ 0.5,
    Cd == "N" ~ 0
  )
  I <- case_when(
    Id == "C" ~ 1,
    Id == "P" ~ 0.5,
    Id == "N" ~ 0
  )
  A <- case_when(
    Ad == "C" ~ 1,
    Ad == "P" ~ 0.5,
    Ad == "N" ~ 0
  )
  RL <- case_when(
    RLd == "OF" ~ 0.15,
    RLd == "TF" ~ 0.1,
    RLd == "W" ~ 0.05,
    RLd == "U" | RLd == "N"~ 0
  )
  resultado <- ((CR*C+IR*I+AR*A)/3)*(1-RL)
  return(resultado)
}
```

C.3. Implementación del modelo

```
# Grafo
grafo <- empty.graph(nodes = c("SECP","PHCP","MICP","DOSR", "RTPR",
                               "PAR","DOSF","MITMF"))
arc.set <- matrix(c("SECP", "PHCP",
                    "SECP", "MICP",
                    "PHCP", "DOSR",
                    "PHCP", "RTPR",
```

```
                        "PHCP", "PAR",
                        "MICP", "DOSR",
                        "MICP", "RTPR",
                        "MICP", "PAR",
                        "DOSR", "DOSF",
                        "DOSR", "MITMF",
                        "RTPR", "DOSF",
                        "RTPR", "MITMF",
                        "PAR", "DOSF",
                        "PAR", "MITMF"),
                      byrow = TRUE, ncol = 2,
                      dimnames = list(NULL, c("from", "to")))
arcs(grafo) <- arc.set
graphviz.plot(grafo)
# Impacto en los nodos:
Imp_SECP <- impacto_cvss("M","C","L","N","L","N","U")
Imp_PHCP <- impacto_cvss("M","C","L","P","L","N","OF")
Imp_MICP <- impacto_cvss("M","C","M","P","M","P","OF")
Imp_DOSR <- impacto_cvss("H","C","M","P","H","C","W")
Imp_RTPR <- impacto_cvss("H","C","M","P","M","C","W")
Imp_PAR <- impacto_cvss("H","C","L","P","M","C","W")
Imp_DOSF <- impacto_cvss("H","C","M","C","H","C","U")
Imp_MITMF <- impacto_cvss("M","C","M","C","H","C","U")
# Impacto:
datos_impacto <- data.frame(nodos = nodes(grafo),
                            score = c(Imp_SECP, #SECP
                                      Imp_PHCP, #PHCP
                                      Imp_MICP, #MICP
                                      Imp_DOSR, #DOSR
                                      Imp_RTPR, #RTPR
                                      Imp_PAR,  #PAR
                                      Imp_DOSF, #DOSF
                                      Imp_MITMF #MITMF
                                      ))
# Probabilidades de vulnerabilidad en cada nodo
prob_vuln <- data.frame(nodos= nodes(grafo),
                        cvss_scores = c(0.86, #SECP
                                        0.82, #PHCP
                                        0.88, #MICP
                                        0.90, #DOSR
                                        0.90, #RTPR
                                        0.90, #PAR
                                        0.90, #DOSF
                                        0.76 #MITMF
```

```
                                        ))
N <- 10000
# Escenarios de Simulacion
# Escenario 1
y1.1 <- data.frame(PHCP = rep(0:1, each = 2),
                   MICP = rep(0:1, times=2))
y1.1 <- mutate(y1.1, probs = c(0,45,45,10))
y2.1 <- data.frame(DOSR = rep(0:1, each = 4),
                   RTPR = rep(0:1, each = 2, times=2),
                   PAR = rep(0:1, times = 4))
y2.1 <- mutate(y2.1, probs = c(0,25,25,7,25,7,7,4))
y3.1 <- data.frame(DOSF = rep(0:1, each = 2),
                   MITMF = rep(0:1, times=2))
y3.1 <- mutate(y3.1, probs = c(0,45,45,10))
riesgo_e1 <- evaluacion_riesgo(N,grafo,prob_vuln,y1.1,y2.1,y3.1,datos_
    impacto)
# Escenario 2
y1.2 <- data.frame(PHCP = rep(0:1, each = 2),
                   MICP = rep(0:1, times=2))
y1.2 <- mutate(y1.2, probs = c(0,60,30,10))
y2.2 <- data.frame(DOSR = rep(0:1, each = 4),
                   RTPR = rep(0:1, each = 2, times=2),
                   PAR = rep(0:1, times = 4))
y2.2 <- mutate(y2.2, probs = c(0,20,25,6,30,7,8,4))
y3.2 <- data.frame(DOSF = rep(0:1, each = 2),
                   MITMF = rep(0:1, times=2))
y3.2 <- mutate(y3.2, probs = c(0,30,60,10))
riesgo_e2 <- evaluacion_riesgo(N,grafo,prob_vuln,y1.2,y2.2,y3.2,datos_
    impacto)
# Escenario 3
y1.3 <- data.frame(PHCP = rep(0:1, each = 2),
                   MICP = rep(0:1, times=2))
y1.3 <- mutate(y1.3, probs = c(0,30,60,10))
y2.3 <- data.frame(DOSR = rep(0:1, each = 4),
                   RTPR = rep(0:1, each = 2, times=2),
                   PAR = rep(0:1, times = 4))
y2.3 <- mutate(y2.3, probs = c(0,25,30,8,20,6,7,4))
y3.3 <- data.frame(DOSF = rep(0:1, each = 2),
                   MITMF = rep(0:1, times=2))
y3.3 <- mutate(y3.3, probs = c(0,60,30,10))
riesgo_e3 <- evaluacion_riesgo(N,grafo,prob_vuln,y1.3,y2.3,y3.3,datos_
    impacto)
# Escenario 4
y1.4 <- data.frame(PHCP = rep(0:1, each = 2),
```

```
                    MICP = rep(0:1, times=2))
y1.4 <- mutate(y1.4, probs = c(0,75,15,10))
y2.4 <- data.frame(DOSR = rep(0:1, each = 4),
                    RTPR = rep(0:1, each = 2, times=2),
                    PAR = rep(0:1, times = 4))
y2.4 <- mutate(y2.4, probs = c(0,15,25,5,35,7,9,4))
y3.4 <- data.frame(DOSF = rep(0:1, each = 2),
                    MITMF = rep(0:1, times=2))
y3.4 <- mutate(y3.4, probs = c(0,15,75,10))
riesgo_e4 <- evaluacion_riesgo(N,grafo,prob_vuln,y1.4,y2.4,y3.4,datos_
    impacto)
# Escenario 5
y1.5 <- data.frame(PHCP = rep(0:1, each = 2),
                    MICP = rep(0:1, times=2))
y1.5 <- mutate(y1.5, probs = c(0,15,75,10))
y2.5 <- data.frame(DOSR = rep(0:1, each = 4),
                    RTPR = rep(0:1, each = 2, times=2),
                    PAR = rep(0:1, times = 4))
y2.5 <- mutate(y2.5, probs = c(0,25,35,9,15,5,7,4))
y3.5 <- data.frame(DOSF = rep(0:1, each = 2),
                    MITMF = rep(0:1, times=2))
y3.5 <- mutate(y3.5, probs = c(0,75,15,10))
riesgo_e5 <- evaluacion_riesgo(N,grafo,prob_vuln,y1.5,y2.5,y3.5,datos_
    impacto)
```

C.4. Gráficos

```
# 1. Sin evidencia
graf1df <- data.frame(Ataque=rep(c("DoS","MitM","DoS-MitM"), 5),
                      Escenario=rep(1:5, each = 3),
                      Probabilidad=c(riesgo_e1$Resultado1[,2],
                                     riesgo_e2$Resultado1[,2],
                                     riesgo_e3$Resultado1[,2],
                                     riesgo_e4$Resultado1[,2],
                                     riesgo_e5$Resultado1[,2]),
                      Impacto=c(riesgo_e1$Resultado1[,3],
                                riesgo_e2$Resultado1[,3],
                                riesgo_e3$Resultado1[,3],
                                riesgo_e4$Resultado1[,3],
                                riesgo_e5$Resultado1[,3]),
                      Riesgo=c(riesgo_e1$Resultado1[,4],
                               riesgo_e2$Resultado1[,4],
```

```
                                riesgo_e3$Resultado1[,4],
                                riesgo_e4$Resultado1[,4],
                                riesgo_e5$Resultado1[,4]))
ggplot(data=graf1df, aes(x=factor(Ataque,
                                  levels=c("DoS","MitM","DoS-MitM")),
                         y=round(Probabilidad,4),
                         fill=factor(Escenario))) +
  geom_bar(stat="identity", position=position_dodge2(), width=0.8) +
  scale_fill_manual(values=c("#b2e061","#fd7f6f","#7eb0d5","#ffb55a","#
    bd7ebe"))+
  theme_light() +
  xlab("Tipo de Ataque") + ylab("Probabilidad") +
  labs(fill = "Escenario")

ggplot(data=graf1df, aes(x=factor(Ataque,
                                  levels=c("DoS","MitM","DoS-MitM")),
                         y=round(Impacto,4),
                         fill=factor(Escenario))) +
  geom_bar(stat="identity", position=position_dodge2(), width = 0.8) +
  scale_fill_manual(values=c("#b2e061","#fd7f6f","#7eb0d5","#ffb55a","#
    bd7ebe"))+
  theme_light() +
  xlab("Tipo de Ataque") + ylab("Impacto") +
  labs(fill = "Escenario")

ggplot(data=graf1df, aes(x=factor(Ataque,
                                  levels=c("DoS","MitM","DoS-MitM")),
                         y=round(Riesgo,4),
                         fill=factor(Escenario))) +
  geom_bar(stat="identity", position=position_dodge2(), width=0.8) +
  scale_fill_manual(values=c("#b2e061","#fd7f6f","#7eb0d5","#ffb55a","#
    bd7ebe"))+
  theme_light() +
  xlab("Tipo de Ataque") + ylab("Riesgo") +
  labs(fill = "Escenario")

# 2. Una evidencia
graf2df <- data.frame(Escenario = rep(1:5,each=6),
                      Evidencia=rep(c("SECP","PHCP","MICP","DOSR","RTPR",
    "PAR"),5),
                      ProbDoS=c(riesgo_e1$Resultado2$Probabilidad[,2],
                                riesgo_e2$Resultado2$Probabilidad[,2],
                                riesgo_e3$Resultado2$Probabilidad[,2],
                                riesgo_e4$Resultado2$Probabilidad[,2],
```

```
                                    riesgo_e5$Resultado2$Probabilidad[,2]),
                        ProbMitM=c(riesgo_e1$Resultado2$Probabilidad[,3],
                                   riesgo_e2$Resultado2$Probabilidad[,3],
                                   riesgo_e3$Resultado2$Probabilidad[,3],
                                   riesgo_e4$Resultado2$Probabilidad[,3],
                                   riesgo_e5$Resultado2$Probabilidad[,3]),
                        ProbDM=c(riesgo_e1$Resultado2$Probabilidad[,4],
                                 riesgo_e2$Resultado2$Probabilidad[,4],
                                 riesgo_e3$Resultado2$Probabilidad[,4],
                                 riesgo_e4$Resultado2$Probabilidad[,4],
                                 riesgo_e5$Resultado2$Probabilidad[,4]),
                        ImpDoS=c(riesgo_e1$Resultado2$Impacto[,2],
                                 riesgo_e2$Resultado2$Impacto[,2],
                                 riesgo_e3$Resultado2$Impacto[,2],
                                 riesgo_e4$Resultado2$Impacto[,2],
                                 riesgo_e5$Resultado2$Impacto[,2]),
                        ImpMitM=c(riesgo_e1$Resultado2$Impacto[,3],
                                  riesgo_e2$Resultado2$Impacto[,3],
                                  riesgo_e3$Resultado2$Impacto[,3],
                                  riesgo_e4$Resultado2$Impacto[,3],
                                  riesgo_e5$Resultado2$Impacto[,3]),
                        ImpDM=c(riesgo_e1$Resultado2$Impacto[,4],
                                riesgo_e2$Resultado2$Impacto[,4],
                                riesgo_e3$Resultado2$Impacto[,4],
                                riesgo_e4$Resultado2$Impacto[,4],
                                riesgo_e5$Resultado2$Impacto[,4]),
                        RiesgoDoS=c(riesgo_e1$Resultado2$Riesgo[,2],
                                    riesgo_e2$Resultado2$Riesgo[,2],
                                    riesgo_e3$Resultado2$Riesgo[,2],
                                    riesgo_e4$Resultado2$Riesgo[,2],
                                    riesgo_e5$Resultado2$Riesgo[,2]),
                        RiesgoMitM=c(riesgo_e1$Resultado2$Riesgo[,3],
                                     riesgo_e2$Resultado2$Riesgo[,3],
                                     riesgo_e3$Resultado2$Riesgo[,3],
                                     riesgo_e4$Resultado2$Riesgo[,3],
                                     riesgo_e5$Resultado2$Riesgo[,3]),
                        RiesgoDM=c(riesgo_e1$Resultado2$Riesgo[,4],
                                   riesgo_e2$Resultado2$Riesgo[,4],
                                   riesgo_e3$Resultado2$Riesgo[,4],
                                   riesgo_e4$Resultado2$Riesgo[,4],
                                   riesgo_e5$Resultado2$Riesgo[,4]))
color_barras <- c("#B2BABB", "#F0B27A", "#76D7C4", "#7FB3D5", "#C39BD3",
    "#D98880")
# 3 graficos de probabilidad
```

```
ggplot(data=graf2df, aes(x=factor(Escenario),
                         y=round(ProbDoS,4),
                         fill=factor(Evidencia, levels=c("SECP","PHCP","
    MICP","DOSR","RTPR","PAR")))) +
  geom_bar(stat="identity", position=position_dodge2(), width=0.8) +
  scale_fill_manual(values=color_barras)+
  theme_light() +
  xlab("Escenario") + ylab("Probabilidad") +
  labs(fill="Evidencia")

ggplot(data=graf2df, aes(x=factor(Escenario),
                         y=round(ProbMitM,4),
                         fill=factor(Evidencia, levels=c("SECP","PHCP","
    MICP","DOSR","RTPR","PAR")))) +
  geom_bar(stat="identity", position=position_dodge2(), width=0.8) +
  scale_fill_manual(values=color_barras)+
  theme_light() +
  xlab("Escenario") + ylab("Probabilidad") +
  labs(fill="Evidencia")

ggplot(data=graf2df, aes(x=factor(Escenario),
                         y=round(ProbDM,4),
                         fill=factor(Evidencia, levels=c("SECP","PHCP","
    MICP","DOSR","RTPR","PAR")))) +
  geom_bar(stat="identity", position=position_dodge2(), width=0.8) +
  scale_fill_manual(values=color_barras)+
  theme_light() +
  xlab("Escenario") + ylab("Probabilidad") +
  labs(fill="Evidencia")

# 3 graficos de impacto
ggplot(data=graf2df, aes(x=factor(Escenario),
                         y=round(ImpDoS,4),
                         fill=factor(Evidencia, levels=c("SECP","PHCP","
    MICP","DOSR","RTPR","PAR")))) +
  geom_bar(stat="identity", position=position_dodge2(), width=0.8) +
  scale_fill_manual(values=color_barras)+
  theme_light() +
  xlab("Escenario") + ylab("Impacto") +
  labs(fill="Evidencia")

ggplot(data=graf2df, aes(x=factor(Escenario),
                         y=round(ImpMitM,4),
```

```
                          fill=factor(Evidencia, levels=c("SECP","PHCP","
    MICP","DOSR","RTPR","PAR")))) +
  geom_bar(stat="identity", position=position_dodge2(), width=0.8) +
  scale_fill_manual(values=color_barras)+
  theme_light() +
  xlab("Escenario") + ylab("Impacto") +
  labs(fill="Evidencia")

ggplot(data=graf2df, aes(x=factor(Escenario),
                         y=round(ImpDM,4),
                         fill=factor(Evidencia, levels=c("SECP","PHCP","
    MICP","DOSR","RTPR","PAR")))) +
  geom_bar(stat="identity", position=position_dodge2(), width=0.8) +
  scale_fill_manual(values=color_barras)+
  theme_light() +
  xlab("Escenario") + ylab("Impacto") +
  labs(fill="Evidencia")

# 3 graficos de riesgo
ggplot(data=graf2df, aes(x=factor(Escenario),
                         y=round(RiesgoDoS,4),
                         fill=factor(Evidencia, levels=c("SECP","PHCP","
    MICP","DOSR","RTPR","PAR")))) +
  geom_bar(stat="identity", position=position_dodge2(), width=0.8) +
  scale_fill_manual(values=color_barras)+
  theme_light() +
  xlab("Escenario") + ylab("Riesgo") +
  labs(fill="Evidencia")

ggplot(data=graf2df, aes(x=factor(Escenario),
                         y=round(RiesgoMitM,4),
                         fill=factor(Evidencia, levels=c("SECP","PHCP","
    MICP","DOSR","RTPR","PAR")))) +
  geom_bar(stat="identity", position=position_dodge2(), width=0.8) +
  scale_fill_manual(values=color_barras)+
  theme_light() +
  xlab("Escenario") + ylab("Riesgo") +
  labs(fill="Evidencia")

ggplot(data=graf2df, aes(x=factor(Escenario),
                         y=round(RiesgoDM,4),
                         fill=factor(Evidencia, levels=c("SECP","PHCP","
    MICP","DOSR","RTPR","PAR")))) +
  geom_bar(stat="identity", position=position_dodge2(), width=0.8) +
```

```
177   scale_fill_manual(values=color_barras)+
178   theme_light() +
179   xlab("Escenario") + ylab("Riesgo") +
180   labs(fill="Evidencia")
181
182 # 3. Dos evidencias
183 graf3df <- data.frame(Ataque=rep(c("DoS","MitM","DoS-MitM"), 5),
184                       Escenario=rep(1:5, each = 3),
185                       Probabilidad=
186                         unlist(c(riesgo_e1$Resultado3[[1]][4,3:5],
187                                  riesgo_e2$Resultado3[[1]][4,3:5],
188                                  riesgo_e3$Resultado3[[1]][4,3:5],
189                                  riesgo_e4$Resultado3[[1]][4,3:5],
190                                  riesgo_e5$Resultado3[[1]][4,3:5])),
191                       Impacto=
192                         unlist(c(riesgo_e1$Resultado3[[2]][4,3:5],
193                                  riesgo_e2$Resultado3[[2]][4,3:5],
194                                  riesgo_e3$Resultado3[[2]][4,3:5],
195                                  riesgo_e4$Resultado3[[2]][4,3:5],
196                                  riesgo_e5$Resultado3[[2]][4,3:5])),
197                       Riesgo=
198                         unlist(c(riesgo_e1$Resultado3[[3]][4,3:5],
199                                  riesgo_e2$Resultado3[[3]][4,3:5],
200                                  riesgo_e3$Resultado3[[3]][4,3:5],
201                                  riesgo_e4$Resultado3[[3]][4,3:5],
202                                  riesgo_e5$Resultado3[[3]][4,3:5])))
203
204 ggplot(data=graf3df, aes(x=factor(Ataque,
205                                   levels=c("DoS","MitM","DoS-MitM")),
206                          y=round(Probabilidad,4),
207                          fill=factor(Escenario))) +
208   geom_bar(stat="identity", position=position_dodge2(), width=0.8) +
209   scale_fill_manual(values=c("#b2e061","#fd7f6f","#7eb0d5","#ffb55a","#
      bd7ebe"))+
210   theme_light() +
211   xlab("Tipo de Ataque") + ylab("Probabilidad") +
212   labs(fill = "Escenario")
213
214 ggplot(data=graf3df, aes(x=factor(Ataque,
215                                   levels=c("DoS","MitM","DoS-MitM")),
216                          y=round(Impacto,4),
217                          fill=factor(Escenario))) +
218   geom_bar(stat="identity", position=position_dodge2(), width = 0.8) +
```

```
219   scale_fill_manual(values=c("#b2e061","#fd7f6f","#7eb0d5","#ffb55a","#
        bd7ebe"))+
220   theme_light() +
221   xlab("Tipo de Ataque") + ylab("Impacto") +
222   labs(fill = "Escenario")
223
224 ggplot(data=graf3df, aes(x=factor(Ataque,
225                                   levels=c("DoS","MitM","DoS-MitM")),
226                          y=round(Riesgo,4),
227                          fill=factor(Escenario))) +
228   geom_bar(stat="identity", position=position_dodge2(), width=0.8) +
229   scale_fill_manual(values=c("#b2e061","#fd7f6f","#7eb0d5","#ffb55a","#
        bd7ebe"))+
230   theme_light() +
231   xlab("Tipo de Ataque") + ylab("Riesgo") +
232   labs(fill = "Escenario")
```