UNIVERSIDAD POLITÉCNICA DE MADRID

ESCUELA TÉCNICA SUPERIOR
DE INGENIEROS DE TELECOMUNICACIÓN

GRADO EN INGENIERÍA DE TECNOLOGÍAS Y SERVICIOS DE TELECOMUNICACIÓN

TRABAJO FIN DE GRADO

DESARROLLO DE UN SISTEMA DE OPTIMIZACIÓN DE CONTRAMEDIDAS EN LA GESTIÓN DE RIESGOS DE CIBERSEGURIDAD

JAVIER MONTESINOS MARTÍ

2025

GRADO EN INGENIERÍA DE TECNOLOGÍAS Y SERVICIOS DE TELECOMUNICACIÓN

TRABAJO FIN DE GRADO

Título: Desarrollo de un sistema de optimización de contramedidas en la gestión de riesgos de ciberseguridad
Autor: D. Javier Montesinos Martí
Tutor: Dña. Carmen Sánchez Zas
Ponente: D. ……
Departamento: Departamento de Ingeniería de Sistemas Telemáticos

MIEMBROS DEL TRIBUNAL

Presidente: D. ……

Vocal: D. ……..

Secretario: D. ……..

Suplente: D. ……..

Los miembros del tribunal arriba nombrados acuerdan otorgar la calificación de: ……

Madrid, a de de 20…

UNIVERSIDAD POLITÉCNICA DE MADRID

ESCUELA TÉCNICA SUPERIOR
DE INGENIEROS DE TELECOMUNICACIÓN

GRADO EN INGENIERÍA DE TECNOLOGÍAS Y SERVICIOS DE TELECOMUNICACIÓN

TRABAJO FIN DE GRADO

DESARROLLO DE UN SISTEMA DE OPTIMIZACIÓN DE CONTRAMEDIDAS EN LA GESTIÓN DE RIESGOS DE CIBERSEGURIDAD

JAVIER MONTESINOS MARTÍ

2025

RESUMEN

En la actualidad no se concibe el mundo y el paradigma social sin hablar de ciberseguridad. Este fenómeno está presente tanto en el ámbito personal, en ordenadores, móviles y la red de dispositivos del hogar, como en la vida laboral. También se manifiesta en las infraestructuras informáticas que componen los equipos de una organización. En este último caso, una protección insuficiente expone a las compañías a ciberataques, causando pérdidas económicas significativas o exponiendo datos sensibles. A esto se suma la evolución continua de la tecnología, que hace que las amenazas sean cada vez más difíciles de mitigar. Por ello, resulta fundamental contar con un plan de gestión de riesgos que reduzca el impacto de estos ataques y blinde las redes informáticas.

Este Trabajo de Fin de Grado propone un sistema de selección de contramedidas basado en la optimización del riesgo residual de una red concreta y de cada uno de los activos que la forman. Además, considera otros factores, como el coste de cada contramedida y un presupuesto máximo para su implementación.

Se ha realizado un estudio previo sobre el funcionamiento actual de la gestión de riesgos y la implementación de planes de contramedidas para entender el problema, siguiendo de cerca la metodología MAGERIT, enfocada en el análisis y gestión de riesgos en sistemas de información.

El objetivo consiste en seleccionar de forma óptima una serie de contramedidas sobre una red definida con un determinado número de equipos interconectados entre sí, identificando las vulnerabilidades de cada uno y su exposición a los ciberataques.

Este proyecto se implementa en Python, utilizando la librería PuLP para modelar y resolver problemas de optimización mediante programación lineal y NetworkX para representar la red como un grafo, lo que permite un mejor análisis visual.

Para validar el funcionamiento del sistema, se llevan a cabo simulaciones en distintos escenarios de red, aplicando las contramedidas seleccionadas de manera óptima. A través de estas simulaciones. Se espera observar una reducción cuantificable del riesgo residual en función de diferentes restricciones presupuestarias, así como una mejora en la priorización y asignación de recursos.

El análisis de los resultados permite identificar tendencias y patrones que faciliten la toma de decisiones estratégicas, optimizando la relación entre la inversión realizada y la mitigación del riesgo. De este modo, se desarrolla una herramienta eficiente que contribuye a mejorar la gestión de riesgos en ciberseguridad.

SUMMARY

Nowadays, the world and the social paradigm are inconceivable without talking about cybersecurity. This phenomenon is present both in the personal sphere, in computers, mobiles and the network of devices at home, and in working life. It also manifests itself in the IT infrastructures that make up an organisation's equipment. In the latter case, insufficient protection exposes companies to cyber-attacks, causing significant economic losses or exposing sensitive data. Added to this is the continuous evolution of technology, which makes threats increasingly difficult to mitigate. It is therefore essential to have a risk management plan that reduces the impact of these attacks and shields computer networks.

This Final Degree Project proposes a countermeasure selection system based on the optimisation of the residual risk of a specific network and of each of its assets. It also considers other factors, such as the cost of each countermeasure and a maximum budget for its implementation.

A preliminary study on the current functioning of risk management and the implementation of countermeasure plans has been carried out to understand the problem, closely following the MAGERIT methodology, focused on the analysis and management of risks in information systems.

The objective is to optimally select a series of countermeasures on a defined network with a certain number of interconnected computers, identifying the vulnerabilities of each one and their exposure to cyber-attacks.

This project is implemented in Python, using the PuLP library to model and solve optimisation problems using linear programming and NetworkX to represent the network as a graph, which allows for better visual analysis.

To validate the performance of the system, simulations are carried out in different network scenarios, applying the selected countermeasures in an optimal way. Through these simulations. It is expected to observe a measurable reduction of residual risk under different budget constraints, as well as an improvement in prioritisation and resource allocation.

The analysis of the results makes it possible to identify trends and patterns that facilitate strategic decision-making, optimising the relationship between investment and risk mitigation. In this way, an efficient tool is developed that contributes to improving cybersecurity risk management.

PALABRAS CLAVE

Ciberseguridad, Gestión de riesgos, MAGERIT, PILAR, Contramedidas, Optimización

KEYWORDS

Cybersecurity, Risk management, MAGERIT, PILAR, Countermeasures, Optimization

AGRADECIMIENTOS

Me gustaría aprovechar este espacio para agradecer a todas las personas que han hecho posible la realización de este proyecto.

A mi familia, papá, mamá, Ana, gracias por vuestro apoyo en cada momento, vuestros consejos y por proporcionarme el entorno y los medios posibles para hacer este trabajo. Gracias por estar ahí cuando lo necesito.

A mi tutora, Carmen, sin ti este trabajo nunca se habría hecho. Gracias por tus consejos, tus conocimientos y paciencia. Gracias por resolverme cada duda que ha surgido y sobre todo por darme la oportunidad de trabajar contigo, ha sido una experiencia magnífica.

A Jimena, por guiarme en el camino a seguir, tú me inculcaste la pasión por las telecomunicaciones.

A Blanca, por estar siempre a mi lado e inspirarme a seguir creciendo.

A mis amigos y compañeros, estos años con vosotros sin duda estarán siempre conmigo. Me habéis hecho una mejor persona, con infinitos momentos que guardaremos para siempre.

A la familia de “Manchanet”, gracias por vuestro apoyo incondicional y por todo el tiempo compartido. Cada uno seguirá su camino, pero el vínculo que hemos creado se mantendrá unido eternamente.

Y a Chema, por estar todos estos días a mi lado desde la primera clase en primero hasta ahora. Cada día aprendo cosas nuevas gracias a ti. Espero que nuestra amistad perdure para siempre.

ÍNDICE DEL CONTENIDO

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

LISTA DE ACRÓNIMOS

ARO	Annualized Rate of Occurrence
CID	Confidencialidad, Integridad y Disponibilidad
CVE	Common Vulnerabilities and Exposures
CVSS	Common Vulnerability Scoring System
ENS	Esquema Nacional de Seguridad
IEC	International Electrotechnical Commission
ISO	International Organization for Standardization
JSON	JavaScript Object Notation
MAGERIT	Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información
ML	Machine Learning
OOP	Programación Orientada a Objetos
PILAR	Procedimiento Informático-Lógico para el Análisis de Riesgos
PuLP	Python Linear Programming

1. INTRODUCCIÓN Y OBJETIVOS

1.1. INTRODUCCIÓN

La ciberseguridad constituye en la actualidad un pilar fundamental para la protección de los sistemas de información, tanto en el ámbito personal como en el profesional. La creciente digitalización y la complejidad de las infraestructuras tecnológicas han incrementado significativamente la exposición a amenazas, haciendo imprescindible una gestión rigurosa y sistemática de los riesgos asociados. En España, la metodología MAGERIT se ha consolidado como referencia para el análisis y la gestión de riesgos en sistemas de información [1], proporcionando un marco estructurado para la identificación de amenazas y vulnerabilidades y también para saber los efectos de mitigación del riesgo residual asociados a cada contramedida. Para cuantificar el coste de cada una se ha recurrido a la herramienta PILAR [2].

No obstante, y a pesar de contar con metodologías consolidadas, la aplicación práctica de la gestión de riesgos presenta grandes retos [3]. Muchas organizaciones encuentran dificultades a la hora de seleccionar e implementar contramedidas de forma óptima, especialmente cuando existen limitaciones presupuestarias y una elevada interconexión entre activos. La selección manual o basada únicamente en criterios cualitativos puede conducir a soluciones poco eficientes y a una protección desigual. Asimismo, la falta de herramientas automatizadas dificulta la adaptación continua frente a la evolución de las amenazas.

Este Trabajo de Fin de Grado aborda esta problemática mediante el desarrollo de un sistema de optimización para la selección de contramedidas en la gestión de riesgos de ciberseguridad. Para llevar a cabo este trabajo se ha optado por el desarrollo de un programa en Python [4] y un catálogo de contramedidas proporcionado en JSON [5]. El sistema implementa técnicas de programación lineal, utilizando la librería PuLP [6], para modelar matemáticamente la relación entre activos, amenazas, vulnerabilidades y contramedidas, resolviendo el problema de asignación óptima bajo una restricción presupuestaria definida por el usuario. La representación de la red mediante la librería NetworkX [7] permite modelar los activos como nodos en un grafo, facilitando tanto el análisis estructural como la visualización de las contramedidas seleccionadas y su impacto sobre los distintos elementos de la red.

Los resultados obtenidos a través de simulaciones en distintos escenarios de red demuestran que el sistema es capaz de priorizar y seleccionar de manera eficiente las contramedidas más adecuadas en función de la topología de la red y las restricciones económicas. El análisis de estos escenarios permite observar una reducción significativa del riesgo residual, así como una mejora en la asignación de recursos y en la toma de decisiones estratégicas. Además, el sistema ha demostrado ser escalable y flexible, permitiendo su adaptación a distintas configuraciones de red y a diferentes niveles de riesgo aceptable.

En conjunto, este proyecto contribuye a la mejora de la gestión de riesgos en ciberseguridad, proporcionando una base objetiva, cuantificable y automatizada para la selección eficiente de contramedidas, reforzando la protección de los activos digitales en entornos reales.

1.2. OBJETIVOS

El objetivo principal de este trabajo es desarrollar un sistema capaz de recomendar una serie de contramedidas, seleccionadas en función de su coste estimado y un presupuesto previamente definido. Estas contramedidas estarán orientadas a mitigar un conjunto de amenazas y vulnerabilidades que afectan a determinados activos, con el fin de reducir al máximo el riesgo residual sin superar el límite presupuestario establecido.

A partir de la identificación de la problemática inicial, se plantean una serie de objetivos específicos que deberán alcanzarse progresivamente hasta completar el desarrollo del Trabajo de Fin de Grado:

1) **Estudiar y seleccionar la metodología de gestión de riesgos más adecuada:** El primer objetivo es analizar las principales metodologías de gestión de riesgos en ciberseguridad, con especial atención a MAGERIT, para fundamentar el desarrollo del sistema en un marco metodológico robusto y reconocido.

2) **Definir y estructurar los datos de entrada del sistema:** Se identifican y estructuran los datos necesarios para el funcionamiento del sistema, incluyendo la definición de activos, amenazas, vulnerabilidades, así como la utilización del formato JSON para el catálogo de contramedidas.

3) **Modelar el problema de selección de contramedidas como un problema de optimización:** Se desarrolla un modelo matemático que relacione activos, amenazas, vulnerabilidades, contramedidas, costes y restricciones presupuestarias, permitiendo la formulación del problema como un programa lineal.

4) **Implementar el sistema de optimización y visualización:** Se implementa el sistema en Python, empleando la librería PuLP para resolver el modelo de optimización y NetworkX para la representación gráfica de la red, facilitando el análisis visual de las soluciones propuestas.

5) **Validar el sistema mediante simulaciones y análisis de resultados:** Una vez desarrollado el sistema, se realizan simulaciones en distintos escenarios de red y presupuesto para analizar su comportamiento, evaluar su rendimiento y comprobar su capacidad para priorizar y seleccionar contramedidas que minimicen el riesgo residual dentro de las restricciones establecidas.

1.3. ESTRUCTURA DEL TRABAJO

El documento se organiza en cinco capítulos diferenciados:

Capítulo 1: Introducción y objetivos
Presenta el contexto de la ciberseguridad, la problemática identificada en la selección óptima de contramedidas y la solución propuesta: un sistema de optimización. Define los objetivos del trabajo y describe la estructura del documento.

Capítulo 2: Marco teórico
Aborda los fundamentos teóricos de la gestión de riesgos, profundizando en las metodologías MAGERIT y PILAR. Incluye conceptos clave de optimización matemática y describe las herramientas tecnológicas empleadas: Python, la librería PuLP para programación lineal y NetworkX para modelado de redes.

Capítulo 3: Desarrollo
Detalla la arquitectura del sistema, incluyendo el catálogo de contramedidas en formato JSON, el modelado matemático del riesgo residual, el algoritmo de optimización mediante programación lineal y la representación gráfica de la red como grafo.

Capítulo 4: Validación
Analiza y comenta los resultados de distintas simulaciones sobre la red, contrastando el rendimiento del sistema frente a restricciones presupuestarias y topologías variables. Incluye la validación del sistema mediante casos de uso reales evaluando sus resultados.

Capítulo 5: Conclusiones y líneas futuras
Sintetiza los logros principales del trabajo, evalúa el cumplimiento de los objetivos y propone líneas futuras de investigación.

2. MARCO TEÓRICO

El capítulo actual introduce los conceptos clave que sustentan el desarrollo del sistema. Primero, se explica la gestión de riesgos en ciberseguridad [8] y las metodologías empleadas, como MAGERIT [1]. Seguidamente, se aborda el papel de la optimización en la selección de contramedidas y, finalmente, se describen las herramientas utilizadas en el proyecto.

2.1. GESTIÓN DE RIESGOS

La gestión de riesgos [9] es un proceso sistemático orientado a identificar, evaluar y controlar amenazas que puedan afectar el logro de los objetivos de una organización. Este enfoque se aplica en múltiples ámbitos y tiene como finalidad reducir la incertidumbre y mitigar los impactos negativos derivados de eventos adversos. La gestión de riesgos permite priorizar acciones y recursos mediante el análisis de la probabilidad y el impacto de distintos escenarios, favoreciendo la toma de decisiones informadas y sostenibles en el tiempo.

En el contexto digital actual, la gestión de riesgos en ciberseguridad se ha consolidado como una función estratégica imprescindible. La creciente exposición a ciberamenazas, junto con la alta dependencia de activos tecnológicos, ha incrementado la probabilidad e impacto de incidentes de seguridad, lo que obliga a adoptar enfoques estructurados y proactivos para su mitigación [10] [11].

Esta disciplina, aplicada a la protección de la información y los sistemas digitales, integra procesos para identificar vulnerabilidades, evaluar riesgos y aplicar controles que reduzcan su probabilidad o severidad. A diferencia de enfoques puramente reactivos, la gestión de riesgos permite anticiparse a amenazas, optimizando el uso de recursos y fortaleciendo la resiliencia organizacional [12].

Existen marcos metodológicos ampliamente reconocidos para la implementación de la gestión de riesgos en entornos digitales. Entre ellos destaca la Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información (MAGERIT), desarrollado por la Administración General del Estado en España, que proporciona una estructura formal y detallada para identificar, valorar y tratar riesgos asociados a los sistemas de información. MAGERIT está orientado a facilitar la incorporación de la gestión de riesgos en los procesos organizativos, promoviendo una cultura de seguridad continua y sostenible. Asimismo, metodologías internacionales como el NIST Cybersecurity Framework [13] o la norma ISO/IEC 27001 [14] ofrecen enfoques estructurados que pueden complementarse con MAGERIT, especialmente en contextos que requieren alineación con estándares globales [1].

Los riesgos en ciberseguridad se agrupan generalmente en tres dimensiones interdependientes [15]: técnica, humana y organizativa. En la dimensión técnica se encuentran las amenazas asociadas a vulnerabilidades en los sistemas, configuraciones inadecuadas o falta de mecanismos de protección. En el plano humano, los errores involuntarios y las técnicas de ingeniería social constituyen vectores frecuentes de ataque. Por su parte, los riesgos

organizativos derivan de la falta de políticas adecuadas, recursos limitados o estructuras de gobernanza deficientes. La interacción de estas dimensiones evidencia la necesidad de un enfoque integral que combine tecnología, formación y gestión para minimizar el riesgo residual.

Para priorizar las acciones y recursos en la gestión de riesgos, es habitual emplear una matriz de probabilidad e impacto como en la figura 2.1. Esta herramienta permite clasificar los riesgos en función de la probabilidad de que ocurran y el impacto que tendrían sobre la organización, facilitando la toma de decisiones informadas y la asignación eficiente de recursos.

Risk Management Matrix		Impact				
		Negligible	Marginal	Moderate	Critical	Catastrophic
Probability	Almost Certain	Low Risk	Moderate Risk	High Risk	Extreme Risk	Extreme Risk
	Likely	Minimum Risk	Low Risk	Moderate Risk	High Risk	Extreme Risk
	Possible	Minimum Risk	Low Risk	Moderate Risk	High Risk	High Risk
	Unlikely	Minimum Risk	Low Risk	Low Risk	Moderate Risk	High Risk
	Rare	Minimum Risk	Minimum Risk	Low Risk	Moderate Risk	High Risk

Figura 2.1 - Matriz de gestión de riesgos [16]

Ahora, se procederá a definir los elementos fundamentales que conforman el proceso de gestión de riesgos en ciberseguridad [17]:

Activo: cualquier recurso de la organización necesario para el desarrollo de sus actividades, cuya pérdida o deterioro supone un perjuicio o coste. Los activos pueden ser información, sistemas, personas, infraestructuras, etc. Su protección es el objetivo principal de la gestión de riesgos.

Amenaza: circunstancia o evento desfavorable que puede ocurrir y producir consecuencias negativas sobre los activos, provocando su indisponibilidad, mal funcionamiento o pérdida de valor.

Vulnerabilidad: debilidad presente en los activos que facilita la materialización de una amenaza. Puede estar relacionada con hardware, software, redes, personal o la propia organización.

Impacto: consecuencia derivada de la materialización de una amenaza sobre un activo, aprovechando una vulnerabilidad. El impacto se mide habitualmente en términos de degradación del valor del activo, pudiendo afectar también a la reputación, el rendimiento o la continuidad del negocio.

Probabilidad: posibilidad de que un suceso o amenaza ocurra y cause un impacto negativo. Se puede estimar a partir de datos históricos, experiencia previa o juicio de expertos.

Contramedida: medida o conjunto de acciones orientadas a reducir la probabilidad o el impacto de una amenaza sobre un activo. Incluye controles técnicos, organizativos o procedimentales.

Riesgo residual: nivel de riesgo que permanece después de aplicar las contramedidas. Representa la exposición final de la organización y debe estar por debajo del umbral de tolerancia definido.

En el siguiente apartado se describirá más a fondo la metodología MAGERIT.

2.1.1. MAGERIT

Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información (MAGERIT) [1]es un marco metodológico desarrollado en 1997 por el Consejo Superior de Administración Electrónica (CSAE) del Ministerio de Hacienda y Administraciones Públicas de España, con el objetivo de identificar, evaluar y mitigar riesgos asociados a los sistemas de información en entornos públicos y privados. Surgió como respuesta a la creciente digitalización de la Administración Pública, buscando garantizar la protección de activos tecnológicos mediante un enfoque estructurado que integra principios de confidencialidad, integridad y disponibilidad (CID) [18]. La metodología, actualizada en su versión 3.0 en 2012, se articula en tres pilares documentales: el Libro I (proceso metodológico), el Libro II (catálogos de activos, amenazas y salvaguardas) y el Libro III (técnicas de análisis cuantitativas y cualitativas). Alineada con estándares como ISO 31000 y el Esquema Nacional de Seguridad (ENS), MAGERIT utiliza herramientas como PILAR para automatizar la gestión de riesgos, combinando modelos matemáticos con prácticas organizativas adaptativas [19].

MAGERIT tiene objetivos claramente definidos y principios metodológicos que garantizan su aplicación en todo tipo de contextos. Su diseño responde a la necesidad de estandarizar la gestión de riesgos tecnológicos.

Sus objetivos están divididos en:

Directos

- ◊ **Concienciación institucional:** Sensibilizar a las organizaciones sobre la existencia de ciertos riesgos que aparecen con el uso de los sistemas de información y la necesidad inherente de gestionarlos tanto activa como proactivamente.
- ◊ **Provisión de un método sistemático:** Ofrecer un marco estructurado capaz de detectar, analizar y evaluar riesgos asociados a activos.
- ◊ **Planificación de contramedidas:** Facilitar su selección e implementación, optimizando la relación coste-beneficio.

Indirectos

- ◊ **Preparación para auditorías y certificaciones:** Establecer una base de documentos que permita a las organizaciones someterse a procesos como los exigidos por el Esquema Nacional de Seguridad (ENS) [20] o la norma ISO/IEC 27001 [21].
- ◊ **Articulación con marcos normativos:** Garantizar la compatibilidad con estándares internacionales.

MAGERIT se sustenta en 6 principios que orientan su aplicación práctica:

1) **Enfoque sistemático:** Reconoce la interdependencia entre activos, donde la degradación de uno puede afectar a otros interconectados. Este principio exige analizar tanto el impacto directo como el impacto repercutido, calculando el valor acumulado de los activos en red.

2) **Proporcionalidad:** Las medidas de seguridad deben ser proporcionales al valor del activo y al riesgo estimado. MAGERIT evita soluciones genéricas, promoviendo la selección de salvaguardas basadas en análisis específicos de impacto y probabilidad.

3) **Ciclo de mejora continua:** La metodología exige revisiones periódicas para adaptarse a cambios tecnológicos, nuevas amenazas o modificaciones en los procesos organizativos. Este principio se materializa en la fase de cálculo del riesgo residual, donde se verifica la eficacia de las contramedidas implementadas.

4) **Triada CID (Confidencialidad, Integridad y Disponibilidad):** es el núcleo de valoración de activos:

   Confidencialidad: Garantizar que la información solo sea accesible a entidades autorizadas.

   Integridad: Mantener la exactitud y completitud de los datos durante su ciclo de vida.

Disponibilidad: Asegurar el acceso a los sistemas y servicios críticos cuando se requieran.

5) **Adaptabilidad metodológica:** MAGERIT admite modelos híbridos, combinando técnicas cualitativas (matrices de impacto-probabilidad) para riesgos estratégicos con enfoques cuantitativos (cálculo monetario de pérdidas anuales esperadas) para riesgos operativos.

6) **Documentación estandarizada:** consiste en la generación de informes que incluyan mapa de riesgos, estado de riesgo actual y plan de seguridad.

MAGERIT se organiza con un proceso cíclico y estructurado compuesto por cinco fases interconectadas que garantizan una gestión integral y adaptable de los riesgos tecnológicos dichas fases, descritas en el Libro I citado anteriormente, se retroalimentan iterativamente para mantener la eficacia del modelo en posibles cambios en el entorno tecnológico o en la aparición de amenazas. Estas son las fases:

En primer lugar, se lleva a cabo la **identificación de activos**, donde se detectan y catalogan los activos relevantes para la organización, se establecen las relaciones entre ellos y se realiza una valoración según su importancia y criticidad para el negocio. Esta etapa es fundamental, ya que la protección de los activos constituye el objetivo principal de la gestión de riesgos.

A continuación, se procede a la **identificación de amenazas**, que consiste en reconocer y analizar las amenazas que pueden afectar a los activos previamente identificados. Cada amenaza se caracteriza y valora en función de una estimación de su probabilidad de ocurrencia y el impacto potencial que podría tener sobre la organización.

La siguiente fase es la **identificación de las salvaguardas**, donde se examinan y valoran las medidas de protección existentes o potenciales. El propósito de esta etapa es determinar qué salvaguardas son pertinentes para limitar el daño causado por las amenazas, así como monitorizar los riesgos de forma continua para mantener el sistema protegido.

Posteriormente, se realiza la **estimación del estado de riesgo**, cuyo objetivo es calcular y anticipar tanto los escenarios posibles como los más probables en función de la información recopilada sobre activos, amenazas y salvaguardas. Esta estimación permite obtener una visión clara del nivel de riesgo al que está expuesta la organización.

Por último, se aborda la **evaluación del impacto y del riesgo residual,** así como su tratamiento. En esta fase, se analiza el riesgo que permanece tras la aplicación de las salvaguardas (riesgo residual) y, en función de su nivel, se decide el tratamiento más adecuado para reducirlo, ya sea mediante la implementación de nuevas medidas, la transferencia del riesgo, su aceptación o su eliminación.

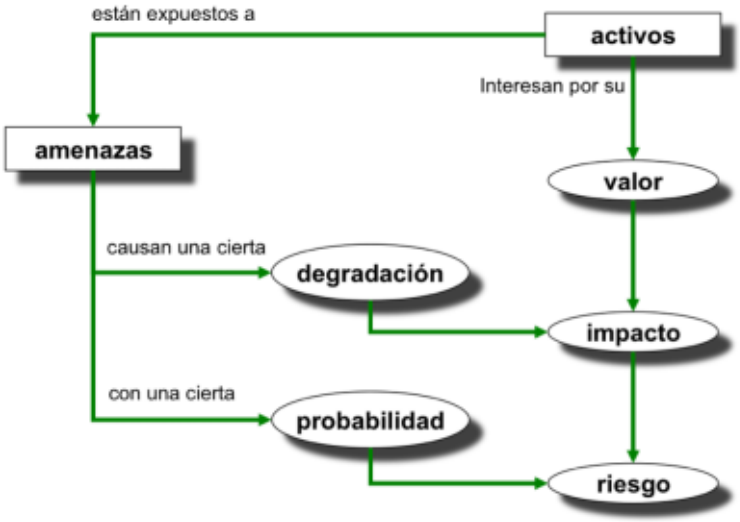

Figura 2.2 - Funcionamiento de MAGERIT [22]

La figura 2.2 sintetiza el flujo fundamental del análisis de riesgos según MAGERIT. En ella se observa cómo los activos, que representan los elementos de valor para la organización, están expuestos a diferentes amenazas. El interés por proteger estos activos radica en su valor, ya que cualquier amenaza que se materialice puede causar una degradación de sus propiedades esenciales, como la confidencialidad, la integridad o la disponibilidad. Esta degradación genera un impacto sobre la organización, cuya magnitud depende directamente del valor del activo afectado. Además, cada amenaza lleva asociada una probabilidad de ocurrencia, lo que permite estimar el riesgo como la combinación del impacto potencial y la probabilidad de que se produzca el evento adverso. Este esquema refleja la lógica subyacente de la gestión de riesgos en MAGERIT, donde la identificación y valoración de activos, amenazas, impacto y probabilidad constituyen los pilares para calcular y tratar el riesgo de forma sistemática y fundamentada.

2.1.2. PILAR

PILAR [23] emerge como un sistema pionero en la operacionalización de la metodología MAGERIT, desarrollado por el Centro Criptológico Nacional [24] (CCN) para transformar los principios teóricos de gestión de riesgos en procesos automatizados y estandarizados [25]. Esta herramienta de código abierto fusiona marcos normativos como el Esquema Nacional de Seguridad [20] (ENS) y la ISO/IEC 27002 [26] con algoritmos cuantitativos, permitiendo a organizaciones públicas y privadas mapear sus activos críticos, evaluar amenazas contemporáneas y diseñar estrategias de mitigación basadas en análisis costo-beneficio. Su arquitectura modular integra catálogos dinámicos de amenazas, modelos matemáticos de cálculo de riesgo residual y generadores de documentación regulatoria,

estableciendo un puente entre la teoría de la seguridad informática y las necesidades operativas reales.

La sinergia entre PILAR y MAGERIT se materializa a través de un flujo de trabajo estructurado. Durante la identificación de activos, la herramienta facilita el inventario jerárquico de recursos tecnológicos y organizacionales, asignando valores económicos mediante modelos de dependencia que ponderan la criticidad relativa de cada componente. Por ejemplo, en sistemas de pago electrónico, PILAR calcula automáticamente la contribución porcentual de servidores específicos al valor total del servicio, eliminando subjetividades en la valoración inicial [27].

En la fase de análisis de amenazas, los catálogos preconfigurados de PILAR, actualizados con datos de ciberinteligencia del CCN, correlacionan vulnerabilidades técnicas con vectores de ataque probables. La plataforma implementa árboles de amenazas interactivos que visualizan rutas de explotación potenciales, como la propagación de vulnerabilidades en configuraciones de *firewall* mal implementadas, calculando simultáneamente la frecuencia anual esperada (ARO) mediante modelos probabilísticos basados en historiales de incidentes [27].

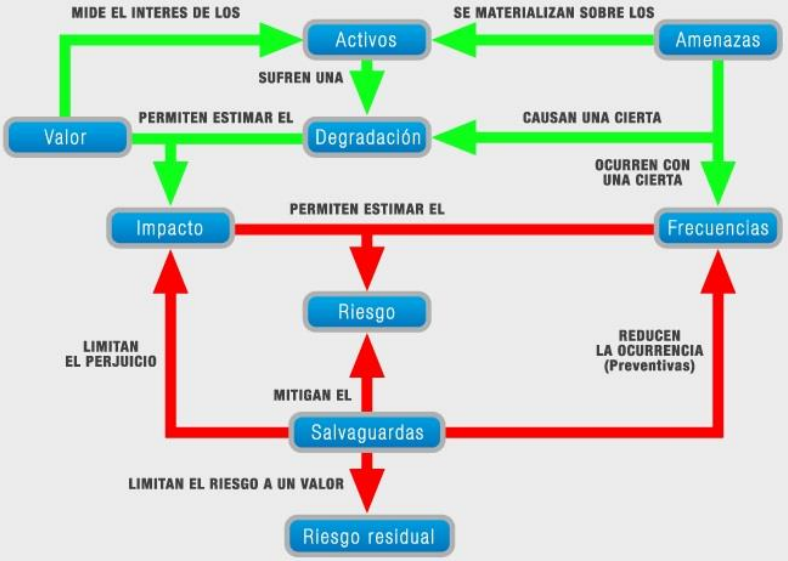

Figura 2.3 - Implementación de PILAR y MAGERIT [28]

En el contexto de este Trabajo de Fin de Grado, el aspecto de mayor relevancia es el cálculo del Riesgo Residual mediante la herramienta PILAR, que se realiza a partir de la siguiente fórmula:

$$\boldsymbol{Riesgo\ Residual} = Probabilidad \times Impacto$$

En el sistema desarrollado, los valores de probabilidad e impacto, así como los costes asociados a cada contramedida, se han extraído y adaptado a partir de los catálogos y

recomendaciones de PILAR, asegurando la coherencia con la metodología MAGERIT y la comparabilidad de resultados.

El cálculo del riesgo residual de esta forma permite cuantificar de forma objetiva el nivel de exposición que permanece en un sistema tras la aplicación de las contramedidas seleccionadas. Esta aproximación, utilizada en la herramienta PILAR, aporta varias ventajas clave: por un lado, facilita la comparación directa entre distintos escenarios de seguridad y la priorización de recursos, ya que traduce la complejidad del entorno tecnológico a valores numéricos fácilmente interpretables. Por otra parte, permite identificar de manera precisa qué amenazas continúan representando un peligro significativo y si el nivel de riesgo residual se encuentra dentro de los umbrales aceptables definidos por la organización. Además, PILAR automatiza la generación de informes y documentación exigida por los principales estándares de seguridad, lo que facilita el cumplimiento normativo, la trazabilidad y la transparencia de las decisiones tomadas en la gestión de riesgos.

2.2. OPTIMIZACIÓN

La optimización matemática constituye un campo fundamental que busca encontrar la mejor solución posible a un problema específico dentro de un conjunto de alternativas factibles, sujetas a restricciones determinadas. En el contexto de la gestión de riesgos, la optimización permite tomar decisiones eficientes cuando los recursos son limitados y es necesario maximizar beneficios o minimizar costes y riesgos. Este enfoque resulta especialmente relevante para la selección de contramedidas de seguridad, donde es imprescindible encontrar el equilibrio óptimo entre la reducción del riesgo residual y los recursos económicos disponibles [29].

Todos los problemas de optimización necesitan tres elementos fundamentales para ser formulados:

- **Función objetivo:** Es la expresión matemática que define el objetivo a alcanzar, como puede ser minimizar el riesgo o maximizar la eficacia de las medidas implementadas.

- **Variables de decisión:** Representan las diferentes alternativas o acciones que se pueden tomar para influir en el resultado final.

- **Restricciones:** Son las limitaciones, ya sean de presupuesto, recursos o normativas, que acotan el conjunto de soluciones posibles.

Ahora, vamos a adaptar estos elementos a un problema de optimización en el ámbito de la ciberseguridad. Se tiene un catálogo de contramedidas basado en MAGERIT y un sistema con una serie de activos interconectados, el objetivo será proteger de la forma más eficiente el sistema disminuyendo al máximo posible el riesgo residual del mismo con una limitación clara, el presupuesto. Cada contramedida tendrá un coste asociado y no se puede superar este

límite mencionado antes, por lo que es necesario seleccionar de forma óptima las contramedidas.

En este caso la función objetivo es minimizar el riesgo residual del sistema completo. Las variables de decisión serán las contramedidas que en función de su elección influirán en mayor o menor medida sobre el resultado final. Por último, la restricción es el presupuesto ya que limitan el número de contramedidas seleccionadas.

La optimización en la gestión de riesgos de ciberseguridad permite estructurar y resolver de manera sistemática el desafío de asignar recursos limitados para maximizar la protección de los activos críticos. Al formular el problema mediante una función objetivo clara, variables de decisión precisas y restricciones concretas, es posible identificar el conjunto óptimo de contramedidas que minimizan el riesgo residual sin exceder el presupuesto disponible. De esta manera es posible abordar la toma de decisiones sobre un tema tan importante como proteger los activos de una manera más eficiente. Además, se tendrá una adaptación continua a un entorno tan dinámico como es la ciberseguridad. Otro factor importante es que así las inversiones en este ámbito aportarán el mayor valor posible y los sistemas de todas las organizaciones serán más seguros.

2.3. HERRAMIENTAS

En este apartado se describen las herramientas, entorno y lenguaje de programación usado, así como las librerías necesarias para realizar este proyecto.

2.3.1. PYTHON

Para el desarrollo del sistema se ha elegido lenguaje de programación Python.

Python es uno de los lenguajes de programación más usados del mundo en todo tipo de tecnologías, tales como aplicaciones web, desarrollo de software, ciencia de datos o Machine Learning (ML). Es un lenguaje de alto nivel y orientado a objetos (OOP), esto quiere decir que el código se organiza en términos de objetos que son unidades que contienen datos (atributos) y acciones (métodos) que actúan sobre estos datos.

Se ha elegido Python por los numerosos beneficios que ofrece como [30]:

- Permite que los desarrolladores de código sean más productivos, ya que se puede escribir un programa con un menor número de líneas de código que con cualquier otro lenguaje.

- La comunidad activa de Python es enorme, millones de desarrolladores en el mundo prestan su apoyo, y se puede obtener soporte de una manera muy rápida.

- ◊ Python es válido para cualquier tipo de sistema operativo en la actualidad, por lo que se puede ejecutar en cualquier ordenador.

- ◊ Es capaz de trabajar e integrarse con otros lenguajes muy utilizados como C++, C o Java [31].

Aparte de estas ventajas, en Python existen las bibliotecas que son una colección de códigos usados con frecuencia que se pueden incluir en los programas para evitar tener que empezar a programar desde cero. De forma predeterminada Python incluye la biblioteca estándar con una gran cantidad de funciones que se pueden utilizar para muy diversas tareas.

Hoy en día hay más de 137.000 bibliotecas de Python disponibles. En este Trabajo de Fin de Grado se han utilizado las siguientes:

Matplotlib: su finalidad es la visualización de datos. Se usa para dibujar y mostrar el grafo generado por NetworkX, cuyo funcionamiento se detallará más adelante. Además, también es usada para añadir anotaciones y detalles sobre el grafo obtenido.

JSON: librería estándar que sirve para en este caso, leer datos en formato JSON [32].

sys: biblioteca que proporciona acceso a variables y funciones del sistema.

Junto a ellas se han utilizado dos más que merecen su propio apartado.

2.3.2. PULP

PuLP es una librería de Python utilizada para modelar y resolver problemas de optimización mediante la programación lineal [33]. Es de código abierto, por lo que es accesible para cualquier usuario, esto facilita su uso dentro de cualquier tipo de proyecto que necesite realizar cálculos de esta índole. Gracias a esto es una herramienta usada en investigación operativa, logística, finanzas y muchas más áreas.

Su funcionamiento se basa en la definición de un problema de optimización, que puede ser tanto de maximización como de minimización. Es capaz de plantear la función objetivo, crear variables de decisión y establecer restricciones, formuladas como expresiones matemáticas. Todo ello hace que se llegue a una solución eficiente y optimizada porque trata el problema con los elementos clave de la optimización.

La comunidad activa y la compatibilidad con diferentes librerías hace que PuLP sea una opción robusta y flexible para abordar diversos problemas complejos de este tipo.

La instalación y su documentación oficial se encuentran en su repositorio oficial de GitHub [34].

2.3.3. NETWORKX

NetworkX [35] es una biblioteca de Python cuyo objetivo es la creación, gestión y análisis de grafos y redes complejas. Sus clases principales permiten modelar tanto grafos dirigidos como no dirigidos, con soporte para cualquier forma de arista y atributos personalizados.

Combina flexibilidad y rendimiento resultando ser una herramienta muy importante para aplicaciones en ciencia de datos, investigación de operaciones y análisis de redes. Su diseño orientado a objetos facilita la extensión y adaptación a problemas complejos.

En este caso se ha usado para representar una red con sus activos y dirigir las vulnerabilidades, amenazas y contramedidas a cada nodo asignado. Permitiendo una comprensión del problema de optimización viendo de forma gráfica las contramedidas seleccionadas. Gracias a esta herramienta se puede mostrar una visión general del sistema y como queda tras haberse completado todos los cálculos.

Alojado en GitHub, permite su acceso a todos los usuarios que necesiten utilizar esta herramienta [36].

3. DESARROLLO

El tercer capítulo de este proyecto describe en detalle la implementación del sistema de optimización diseñado para la selección de contramedidas en la gestión de riesgos de ciberseguridad. Se expone el proceso completo seguido durante el desarrollo del Trabajo de Fin de Grado, desde la formulación matemática del problema y la estructuración de los datos en formato JSON, hasta la integración de metodologías como MAGERIT y la herramienta PILAR para la valoración de las contramedidas a seleccionar.

Se describe cómo los principios de MAGERIT se transforman en un modelo de optimización lineal que permite seleccionar las contramedidas propuestas de una forma automática y eficaz bajo una restricción de presupuesto. Tras implementar la solución del problema se presenta de una manera gráfica para poder analizar resultados y sacar conclusiones.

La arquitectura propuesta prioriza la escalabilidad y modularidad [37], permitiendo que esta se adapte a todo tipo de topología de red y diferentes escenarios. Cada componente del sistema está diseñado para respetar la coherencia de las metodologías empleadas, a la vez que aporta automatización y optimización para superar las barreras que tenían los criterios cualitativos en la elección de contramedidas.

3.1. ARQUITECTURA DEL SISTEMA

La arquitectura del sistema desarrollado para la optimización de contramedidas en ciberseguridad es un modelo programado donde la información fluye desde los datos de entrada (activos, amenazas, vulnerabilidades y contramedidas) hasta la visualización y generación de informes. El sistema está compuesto fundamentalmente por dos ficheros: uno en formato JSON que define el catálogo de contramedidas adaptadas para este caso de uso y otro en Python que implementa el problema de optimización, cálculo de riesgos residuales y visualización gráfica.

Este proyecto se centra sobre todo en la optimización y el cálculo de las contramedidas aplicadas y su posterior representación, por lo que los datos de entrada pueden ser adaptados para cualquier topología de red.

La figura 3.1 mostrada a continuación presenta una versión simplificada de la arquitectura global del sistema.

En ella se encuentran tres flujos de entrada que son el fichero JSON con el catálogo de contramedidas, la definición de activos, amenazas y vulnerabilidades que se describirán más adelante y el presupuesto límite que no es un valor automático, sino que lo introduce el usuario mediante un input tras la ejecución del fichero *sistema_optimizacion.py*.

El sistema genera tres flujos de salida principales tras el procesamiento y optimización. El primer flujo produce una representación visual mediante NetworkX que muestra un grafo

dirigido incluyendo métricas de presupuesto y riesgo superpuestas en la visualización. El segundo flujo genera un archivo de resultados estructurado (resultados.txt) que contiene el detalle completo de contramedidas aplicadas, riesgo residual por activo y métricas comparativas entre el estado inicial y final del sistema. El tercer flujo calcula y presenta el riesgo residual final optimizado como un valor numérico resultante del modelo de programación lineal, lo que permite una evaluación cuantitativa de la efectividad de las contramedidas seleccionadas bajo la restricción presupuestaria definida por el usuario.

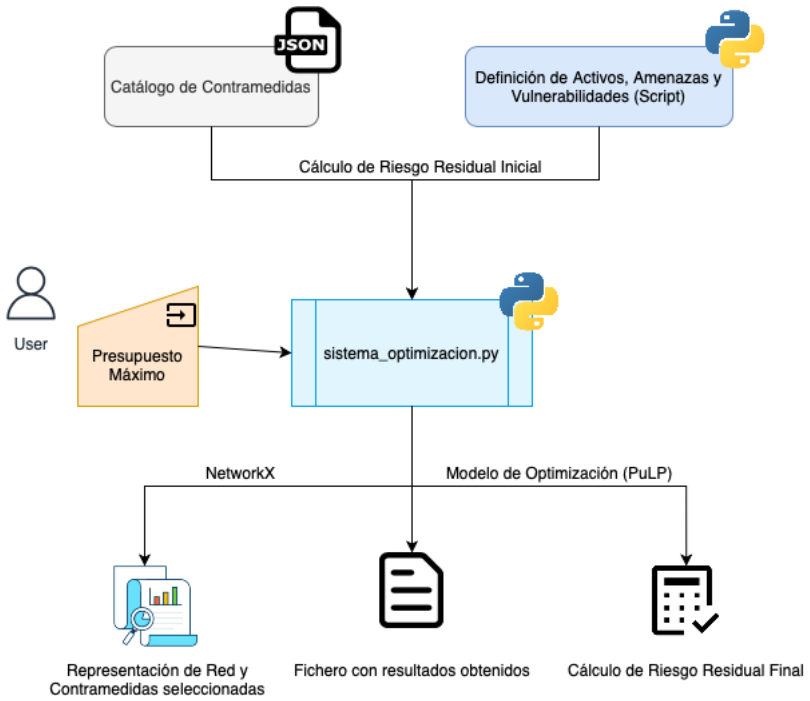

Figura 3.1 - Arquitectura del sistema

A continuación, se presentarán de una forma desarrollada todos los flujos tanto de entrada como de salida de la arquitectura para que sea comprendida en su totalidad.

3.2. DATOS DE ENTRADA

Se describirá la estructura y características de los flujos de entrada que alimentan el sistema de optimización desarrollado en este Trabajo de Fin de Grado. La arquitectura de datos implementada constituye el fundamento sobre el cual opera el algoritmo de selección de contramedidas, estableciendo las relaciones entre activos, amenazas, vulnerabilidades y medidas de protección disponibles. Si bien la configuración específica presentada responde a un problema de optimización llevado a la realidad en concreto, es importante destacar que el sistema ha sido diseñado con criterios de escalabilidad y adaptabilidad, permitiendo su aplicación a diferentes topologías de red, catálogos de contramedidas y escenarios organizacionales.

La modularidad de la estructura de datos garantiza que cualquier organización pueda adaptar el modelo a sus necesidades específicas, siempre que se respete la arquitectura de información definida. Esta flexibilidad resulta fundamental para la aplicabilidad práctica del sistema, ya que permite su implementación en entornos heterogéneos sin requerir modificaciones en el cuerpo del código creado. Los datos de entrada deben mantener coherencia estructural con los formatos establecidos para asegurar la correcta ejecución del código de optimización y la generación de resultados válidos.

3.2.1. CATÁLOGO DE CONTRAMEDIDAS

Las contramedidas para los casos de uso de este proyecto han sido escogidas a raíz de hacer un estudio previo de las prácticas más usadas por las empresas [38]. Su desarrollo forma el núcleo operativo del sistema ya que aquí residen las medidas de seguridad disponibles para gestionar las amenazas y prevenir o reaccionar ellos mitigando el riesgo residual del sistema y de cada activo. Su diseño sigue los principios de la metodología MAGERIT.

Para este Trabajo de Fin de Grado se ha hecho una lista de 26 contramedidas presentadas en un formato JavaScript Object Notation (JSON), ya que este formato es ligero, legible por humanos y ampliamente compatible con Python. Además, facilita la estructuración jerárquica de datos, lo que permite representar de forma clara las propiedades de cada contramedida. Python ofrece bibliotecas nativas como json que facilitan su lectura, escritura y manipulación [39]. Esto permite una integración directa con el modelo de optimización desarrollado, simplificando el flujo de entrada de datos.

Contramedida	Descripción	Amenazas Mitigadas (MAGERIT v3)
M0-DoNothing	Opción nula que representa no aplicar ninguna contramedida	Disponible para todas las amenazas (no mitiga ninguna, es seleccionada si el coste de cualquier contramedida individualmente es mayor que la restricción presupuestaria
M1-Firewall	Filtrado de tráfico no autorizado	Deficiencias organizativas, Vulnerabilidades de software, Errores de encaminamiento
M2-Antivirus	Detección de malware en *endpoints*	Difusión de software dañino, Errores de configuración
M3-Backup	Copias de seguridad automáticas	Destrucción de información, Errores de mantenimiento
M4-IDS	Detección de intrusiones en tiempo real	Acceso no autorizado, Análisis de tráfico
M5-Encryption	Cifrado de datos en tránsito/reposo	Interceptación de información, Fugas de datos
M6-AccessControl	Gestión granular de permisos	Abuso de privilegios, Suplantación de identidad
M7-VPN	Tunelización segura para acceso remoto	Interceptación de información, Errores de encaminamiento
M8-Training	Formación en concienciación	Ingeniería social, Errores humanos
M9-Logging	Registros de auditoría centralizados	Manipulación de logs, Repudio de acciones
M10-Patching	Actualización automatizada de parches	Vulnerabilidades de software, Errores de mantenimiento
M11-Monitoring	Supervisión continua de	Denegación de servicio, Caídas por agotamiento de

Contramedida	Descripción	Amenazas Mitigadas (MAGERIT v3)
	la red	recursos
M12-Redundancy	Réplicas de sistemas críticos	Pérdida de equipos, Destrucción física
M13-AccessLogs	Trazabilidad de accesos	Accesos no autorizados, Análisis de tráfico
M14-DataMasking	Enmascaramiento de datos sensibles	Fugas de información, Divulgación accidental
M15-PasswordPolicy	Políticas robustas de contraseñas	Ataques por fuerza bruta, Suplantación
M16-NetworkSegmentation	Aislamiento de segmentos de red	Propagación lateral de amenazas, Errores de encaminamiento
M17-DLP	Prevención de fuga de datos	Fugas de información, Divulgación deliberada
M18-EndpointProtection	Protección avanzada en dispositivos	Malware persistente, Explotación de vulnerabilidades
M19-CloudBackup	Respaldos en la nube con redundancia	Destrucción de información, Desastres físicos
M20-PhysicalSecurity	Controles de acceso físico	Robo de equipos, Acceso físico no autorizado
M21-EmailFiltering	Filtrado anti-phishing/SPAM	Ingeniería social, Difusión de software dañino
M22-UserAwareness	Programas de concienciación continua	Errores humanos, Ingeniería social
M23-LoadBalancing	Distribución equilibrada de cargas	Denegación de servicio, Caídas por sobrecarga
M24-FileIntegrity	Verificación de integridad de archivos	Manipulación de programas, Alteración deliberada de información
M25-ApplicationFirewall	Protección específica para aplicaciones web	Inyección de código, Ataques de capa de aplicación

Tabla 3.1 - Contramedidas, descripción y amenazas mitigadas según MAGERIT

En la tabla 3.1 se han descrito las 26 contramedidas que constituyen el archivo *contramedidas.json.* La descripción de cada una permite relacionarla con por lo menos dos amenazas descritas en MAGERIT.

La estructura de este fichero es la de un objeto JSON que contiene una clave por cada contramedida implementada. Estas claves, a su vez, implementan una serie de pares clave-valor que se definirán seguidamente:

Tipo: clasifica la contramedida según cómo actúa esta sobre las amenazas. Pueden ser preventivas (actúan antes de que las amenazas se materialicen) o reactivas (buscan reducir el impacto que tiene una amenaza cuando ya ha actuado sobre el activo).

Coste: valor entero que representa no solo factores puramente económicos, sino que también circunstancias de gestión, costes de uso, de mantenimiento y operación. Este coste ha sido calculado con PILAR y normalizado para manejar valores comparables entre distintas contramedidas, facilitando así su evaluación y priorización en el proceso de análisis de riesgos.

La tabla 3.2 muestra los costes manejados para la implementación de este proyecto.

Coste	Descripción
10-20: Coste Bajo	Medidas sencillas, poco intrusivas, fáciles de implementar.
21-40: Coste Medio	Requieren inversión económica y/o esfuerzo humano moderado.
40+: Coste Alto	Herramientas complejas, caras, que requieren integración profunda.

Tabla 3.2 - Costes de las contramedidas

Reducción de probabilidad y reducción de impacto: diccionarios que cuantifican como la contramedida modifica el riesgo de cada amenaza, reduciendo así el riesgo residual de cada activo y por lo tanto el total.

Aplicada: es un valor booleano que sirve para saber si la contramedida ha sido usada durante el proceso de selección. Monitoriza cada una para la posterior generación de ficheros con los resultados y para facilitar el gráfico en el que se detalla la conclusión sacada de cada caso de uso.

Cabe destacar el papel particular de la contramedida **M0-DoNothing**, que representa la opción de no aplicar nada ante determinadas amenazas. Esta contramedida, aunque pueda parecer trivial, cumple una función clave en el modelo: establecer una base de comparación que permite evaluar de forma cuantitativa el beneficio real de aplicar cualquier otra medida. Su coste es 0 y no reduce ni la probabilidad ni el impacto de ninguna amenaza, lo que la convierte en el punto de partida para el cálculo del riesgo residual. Gracias a su inclusión, el modelo puede determinar de forma objetiva si la implementación de una contramedida compensa su coste, en comparación con la alternativa de no actuar.

La definición estructurada y detallada de las contramedidas constituye un pilar fundamental en este Trabajo de Fin de Grado, no solo como parte del diseño metodológico, sino también como elemento funcional indispensable en el modelo de optimización. La representación en formato JSON no solo ha permitido una integración eficiente con Python, sino que también ha facilitado el manejo automatizado y reproducible de los datos relacionados con las medidas de seguridad.

3.2.2. AMENAZAS

Las amenazas constituyen el elemento central del riesgo que se aborda en este proyecto, donde su detección y valoración resulta imprescindible para el análisis sistemático desarrollado siguiendo MAGERIT. Su identificación forma parte del proceso fundamental de

análisis de riesgos, representando eventos adversos que pueden aprovechar vulnerabilidades para comprometer la seguridad de los activos del sistema de información.

Se ha definido un conjunto de 7 amenazas principales para este proyecto, seleccionadas por su relevancia en el actual panorama de ciberseguridad y su capacidad de impacto sobre activos críticos organizacionales. La selección se ha realizado considerando las amenazas más frecuentes identificadas en estudios del sector, garantizando correspondencia directa con el catálogo oficial MAGERIT ubicado en el Catálogo de Elementos [40]. Tal y como se ha comentado antes, este número es modificable para cada usuario en función de su contexto y aplicación específica.

Cada amenaza se ha modelado en el código Python mediante una estructura de diccionario que incluye dos parámetros: probabilidad e impacto. Esta aproximación permite cuantificar matemáticamente el riesgo y facilita la integración con el algoritmo de optimización desarrollado. La variable amenazas se define como diccionario donde cada clave representa identificador único (T1 a T7) y cada valor contiene atributos probabilidad e impacto, calculando así de forma automatizada el riesgo base antes de aplicar contramedidas.

La probabilidad representa la frecuencia estimada de ocurrencia de cada amenaza, expresada como valor decimal entre 0 y 1, donde valores próximos a 1 indican mayor probabilidad de materialización. El impacto cuantifica las consecuencias económicas y operativas resultantes de la materialización de la amenaza, expresado en unidades monetarias normalizadas mediante PILAR para facilitar comparación y priorización de riesgos.

Las amenazas implementadas abarcan desde el **acceso no autorizado** (T1), que representa entrada ilícita a sistemas explotando debilidades en autenticación y autorización, hasta la **alteración de datos** (T2), definida como modificación no autorizada de información aplicable tanto a PCs como routers. El **código malicioso** (T3) está contemplado en el catálogo oficial de MAGERIT incluyendo virus, troyanos y variantes que se camuflan como software legítimo, mientras que el **agotamiento de recursos** (T4) se corresponde con saturación de CPU, RAM y red. El **fallo de hardware** (T5) presenta equivalencia con fallos de componentes físicos, incluyendo discos, impresoras y routers. La **pérdida de disponibilidad** (T6) está contemplada como amenaza separada de interrupción del servicio, mientras que la **interceptación de comunicaciones** (T7) se refiere a la amenaza de que una persona o entidad no autorizada acceda y escuche o capture la información que se transmite a través de un canal de comunicación.

La cuantificación de cada amenaza se basa en evaluar su probabilidad de ocurrencia y el impacto potencial en caso de materialización. Los valores se han establecido considerando estudios sectoriales, bases de datos de incidentes de seguridad y experiencia acumulada en proyectos similares. Esta aproximación mantiene coherencia con MAGERIT y facilita integración con herramientas como PILAR para análisis cuantitativo de riesgos.

Código	Amenaza	Probabilidad	Impacto
T1	Acceso no autorizado	0.5	500
T2	Alteración de datos	0.4	500
T3	Código malicioso	0.7	600
T4	Agotamiento de recursos	0.4	700
T5	Fallo de hardware	0.3	400
T6	Pérdida de disponibilidad	0.4	460
T7	Interceptación de comunicaciones	0.6	540

Tabla 3.3 - Catálogo de amenazas

La tabla 3.3 presenta las amenazas implementadas en el sistema de optimización, donde cada valor de probabilidad representa la frecuencia de que una amenaza se materialice, y el impacto representa la magnitud de las consecuencias negativas que dicha amenaza tendría sobre los activos afectados. Este impacto se expresa mediante un valor numérico proporcional a la severidad del daño, considerando aspectos como pérdidas económicas, interrupción de servicios o afectación a la confidencialidad, integridad o disponibilidad. Estos valores sirven como entrada fundamental para el algoritmo de minimización del riesgo residual, permitiendo la evaluación cuantitativa de diferentes estrategias de mitigación.

La definición estructurada de estas amenazas constituye un elemento fundamental para la efectividad del modelo de optimización desarrollado, proporcionando la base cuantitativa necesaria para la toma de decisiones informadas en la gestión de riesgos de ciberseguridad. Su integración con el catálogo de contramedidas permite establecer relaciones precisas entre amenazas y medidas de protección, optimizando la asignación de recursos de seguridad bajo restricciones presupuestarias y maximizando la reducción del riesgo residual tanto a nivel global como por activo individual.

La correspondencia directa con el catálogo MAGERIT garantiza trazabilidad metodológica y validez de los análisis realizados, facilitando tanto comprensión del modelo como su aplicabilidad en entornos reales donde se requiera cumplimiento normativo. Esta alineación metodológica refuerza la solidez del sistema desarrollado y asegura que los resultados obtenidos sean coherentes con las mejores prácticas establecidas en la gestión de riesgos de sistemas de información.

3.2.3. VULNERABILIDADES

Las vulnerabilidades representan debilidades presentes en los activos que facilitan la materialización de amenaza. Su identificación y cuantificación forma parte esencial del

proceso sistemático desarrollado en este proyecto, proporcionando una base técnica necesaria para calcular incrementos en probabilidad de ocurrencia de eventos adversos. La correcta modelización de vulnerabilidades permite establecer relaciones precisas entre debilidades específicas y amenazas concretas, facilitando así evaluación más precisa del riesgo residual del sistema.

Se ha definido un conjunto de 5 vulnerabilidades principales basadas en *Common Vulnerabilities and Exposures* (CVE) reales, seleccionadas por su relevancia en el panorama actual de ciberseguridad y su capacidad de impacto sobre las amenazas identificadas [41]. La selección se ha realizado considerando que muestren concordancia con las amenazas y activos seleccionados, garantizando así aplicabilidad práctica del modelo desarrollado.

Cada vulnerabilidad ha sido modelada dentro del código Python mediante una estructura de diccionario que incluye el parámetro fundamental *aumento_prob*, que cuantifica el incremento de probabilidad que aporta la vulnerabilidad a las amenazas asociadas. Esta aproximación permite la cuantificación matemática directa del impacto de cada debilidad sobre el riesgo base del sistema, facilitando su integración con el algoritmo de optimización desarrollado. La variable vulnerabilidades se define como diccionario donde cada clave representa un identificador CVE único y cada valor contiene el atributo *aumento_prob*, permitiendo así el cálculo automatizado de probabilidades ajustadas antes de la aplicación de contramedidas.

El mapeo de vulnerabilidades a amenazas se establece mediante el diccionario *cve_amenaza_map*, que define las relaciones específicas entre cada CVE y las amenazas que puede potenciar. Esta estructura permite que una vulnerabilidad pueda afectar múltiples amenazas simultáneamente, reflejando la realidad de los entornos de ciberseguridad donde una sola debilidad puede abrir múltiples vectores de ataque.

La cuantificación de cada vulnerabilidad se basa en la evaluación de su capacidad para incrementar la probabilidad de materialización de amenazas específicas. Los valores de incremento han sido establecidos considerando análisis de *Common Vulnerability Scoring System* (CVSS), estudios sectoriales y bases de datos especializadas en vulnerabilidades. Esta aproximación mantiene coherencia con la metodología MAGERIT y facilita integración con herramientas como PILAR para análisis cuantitativo de riesgos.

La tabla 3.4 presenta las vulnerabilidades implementadas en el sistema de optimización, donde cada valor de aumento de probabilidad representa el incremento directo que aporta la vulnerabilidad a la frecuencia estimada de ocurrencia de las amenazas asociadas. Estos valores se integran automáticamente en el cálculo del riesgo base mediante la suma acumulativa de incrementos por amenaza, permitiendo modelar escenarios donde múltiples vulnerabilidades afectan la misma amenaza.

CVE	Descripción	Aumento Prob.	Amenazas Afectadas
CVE-2024-0012	Autenticación eludida	+0.47	T1
CVE-2024-9474	Escalada de privilegios	+0.30	T1, T3
CVE-2024-36462	Consumo incontrolado de recursos	+0.54	T4
CVE-2024-45700	Negación de servicio (DoS) por agotamiento de recursos	+0.40	T4
CVE-2024-42333	Alteración de datos por fuga de memoria	+0.35	T2

Tabla 3.4 - Vulnerabilidades y sus propiedades

La correspondencia directa con identificadores CVE reales garantiza trazabilidad técnica y validez de los análisis realizados, facilitando tanto comprensión del modelo como su aplicabilidad en entornos reales donde se requiera gestión específica de vulnerabilidades conocidas. Esta alineación con estándares internacionales refuerza la solidez del sistema desarrollado y asegura que los resultados obtenidos sean coherentes con las mejores prácticas establecidas en la gestión de vulnerabilidades de sistemas de información.

3.2.4. ACTIVOS

En el contexto de la gestión de riesgos de ciberseguridad, los activos constituyen los elementos de valor fundamental que requieren protección dentro del sistema de información desarrollado en este proyecto. Su caracterización precisa resulta esencial para establecer el contexto operativo sobre el cual actúan las amenazas identificadas y hacia el cual se dirigen las contramedidas seleccionadas mediante el algoritmo de optimización. La definición adecuada de estos elementos permite evaluar de forma cuantitativa el riesgo residual tanto a nivel individual como global del sistema.

Se ha establecido un conjunto de 5 activos principales que conforman una topología de red representativa de entornos organizacionales típicos, seleccionados para abarcar diferentes categorías de equipos y funcionalidades críticas presentes en sistemas de información empresariales. Esta selección incluye equipos informáticos de usuario final, dispositivos compartidos de red, servidores de almacenamiento de datos y equipos de conectividad, garantizando así una representación realista de los componentes que habitualmente requieren protección en infraestructuras tecnológicas actuales.

La modelización de cada activo se ha implementado en el código Python mediante estructura de diccionario que especifica las amenazas particulares a las que se encuentra expuesto cada elemento, estableciendo relaciones directas entre componentes físicos y vectores de riesgo específicos. Esta aproximación facilita el cálculo del riesgo base individual de cada activo e integra de forma eficiente con el algoritmo de optimización desarrollado. La variable activos se define como diccionario donde cada clave representa el identificador único (A1 a A5) y cada valor contiene el atributo amenazas, especificando qué amenazas del catálogo (T1 a T7) pueden materializarse sobre cada elemento particular de la red.

La topología implementada está formada por **PC1** (A1), que representa un equipo informático de usuario expuesto a acceso no autorizado y alteración de datos por su función de procesamiento de información sensible, **PC2** (A2), que constituye un terminal de usuario afectado por código malicioso y agotamiento de recursos debido a su exposición a descargas y ejecución de software. La **impresora** (A3) funciona como dispositivo compartido expuesto a fallos de hardware y pérdida de disponibilidad por su naturaleza de recurso común y dependencia de componentes físicos críticos. El **Servidor de Almacenamiento** (A4) representa el elemento más crítico del sistema, responsable del almacenamiento centralizado de datos y expuesto tanto a interceptación de comunicaciones como a acceso no autorizado debido a la concentración de información valiosa que gestiona. El **Router** (A5) actúa como elemento de conectividad estratégico expuesto a alteración de datos y código malicioso por su función de gestión del tráfico de red y su posición central en la infraestructura de comunicaciones.

La caracterización de cada activo se fundamenta en su función operativa específica, criticidad para la continuidad del negocio y exposición particular a amenazas identificadas en el catálogo desarrollado. Los valores asignados se han establecido considerando estudios de infraestructuras tecnológicas típicas, análisis de vectores de ataque comunes documentados en la literatura especializada y experiencia acumulada en proyectos de gestión de riesgos similares.

La tabla 3.5 presenta los activos implementados en el sistema de optimización, donde cada elemento se relaciona con amenazas específicas según su naturaleza funcional y posición dentro de la arquitectura de red. Esta distribución permite que el algoritmo evalúe el impacto diferenciado de las contramedidas sobre cada categoría de activo, optimizando la protección según las características particulares y la criticidad relativa de cada elemento del sistema.

Código	Activo	Tipo	Amenazas Expuestas
A1	PC1	Equipo informático de usuario.	T1, T2
A2	PC2	Equipo informático de usuario.	T3, T4

Código	Activo	Tipo	Amenazas Expuestas
A3	Impresora	Dispositivo de impresión compartido.	T5, T6
A4	Servidor de almacenamiento	Servidor de datos.	T7, T1
A5	Router Principal	Equipo de red.	T2, T3

Tabla 3.5 - Activos definidos

El modelo matemático implementado en Python utiliza esta estructura para calcular el riesgo base individual de cada activo mediante la suma de riesgos específicos de las amenazas asociadas, aplicando posteriormente las reducciones correspondientes según las contramedidas seleccionadas por el proceso de optimización. Esta aproximación permite que el algoritmo evalúe automáticamente el beneficio de proteger activos específicos frente al coste de las medidas necesarias, priorizando la asignación de recursos según la criticidad operativa y exposición al riesgo de cada elemento.

La configuración definida refleja un entorno representativo donde coexisten equipos de diferentes tipos y funciones operativas, desde terminales de usuario final hasta infraestructura crítica de red y almacenamiento de datos. Esta diversidad permite validar la capacidad del sistema para gestionar escenarios heterogéneos y adaptar la selección de contramedidas a las necesidades específicas de protección de cada tipo de activo, maximizando la eficiencia de la inversión en seguridad bajo restricciones presupuestarias definidas.

La estructuración detallada de estos activos constituye el elemento fundamental para la efectividad del modelo de optimización desarrollado, proporcionando contexto operativo necesario para la evaluación realista de riesgos y la selección informada de contramedidas de protección. Su integración con las amenazas permite establecer relaciones precisas entre elementos físicos de la red, vectores de ataque potenciales y medidas de protección disponibles, facilitando así una gestión integral y eficiente del riesgo residual en el sistema de información analizado.

3.2.5. PRESUPUESTO MÁXIMO

El presupuesto máximo constituye el último flujo de entrada de la arquitectura del sistema y representa la restricción fundamental bajo la cual opera el algoritmo de selección de contramedidas. Este parámetro se introduce dinámicamente mediante interacción directa con el usuario a través de un *input()* implementado en Python, permitiendo adaptar el análisis a diferentes escenarios presupuestarios sin necesidad de cambiar el código fuente.

La flexibilidad de este enfoque facilita la evaluación de múltiples estrategias de inversión en seguridad y permite comparar el impacto de diferentes niveles de financiación sobre la reducción del riesgo residual del sistema. El valor introducido actúa como restricción principal en el modelo desarrollado mediante PuLP, formulándose matemáticamente mediante la ecuación que garantiza que la suma total de costes de las contramedidas seleccionadas no exceda el límite establecido.

Cada contramedida tiene un coste basado en PILAR asegurando coherencia en los valores dados. La normalización mediante PILAR permite estandarizar estos costes heterogéneos en valores comparables, facilitando la evaluación objetiva entre diferentes tipos de contramedidas.

La implementación de este componente refuerza la aplicabilidad práctica del sistema, haciendo que las soluciones propuestas sean tanto técnicamente efectivas como económicamente viables dentro de cada organización. Este valor refleja las limitaciones reales que enfrentan las organizaciones, donde los recursos no solo son económicos sino también humanos, tecnológicos y operacionales.

3.3. PLANTEAMIENTO DEL PROBLEMA DE OPTIMIZACIÓN

La problemática reside en traducir los principios metodológicos de MAGERIT a un problema de optimización formal, es decir, estudiar todos los elementos que componen la gestión de riesgos de ciberseguridad y ponerlos en el contexto adecuado con una función objetivo, variables de decisión y restricciones, componentes fundamentales del problema.

El modelo matemático implementado en Python utiliza los parámetros presentados en el apartado de amenazas (probabilidad e impacto) para calcular el riesgo potencial mediante la fórmula:

$$Riesgo\ potencial\ =\ Probabilidad\ \times\ Impacto$$

A este cálculo se sumará el aumento de probabilidad de su CVE correspondiente, en el caso de que existiese. Así, las probabilidades se ajustan del siguiente modo, aplicando posteriormente un límite máximo de 1.0 para mantener coherencia probabilística:

$$Probabilidad\_ajustada\ =\ Probabilidad\_base\ +\ \sum aumento\_prob$$

Esta estructura permite que el algoritmo de optimización evalúe automáticamente el impacto de vulnerabilidades específicas sobre cada amenaza y seleccione contramedidas que mitiguen tanto el riesgo base como el incremento asociado a las debilidades identificadas.

La **función objetivo** busca minimizar el riesgo residual total de la red, según MAGERIT. El cálculo de este para un activo concreto es:

$$Riesgo\ residual_i = (Prob_i \times Imp_i) - (\sum_{c \in C_i} x_c \cdot Reducción\ de\ Prob_{c,i} + \sum_{c \in C_i} x_c \cdot Reducción\ de\ Imp_{c,i})$$

Donde C_i es el conjunto de medidas aplicables para el activo i. En este contexto *Prob* e *Imp* quieren decir Probabilidad e Impacto, respectivamente.

La minimización global es expresada como:

$$\min \sum_{i\ \epsilon\ Activos} Riesgo\ residual_i$$

Para llegar a esta fórmula las contramedidas aplican un tipo de reducción según su tipo de mitigación. La tabla 3.6 explica cómo se calculan las reducciones descritas en la función objetivo:

Campo	Tipo de mitigación	Modificación del riesgo
"reduccion_probabilidad"	Preventiva	$Probabilidad_{residual} = Probabilidad_{Base} \cdot (1 - \alpha)$
"reduccion_impacto"	Reactiva	$Impacto_{residual} = Impacto_{Base} - \beta$

Tabla 3.6 - Reducción de riesgo por las contramedidas

Para definir este problema con PuLP se tiene reservada la palabra *LpProblem*, que es usada en Python para definir un problema de optimización. A esta clase principal se le pasa la constante *LpMinimize* que dice que el cálculo que se está modelando busca minimizar la función objetivo.

Las **variables de decisión** corresponden a cada contramedida dentro del catálogo recogido en el archivo *contramedidas.json*. Estas son de tipo binaria, pueden obtener sólo dos valores (0 ó 1). Si la contramedida tiene asignada un 1 quiere decir que ha sido aplicada y por lo tanto se encuentra activa. En cambio, si el valor asociado es un 0 la contramedida no habrá sido seleccionada. Para dar esta asignación en Python se reserva la palabra *LpVariable,* a la que se le asigna la constante de categoría con *LpBinary.*

Por último, la **restricción** aplicada en este caso es la del presupuesto límite. Cada contramedida tiene un coste basado en PILAR [42] asegurando coherencia en los valores dados.

El presupuesto límite es introducido por el usuario mediante un *input()*. La suma de costes de las contramedidas seleccionadas en ningún caso puede superar el número introducido por el usuario:

$$\sum(coste \in Contramedidas) \leq Presupuesto\_máximo$$

La integración de todos estos componentes permite que el sistema evalúe automáticamente múltiples configuraciones de contramedidas, seleccionando aquella combinación que proporcione la máxima reducción del riesgo residual dentro del presupuesto disponible. Esta aproximación automatizada supera las limitaciones de los métodos cualitativos tradicionales, proporcionando decisiones objetivas y cuantificadas para la gestión de riesgos de ciberseguridad.

La formulación desarrollada mantiene coherencia con la metodología MAGERIT mientras incorpora capacidades de optimización avanzadas, facilitando la toma de decisiones informadas en entornos donde los recursos son limitados y la efectividad de las medidas de seguridad debe ser maximizada. Esta integración entre principios metodológicos establecidos y técnicas de optimización modernas constituye la base fundamental sobre la cual opera todo el sistema desarrollado en este proyecto.

3.4. FLUJOS DE SALIDA

El sistema de optimización desarrollado genera tres flujos de salida principales que permiten analizar y validar los resultados obtenidos tras el proceso de selección de contramedidas. Estos flujos proporcionan diferentes perspectivas del análisis realizado, desde representaciones visuales interactivas hasta informes detallados con métricas comparativas, facilitando tanto la comprensión técnica como la toma de decisiones estratégicas en la gestión de riesgos.

La arquitectura de salida se ha diseñado para satisfacer diferentes necesidades de usuario, desde técnicos especializados que requieren datos granulares hasta directivos que necesitan síntesis ejecutivas de los resultados. Cada flujo complementa a los demás proporcionando visión integral del estado del sistema antes y después de la optimización.

El primer flujo de salida consiste en representación gráfica generada mediante la librería NetworkX que modela la topología de red como grafo dirigido. Esta visualización permite observar de forma inmediata las relaciones entre activos, amenazas, vulnerabilidades CVE y contramedidas seleccionadas, facilitando el análisis visual de la estructura de seguridad implementada.

La función *crear_grafo()* implementada en Python construye automáticamente la estructura del grafo agregando nodos diferenciados por tipo y estableciendo aristas que representan las relaciones de exposición y mitigación. Los activos se representan como nodos azules conectados secuencialmente, mientras que las amenazas aparecen como nodos rojos vinculados a los activos que afectan. Las vulnerabilidades CVE se muestran como nodos morados que incrementan la probabilidad de amenazas específicas, aumentado el riesgo potencial de cada activo y las contramedidas seleccionadas aparecen como nodos verdes conectadas a las amenazas que mitigan. En la figura 3.2 se puede ver un ejemplo de cómo se representa una red con esta biblioteca.

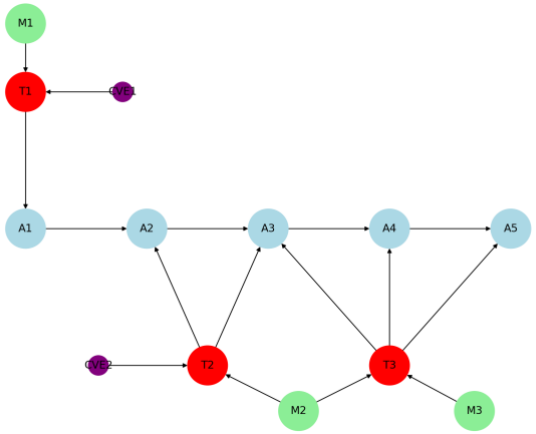

Figura 3.2 - Ejemplo de red con NetworkX

El segundo componente es un archivo de texto estructurado denominado *resultados.txt* que contiene el detalle completo del análisis realizado y las decisiones tomadas por el algoritmo de optimización. Este archivo proporciona trazabilidad completa del proceso y permite auditorías posteriores o análisis comparativos entre diferentes ejecuciones del sistema.

La estructura del archivo incluye inicialmente un inventario detallado de la configuración analizada, listando cada activo con sus amenazas asociadas para establecer el contexto base del análisis. Posteriormente se presenta el riesgo residual inicial calculado antes de aplicar contramedidas, proporcionando la línea base para evaluar la efectividad de las medidas seleccionadas.

La sección central del informe detalla las contramedidas específicas seleccionadas por el algoritmo de optimización bajo la restricción presupuestaria definida. Esta información resulta fundamental para la implementación práctica de las recomendaciones del sistema. A continuación, se presenta el análisis granular del riesgo residual por activo individual, permitiendo identificar elementos que requieren atención adicional o validar la distribución equilibrada de la protección.

La implementación utiliza redirección de salida estándar mediante *sys.stdout* para automatizar la generación del archivo, garantizando consistencia en el formato y completitud de la información registrada. El mecanismo de restauración automática de la salida estándar mediante bloques *try-finally* asegura la robustez del proceso incluso en caso de errores durante la generación del informe [43].

El tercer flujo proporciona el valor numérico del riesgo residual final calculado por el modelo de programación lineal, representando la métrica central que cuantifica la efectividad de la solución optimizada. Este valor resulta del proceso de minimización ejecutado por PuLP considerando todas las restricciones definidas y representa el nivel mínimo de riesgo alcanzable con el presupuesto disponible.

El cálculo se realiza mediante la evaluación automática de la función objetivo una vez resuelto el problema de optimización, proporcionando un valor objetivo que permite comparaciones directas entre diferentes escenarios presupuestarios o configuraciones de red.

La accesibilidad de este resultado a través de *modelo.objective.value()* facilita su integración en análisis posteriores o sistemas de monitorización continua.

Este flujo de salida numérico complementa las representaciones visuales y textuales proporcionando la base cuantitativa necesaria para validar el cumplimiento de objetivos de reducción de riesgo y justificar las inversiones en seguridad realizadas. Su naturaleza precisa permite establecer umbrales de riesgo aceptable y evaluar la necesidad de ajustes presupuestarios para alcanzar niveles de protección específicos.

La integración de estos tres flujos de salida proporciona una visión completa y multidimensional de los resultados del análisis.

3.5. CÓDIGO FUENTE

La arquitectura descrita ha sido implementada, tal como se ha indicado previamente, en diferentes ficheros: *gestionderiesgos.py* para el desarrollo principal en Python, *contramedidas.json* para el catálogo de medidas de seguridad y un archivo de salida en *resultados.txt* que recoge toda la información generada durante la ejecución. A ello se suma el grafo que genera para la visualización del informe completo y tener un plano general del problema.

Todos estos ficheros han sido subidos a un repositorio en GitHub y estarán accesibles a través del siguiente enlace (Ver Anexo C: Documentación del proyecto en Github):

https://github.com/JavierMontesinos/Sistema-de-optimizacion-de-contramedidas

Gracias a ello, es posible consultar en detalle cómo se ha resuelto el problema de optimización mediante Python y cómo se realiza la integración con el fichero JSON para conformar el sistema completo de optimización.

La figura 3.3 muestra cómo queda el repositorio ya subido a la página web.

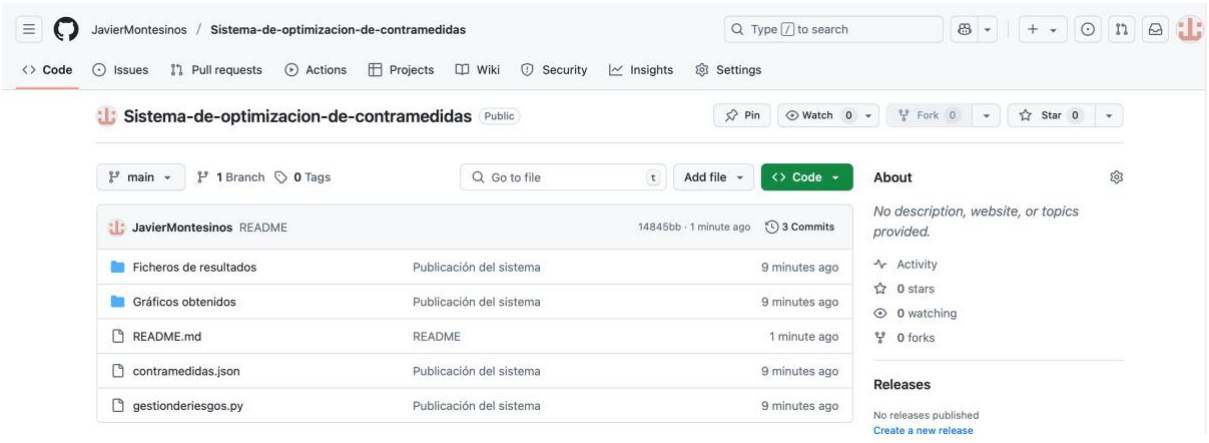

Figura 3.3 - Repositorio de GitHub

4. VALIDACIÓN

Para la redacción de este capítulo se ha probado el programa definido en el desarrollo de la memoria de este Trabajo de Fin de Grado (figura 3.1), los resultados obtenidos por el mismo serán comparados y analizados con los cálculos que se esperarían alcanzar.

La validación del proyecto será probada de forma independiente realizando casos de uso, situaciones llevadas a la realidad de una organización o empresa. Primero se presentará cada uno, seguidamente se comentarán los resultados obtenidos de la simulación hecha en el código implementado en Python y se verificará si el resultado obtenido es válido.

La validación del sistema desarrollado permite evaluar tanto la selección de contramedidas como la reducción efectiva del riesgo residual bajo diferentes restricciones económicas. Cada caso de uso permitirá analizar el comportamiento del algoritmo de optimización implementado mediante PuLP, verificando que las soluciones propuestas respeten las restricciones presupuestarias establecidas y minimicen efectivamente el riesgo residual total del sistema. La progresión de los casos de uso, con presupuestos distintos, permitirá demostrar cómo se adapta el modelo de optimización ante las restricciones y su capacidad para adaptarse a diferentes escenarios organizacionales reales.

Hay cuatro casos de uso definidos, estos se diferencian por el presupuesto introducido por el usuario con el *input()*, el primero tendrá un presupuesto máximo de 8, para ver que se selecciona bien la contramedida especial M0-DoNothing, el segundo manejará la restricción de un presupuesto bajo para que se seleccione un número pequeño de contramedidas, para el tercero se seguirá subiendo el presupuesto para disminuir más el riesgo residual final, y por último, se comparará cómo varía la selección de contramedidas ante una misma amenaza con dos presupuestos distintos. Se irá viendo progresivamente cómo, al subir el presupuesto, la red definida en el capítulo anterior será cada vez más segura, reduciendo el riesgo de cada amenaza.

El análisis de los resultados se fundamenta en el cálculo del riesgo residual según la fórmula establecida, comparando los valores iniciales y finales para determinar la eficacia de las soluciones propuestas bajo diferentes restricciones presupuestarias. Los resultados se contrastarán para verificar la coherencia del sistema y demostrar la capacidad del modelo para priorizar recursos de seguridad de manera objetiva en entornos reales de gestión de riesgos.

Para estos casos de uso se ha definido una topología de red específica que incluye cinco activos interconectados representando un entorno organizacional típico. Esta red cuenta con dos PCs cada uno de un usuario (A1 y A2), una impresora (A3), un servidor de almacenamiento (A4) y un router principal (A5). Estos están apuntados por las amenazas y vulnerabilidades definidas en el desarrollo, todos estos elementos están representados en la figura 4.1. Esta es la red sobre la que se va a validar el sistema desarrollado, proporcionando un escenario realista donde el problema de optimización implementado debe resolverse. Este será una constante a lo largo de todo el proceso de validación lo que cambia serán las

contramedidas seleccionadas dentro del catálogo y el presupuesto mínimos como se ha mencionado antes.

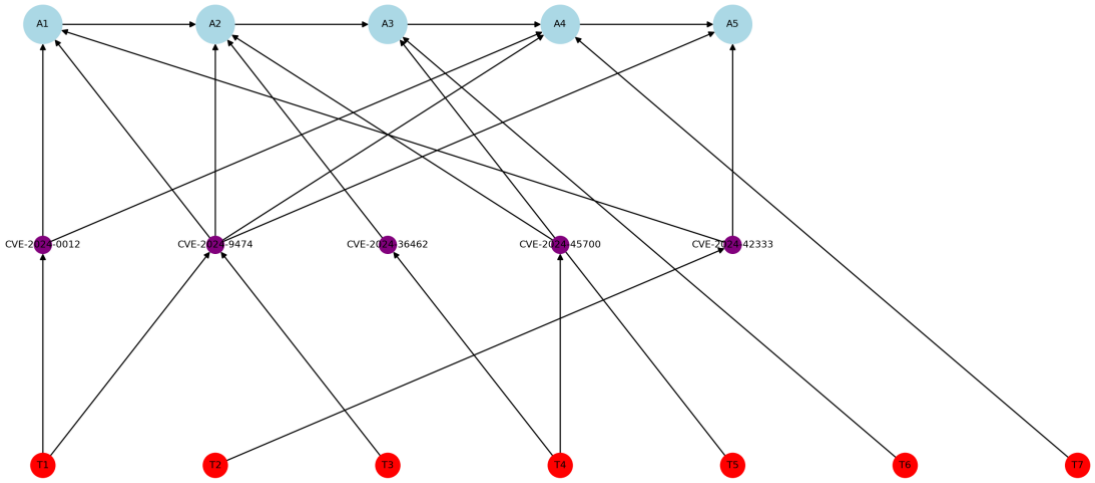

Figura 4.1 - Red representada por NetworkX

4.1. CÁLCULO DE RIESGOS POTENCIALES INICIALES

Para establecer la línea base del análisis de riesgos, se ha procedido al cálculo del riesgo potencial inicial de cada activo del sistema antes de la aplicación de cualquier contramedida. Este cálculo se fundamenta en la metodología MAGERIT y utiliza la fórmula de PILAR donde el riesgo potencial se obtiene mediante la multiplicación de la probabilidad por el impacto de cada amenaza. La importancia de este cálculo radica en que proporciona el valor de referencia necesario para evaluar posteriormente la efectividad de las contramedidas seleccionadas por el algoritmo de optimización.

El proceso de cálculo considera tanto las probabilidades base de cada amenaza como los incrementos introducidos por las vulnerabilidades CVE identificadas en el sistema. Las vulnerabilidades CVE actúan como factores multiplicadores que incrementan la probabilidad de materialización de amenazas específicas, reflejando así el impacto real de las debilidades técnicas presentes en cada activo. Cuando múltiples vulnerabilidades afectan la misma amenaza, sus incrementos se suman de forma acumulativa, aplicando posteriormente un límite máximo de 1.0 para mantener la coherencia probabilística del modelo matemático implementado.

El riesgo potencial esperado en esta red es de 4278.0 y coincide con el riesgo calculado que se desglosa en la tabla 4.1

Activo	Amenazas	CVE Asociado	Aumento Probabilidad	Probabilidad Final	Impacto	Riesgo Potencial Inicial
A1 PC1	T1, T2	CVE-2024-0012 CVE-2024-9474 CVE-2024-42333	+ 0.77, + 0.35	1.00, 0.75	500, 500	875.0
A2 PC2	T3, T4	CVE-2024-9474 CVE-2024-36462 CVE-2024-45700	+0.30, +0.94	1.00, 1.00	600, 700	1300.0
A3 Impresora	T5, T6	Sin CVE	0.00	0.30, 0.40	400, 460	304.0
A4 Servidor	T7, T1	CVE-2024-0012 CVE-2024-9474	+0.77	0.60, 1.00	540, 500	824.0
A5 Router	T2, T3	CVE-2024-42333 CVE-2024-9474	+0.35, +0.30	0.75, 1.00	500, 600	975.0

Tabla 4.1 - Cálculo de riesgos residuales iniciales

Los datos de la tabla 4.1 presentan diferentes valores de riesgo entre activos: A2 presenta la mayor exposición con 1300.0 unidades de riesgo, representando el 30.4% del riesgo total del sistema. Esta elevada exposición se debe principalmente a que ambas amenazas que lo afectan que alcanzan probabilidades máximas de 1.0 debido a los incrementos causados por los CVEs que combinadas con impactos considerables de 600 y 700, respectivamente. A3 registra el menor riesgo con 304.0, (7.1% del total), ya que no hay vulnerabilidades que potencien la probabilidad, manteniendo así su base de 0.30 y 0.40.

Los activos A1, A4 y A5 presentan valores intermedios, evidenciando el impacto diferenciado de las vulnerabilidades CVE sobre cada configuración específica. El comportamiento de las vulnerabilidades muestra que CVE-2024-0012 y CVE-2024-9474 afectan conjuntamente a la amenaza T1 en los activos A1 y A4, elevando su probabilidad al máximo posible, mientras que CVE-2024-36462 y CVE-2024-45700 impactan significativamente sobre T4 en A2 con una probabilidad final de 0.94. Esta distribución heterogénea del riesgo establece la base cuantitativa necesaria para que el algoritmo de optimización priorice la protección de activos según su exposición relativa y los recursos disponibles en cada caso de uso.

4.2. CASO DE USO 1: PRESUPUESTO INSUFICIENTE

Para este primer caso de validación se ha configurado el sistema con un presupuesto máximo de 8, lo que permite verificar el comportamiento del algoritmo de optimización cuando no hay recursos disponibles para implementar contramedidas. Este escenario específico ha sido diseñado para comprobar que el sistema selecciona correctamente la contramedida especial

M0-DoNothing, que representa la opción nula para preservar recursos sin alterar el perfil de riesgo del sistema.

El caso de presupuesto nulo constituye un escenario fundamental para validar la correcta implementación de las restricciones presupuestarias en el modelo de programación lineal desarrollado mediante PuLP. Cuando el presupuesto se establece en 8, el algoritmo debe evaluar sistemáticamente las 26 contramedidas disponibles en el catálogo y determinar que únicamente M0-DoNothing cumple con la condición matemática de que la suma de costes no exceda el límite establecido. Esta contramedida especial, con coste 0 y sin capacidad de reducción de probabilidad ni impacto, representa la opción que preserva recursos sin modificar las condiciones de riesgo del sistema.

La tabla 4.2 muestra la ausencia total de variación en los valores de riesgo entre el estado inicial y final del sistema. Esta coincidencia exacta se explica por las características específicas de M0-DoNothing, que no proporciona ninguna reducción de probabilidad ni de impacto sobre las amenazas identificadas, manteniendo inalteradas las condiciones de exposición de cada activo. El sistema conserva exactamente su nivel de riesgo original, confirmando que la contramedida nula cumple su función específica de preservar recursos sin alterar el perfil de seguridad.

Activo	Riesgo Potencial Inicial	Riesgo Residual Final	Variación
PC1 (A1)	875.0	875.0	0.0
PC2 (A2)	1300.0	1300.0	0.0
Impresora (A3)	304.0	304.0	0.0
Servidor de Almacenamiento (A4)	824.0	824.0	0.0
Router Principal (A5)	975.0	975.0	0.0

Tabla 4.2 - Resultados de caso de uso 1

Presupuesto utilizado: 0 de 8 disponible.

Contramedidas aplicadas: M0-DoNothing

Riesgo residual total = Riesgo potencial inicial: 4278.0

La representación gráfica mostrada en la figura 4.2 confirma visualmente el comportamiento esperado del sistema cuando se aplica un presupuesto nulo. La visualización generada

mediante NetworkX muestra claramente la topología de red con los cinco activos (PC1, PC2, Impresora, Servidor de Almacenamiento y Router Principal) representados como nodos azules interconectados de forma secuencial.

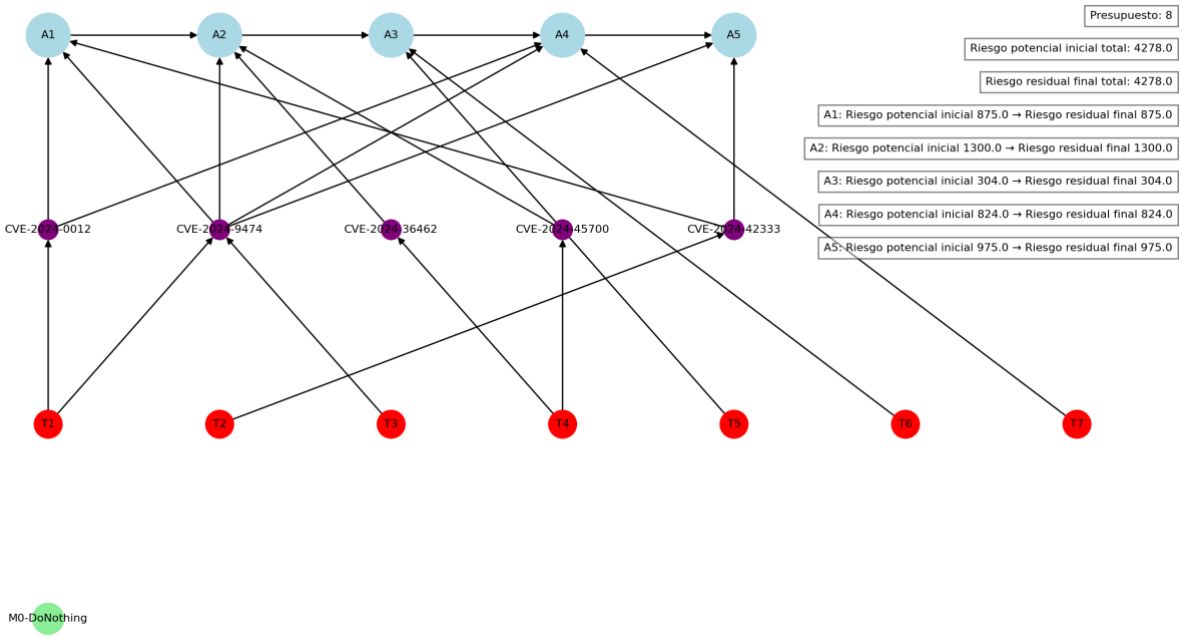

Figura 4.2 - Gráfico de caso de uso 1 por NetworkX

Se observa que únicamente aparece la contramedida M0-DoNothing como el único nodo verde presente en el sistema, confirmando que no se han seleccionado medidas de protección activas para mitigar las amenazas identificadas. Las amenazas aparecen como nodos rojos distribuidos y conectados a los activos correspondientes, mientras que las vulnerabilidades CVE se visualizan como nodos morados que incrementan la probabilidad de amenazas específicas.

Los valores superpuestos en la parte superior derecha de la visualización muestran de forma clara los resultados del análisis: presupuesto de 0, riesgo potencial inicial de 4278.0 y riesgo residual final de 4278.0, confirmando que no ha habido modificación alguna en el nivel de exposición del sistema. Esta coincidencia entre valores iniciales y finales valida que M0-DoNothing cumple su función específica de no alterar el perfil de riesgo cuando no hay recursos disponibles para implementar contramedidas activas.

La representación permite observar de manera inmediata que el sistema mantiene su estado base sin protección adicional, estableciendo así la línea de referencia fundamental para evaluar la efectividad de las contramedidas en los casos de uso posteriores con mayor disponibilidad presupuestaria.

4.3. CASO DE USO 2: PRESUPUESTO BAJO (COSTE = 45)

En este segundo caso de validación se fija un presupuesto máximo de 45 para comprobar que el algoritmo selecciona correctamente las contramedidas más efectivas dentro de un escenario de recursos moderados. El programa evalúa todas las combinaciones posibles de contramedidas cuyo coste total no exceda la restricción presupuestaria calculando el riesgo residual final de cada activo y del sistema completo tras aplicar los factores de reducción de probabilidad o impacto.

En la Figura 3.2 se muestra, a modo de recordatorio, un diagrama visual de contramedidas, generado con la herramienta Jsoncrack [44]. Este diagrama permite observar de forma estructurada y jerárquica las distintas medidas de seguridad definidas, clasificadas como preventivas o reactivas, junto con sus propiedades clave: tipo, coste, estado de aplicación y los efectos que ejercen sobre la probabilidad o el impacto de amenazas específicas. Esta representación facilita la comprensión de la organización interna del archivo y su integración dentro del modelo de análisis de riesgos.

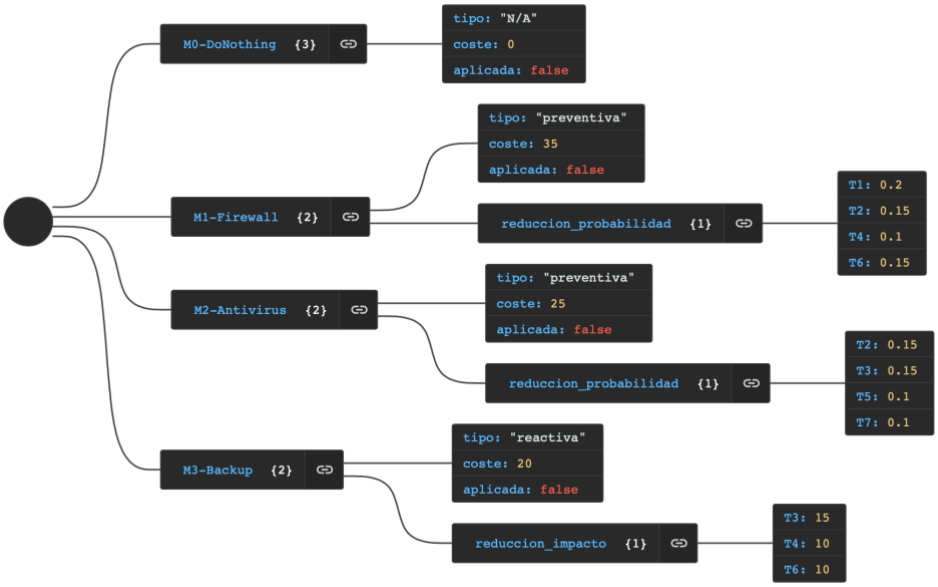

Figura 4.3 - Estructura del fichero contramedidas.json

Al comparar los resultados, el algoritmo determina que la selección óptima es M6-AccessControl y M10-Patching, usando 40 unidades presupuestarias y dejando 5 sin emplear, ya que ninguna alternativa de coste exacto 45 u otro coste menor logra una mayor reducción global del riesgo residual.

El proceso de selección primero calcula el riesgo potencial inicial de cada activo sumando las contribuciones de las amenazas y vulnerabilidades asociadas. Después, genera todas las

combinaciones viables de contramedidas dentro del presupuesto y simula la aplicación de cada combinación para obtener el riesgo residual final. Para terminar, escoge la combinación que maximiza la reducción total del riesgo del sistema. En este caso se comprueba que no existe ninguna contramedida única de coste 45 (por ejemplo, M17-DLP) ni combinación que agote totalmente el presupuesto (como M6-AccessControl + M1-Firewall) que supere la reducción de obtenida con M6-AccessControl y M10-Patching.

La tabla 4.3 recoge el riesgo inicial, final, la reducción absoluta y porcentual para cada activo:

Activo	Riesgo Potencial inicial	Riesgo Residual final	Reducción absoluta	Reducción %
PC1 (A1)	875.0	650.0	225.0	25.7 %
PC2 (A2)	1300.0	1150.0	150.0	11.5 %
Impresora (A3)	304.0	284.0	20.0	6.6 %
Servidor de Almacenamiento (A4)	824.0	616.0	208.0	25.2 %
Router Principal (A5)	975.0	700.0	275.0	28.2 %

Tabla 4.3 - Reducción de riesgo residual individual

Presupuesto utilizado: 40, el máximo posible es de 45 para este caso de uso.

Contramedidas aplicadas: M6-AccessControl y M10-Patching

Riesgo potencial total inicial: 4278.0

Riesgo residual total final: 3400.0

Reducción total: 878.0 (20.5 %)

Este resultado coincide con lo previsto por el diseño del modelo porque M6-AccessControl con coste 10, reduce la probabilidad de la amenaza T1 (Acceso no autorizado) de 0.50 a 0.20 sobre PC1 y Servidor de Almacenamiento, mientras que M10-Patching con coste 30, ajusta las probabilidades de T3 (Código malicioso) y T4 (Agotamiento de recursos) de 0.70 y 0.40 a 0.20 y 0.22 para PC2 y Router Principal, consiguiendo así la máxima reducción conjunta del riesgo residual dentro del límite presupuestario.

El análisis comparativo de estos escenarios alternativos permite extraer conclusiones relevantes sobre el comportamiento del modelo y su aplicabilidad en entornos reales. Lo

primero que se observa es que agotar el presupuesto disponible no implica necesariamente obtener mejores resultados en términos de reducción de riesgo. Tanto la combinación M6-AccessControl + M1-Firewall como la opción de M17-DLP en solitario consumen las 45 unidades presupuestarias, pero ninguna logra superar la efectividad de la solución óptima, que utiliza únicamente 40, llegando a la conclusión de que no basta con gastar el máximo, sino que es imprescindible analizar la cobertura específica de amenazas y la complementariedad de las contramedidas seleccionadas.

Por otro lado, queda claro que la eficacia global del sistema depende en gran medida de que las contramedidas elegidas sean realmente complementarias en los vectores de ataque que mitigan. Las alternativas evaluadas muestran cómo ciertas contramedidas pueden solaparse en sus efectos (por ejemplo, M8-Training tiene un impacto limitado en amenazas de tipo técnico), mientras que otras dejan sin cubrir amenazas críticas (como ocurre con M17-DLP respecto a T1 o T4). Esta falta de complementariedad reduce notablemente la capacidad del sistema para disminuir el riesgo residual de manera eficiente.

Además, los resultados muestran la capacidad del modelo para identificar sinergias entre contramedidas que afectan a activos y amenazas diferentes, optimizando la reducción del riesgo a nivel de sistema completo. En el caso óptimo, la combinación de controles de acceso y parcheo permite abordar amenazas de naturaleza distinta, maximizando la reducción conjunta del riesgo residual y garantizando una protección equilibrada en toda la red.

Para validar la robustez de la solución óptima se compara con tres escenarios alternativos de coste similar como muestra la tabla 4.4:

Combinaciones de contramedidas	Coste	Reducción total	Diferencia vs. óptimo
M6-AccessControl + M1-Firewall	45	830.0	-48.0
M17-DLP (en solitario)	45	760.0	-118.0
M6-AccessControl + M8-Training	30	650.0	-228.0

Tabla 4.4 - Comparativa de selección de contramedidas en caso de uso con bajo presupuesto

En el primer escenario, M6-AccessControl y M1-Firewall agotan el presupuesto, pero no cubre las amenazas T3 ni T4, por lo que la reducción total se queda en 830.0 (–5.5 %).

En el segundo, M17-DLP centra todos los recursos en fugas de información sin mitigar accesos no autorizados ni agotamiento de recursos, alcanzando 760.0 (–13.4 %).

El tercer escenario, M6-AccessControl y M8-Training, sacrifican mitigación de agotamiento de recursos y reduce solo 650.0 unidades (–26.0 %).

Estos resultados confirman que la pareja M6-AccessControl y M10-Patching ofrece la máxima reducción de riesgo residual dentro del presupuesto, dejando un remanente de 5 unidades sin alternativas viables.

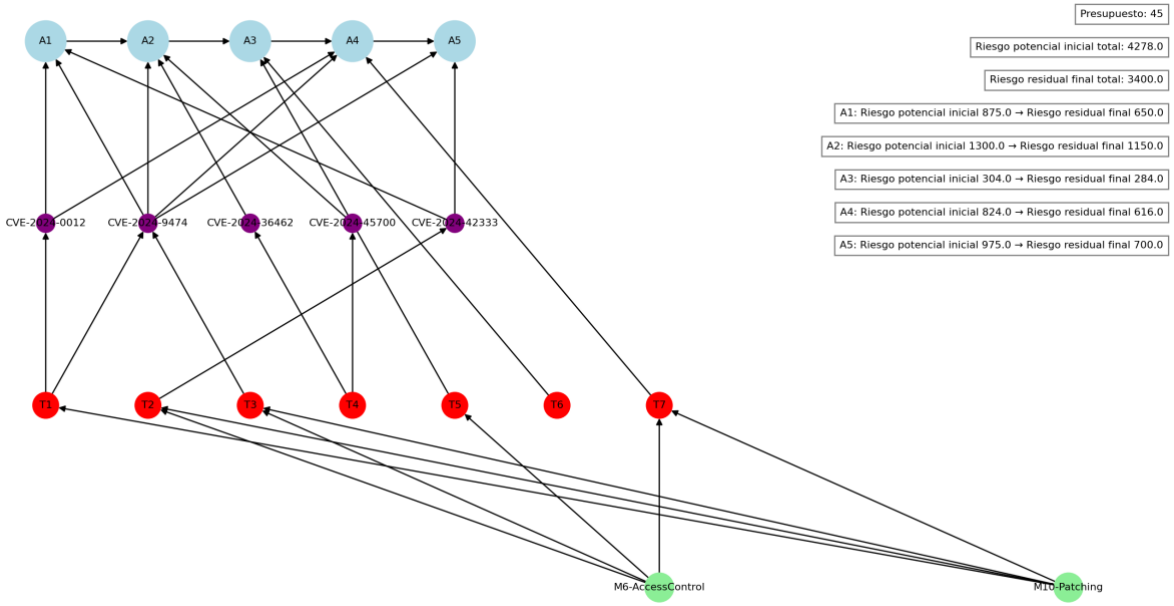

Figura 4.3 - Grafo de NetworkX con contramedidas seleccionadas para un presupuesto de 45

La figura 4.3 muestra la topología resultante tras aplicar M6-AccessControl y M10-Patching, confirmando que la distribución de nodos y aristas refleja fielmente las relaciones de mitigación definidas por el algoritmo de optimización. Se observa que M6-AccessControl están conectado únicamente a las amenazas T2, T3, T5 y T7 que afectan a PC1 y al Servidor de Almacenamiento, validando así su selección para reducir el riesgo en estos dos activos críticos. Del mismo modo, el nodo verde de M10-Patching enlaza con las amenazas T1, T2, T3 y T7 que impactan en PC2 y en el Router Principal, evidenciando que esta contramedida cubre específicamente los vectores de ataque de código malicioso y agotamiento de recursos. Los nodos rojos representan las amenazas, los morados indican las vulnerabilidades identificadas en el sistema y los nodos azules sitúan los activos en la parte superior del grafo.

Para concluir, estos resultados confirman la validez y aplicabilidad práctica del sistema, demostrando que la selección de contramedidas se realiza de manera estratégica, eficiente y acorde con las necesidades reales de una organización. Se cumple así el objetivo de validación independiente planteado en este capítulo.

4.4. CASO DE USO 3: PRESUPUESTO ALTO (COSTE = 200)

En este tercer caso de uso se establece un presupuesto de 200 para comprobar que el algoritmo mantiene su capacidad de selección óptima con un coste considerable donde la

combinatoria de selección de contramedidas es mucho más compleja. El sistema genera todas las combinaciones de contramedidas cuyo coste total iguale o no supere el objetivo presupuestario y se espera que de nuevo seleccione las óptimas minimizando el riesgo residual de cada activo y asegurando la máxima seguridad de la red.

Tras comparar todas las posibles soluciones se concluye que la solución del problema de optimización en este caso son las contramedidas: M1-Firewall, M2-Antivirus, M5-Encryption, M6-AccessControl, M7-VPN, M10-Patching y M16-NetworkSegmentation. Estas conjuntamente tienen un coste total de 200 por lo que ahora sí se llega al máximo del presupuesto y no se deja nada de remanente.

El proceso de cálculo sigue el desarrollo descrito en el primer caso de uso, ofreciendo una cobertura completa ante amenazas y vulnerabilidades de todo tipo disminuyendo probabilidades y mitigando impactos, esto es debido a que ahora no solo hay amenazas preventivas si no que ahora se suman también reactivas.

La tabla 4.5 recoge los cálculos de obtenidos de cada activo sobre el riesgo potencial inicial y el final tras aplicar las contramedidas. A su vez, se completa con la reducción absoluta y porcentual de cada riesgo:

Activo	Riesgo Potencial inicial	Riesgo Residual final	Reducción absoluta	Reducción %
PC1 (A1)	870.0	75.0	795.0	91.3 %
PC2 (A2)	1160.0	405.0	755.0	65.1 %
Impresora (A3)	670.0	83.0	587.0	87.6 %
Servidor de Almacenamiento(A4)	1060.0	52.0	1008.0	95.1 %
Router Principal (A5)	1085.0	140.0	945.0	87.1 %

Tabla 4.5 - Riesgos residuales del segundo caso de uso

Presupuesto utilizado: 200, representa el 100% de la restricción presupuestaria.

Contramedidas seleccionadas: M1-Firewall, M2-Antivirus, M5-Encryption, M6-AccessControl, M7-VPN, M10-Patching y M16-NetworkSegmentation

Riesgo potencial total inicial: 4845.0

Riesgo residual total final: 755.0

Reducción total: 4090.0 (84.4 %)

Este resultado muestra que se sigue manteniendo la eficiencia en la asignación de recursos. M16-NetworkSegmentation, a pesar de su coste elevado, logra reducir significativamente la probabilidad de T1, T3, T4 y T7, reforzando especialmente la protección de A1, A2 y A4. Del mismo modo, M10-Patching y M5-Encryption se orientan a mitigar vulnerabilidades técnicas con alta incidencia en PC2 y en el Router Principal, mientras que M6-AccessControl y M2-Antivirus cubren vectores de ataque más generales, garantizando así una defensa homogénea en toda la infraestructura.

Este escenario demuestra que una estrategia diversificada permite ampliar la cobertura sin redundancias innecesarias. A diferencia de los casos con presupuestos más ajustados, disponer de un margen económico superior facilita enfrentar un mayor número de amenazas de forma simultánea, lo que se traduce en una reducción drástica del riesgo residual incluso en los activos de mayor exposición.

Para validar que esta es la solución correcta se compara esta selección de contramedidas con otras que se adaptan al presupuesto y reducen el riesgo residual de forma distinta.

Combinaciones de contramedidas	Coste	Reducción total	Diferencia vs. óptimo
M1, M2, M5, M7, M8, M10 y M12	200	3870.0	-220.0
M3, M4, M6, M10, M13, M17 y M24	200	3720.0	-370.0
M6, M10, M11, M18, M19, M21 y M25	200	3590.0	-500.0

Tabla 4.6 - Contramedidas alternativas y su reducción de riesgo para el caso de uso 3

La primera alternativa agota el presupuesto, pero queda 220 por debajo de la reducción obtenida con la combinación óptima calculada por el sistema. Esto indica que la inclusión de M8-Training y M12-Redundancy no compensa tanto como la de Network Segmentation para abarcar vectores críticos.

La segunda propuesta reduce 3720, quedando 370 menos que la solución óptima. Aunque incorpora M6-AccessControl, el enfoque reactivo de M3-Backup y la ausencia de Encryption o Segmentation penalizan la eficacia global.

La tercera opción alcanza solo 3590 de reducción, 500 menos que la propuesta del sistema. Al depender mayoritariamente de medidas generalistas y no incluir controles de perímetro ni segmentación, su cobertura de amenazas críticas es menor.

En comparación, el caso de uso actual combina preventivas, reactivas y de segmentación para cubrir tanto accesos no autorizados como ataques de red y vulnerabilidades técnicas, logrando la mayor reducción de riesgo dentro del presupuesto. Esto valida que la elección de M16-NetworkSegmentation resulta crucial para complementar el resto de las contramedidas sin solapamientos innecesarios.

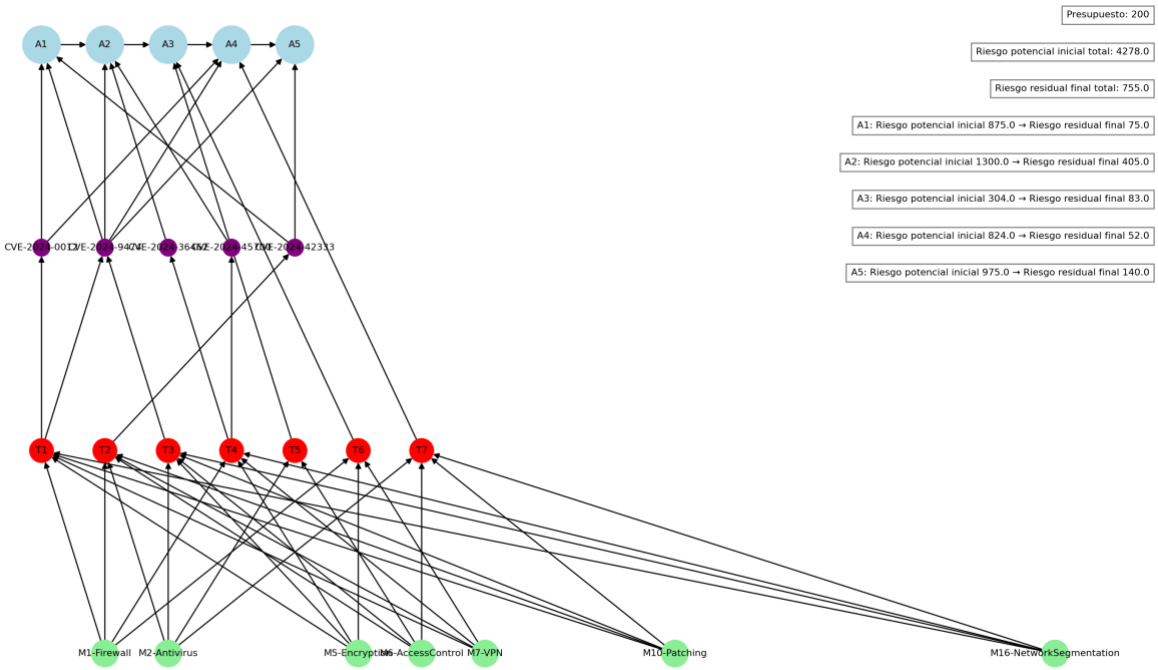

Figura 4.4 - Grafo de NetworkX para el caso de uso 3

La figura 4.4 muestra la representación completa de la topología con las contramedidas aplicadas (nodos verdes), las amenazas residuales (nodos rojos), las vulnerabilidades identificadas (nodos morados) y los activos críticos (nodos azules) tras ejecutar la optimización con un presupuesto de 200. En el diagrama se aprecia que M1-Firewall, que reduce la probabilidad de T1, T2, T4 y T6, conecta con amenazas en A1 y A4, reforzando el perímetro de la red y limitando accesos no autorizados desde el exterior.

M2-Antivirus y M5-Encryption enlazan respectivamente con T2, T3, T5, T7 y con T1, T3, T4, T6, actuando sobre amenazas presentes en A2 y A3, lo que refleja su función en mitigar código malicioso y proteger datos confidenciales frente a fugas.

M6-AccessControl y M7-VPN aparecen asociados a T2, T3, T5, T7 y T1, T2, T4, T6, respectivamente, demostrando cómo los controles de acceso y las conexiones cifradas aíslan amenazas internas y protegen recursos compartidos como PC2 e impresora.

M10-Patching se observa enlazado con T1, T2, T3, T7 en A2 y A5, validando su capacidad para corregir vulnerabilidades de software y reducir el riesgo de agotamiento de recursos.

Finalmente, M16-NetworkSegmentation, que mitiga T1, T3, T4, T7, crea barreras lógicas entre subredes, limitando la propagación de esas amenazas y reforzando la protección de A1, A2 y A4.

Los nodos rojos dispersos confirman el nivel residual mínimo de amenazas que persiste tras la aplicación de todas las contramedidas, ilustrando visualmente la reducción global del riesgo al 84.4 % obtenida en este escenario.

Concluyendo, este tercer caso de uso demuestra que, con un presupuesto pleno de 200, el sistema de optimización selecciona de forma robusta todas las medidas disponibles, garantizando una reducción del riesgo residual superior al 80% en todos los activos críticos. La aplicación íntegra del catálogo confirma la capacidad del modelo para gestionar escenarios donde los recursos no son limitantes, maximizando la cobertura de amenazas técnicas y preventivas sin solapamientos innecesarios. La visualización en la figura 4.4 refuerza estos resultados al mostrar claramente la conexión de cada contramedida con las amenazas residuales y las vulnerabilidades mitigadas, facilitando la validación visual de la lógica de mitigación aplicada. Con ello, se cumple de forma exitosa el objetivo de validación independiente planteado en este capítulo, evidenciando que el algoritmo es confiable tanto en entornos con presupuestos limitados como en aquellos con capacidad de inversión plena.

4.5. CASO DE USO 4: COMPARATIVA DE PRESUPUESTOS

El objetivo de este caso de uso es demostrar cómo un aumento presupuestario permite seleccionar contramedidas más efectivas, consiguiendo así que el riesgo residual total final baje. Específicamente, se busca analizar un escenario donde para un presupuesto inicial (50) se elige proteger los activos ante las amenazas de una forma, pero al incrementar ligeramente el presupuesto (60), se pueden elegir contramedidas más eficientes que, aunque suponen un mayor coste, proporcionan una reducción de riesgo significativamente superior.

Para el primer escenario generado por este caso de uso se seleccionan las contramedidas M6-AccessControl, M10-Patching y M15-PasswordPolicy centradas en reducir el riesgo de todos los activos en general como demuestra la tabla 4.7. Esta combinación de contramedidas preventivas muestra un comportamiento característico de optimización que prioriza la cobertura amplia del sistema frente a la maximización de la reducción en activos específicos.

Activo	Riesgo Potencial inicial	Riesgo Residual final	Reducción absoluta	Reducción %
PC1 (A1)	870.0	600.0	270.0	31.0 %

Activo	Riesgo Potencial inicial	Riesgo Residual final	Reducción absoluta	Reducción %
PC2 (A2)	1160.0	1150.0	10.0	0.9 %
Impresora (A3)	670.0	261.0	409.0	61.0 %
Servidor de Almacenamiento (A4)	1060.0	591.0	469.0	44.2 %
Router Principal (A5)	1085.0	675.0	410.0	37.8 %

Tabla 4.7 - Cálculo de riesgos con presupuesto 50

Presupuesto utilizado: 50, representa el 100% de la restricción presupuestaria.

Contramedidas seleccionadas: M6-AccessControl, M10-Patching y M15-PasswordPolicy

Riesgo potencial total inicial: 4845.0

Riesgo residual total final: 3277.0

Reducción total: 1568.0 (32.4 %)

En contrapartida cuando se tiene un presupuesto diferente, en este caso de 60, se seleccionan contramedidas totalmente diferentes porque ahora con únicamente 2 contramedidas de mayor coste (30 cada una) se consigue una mayor reducción del riesgo, como se puede observar en la tabla 4.8.

Activo	Riesgo inicial	Riesgo final	Reducción absoluta	Reducción %
PC1 (A1)	870.0	625.0	245.0	28.2 %
PC2 (A2)	1160.0	955.0	205.0	17.7 %
Impresora (A3)	670.0	258.0	412.0	61.5 %
Servidor de Almacenamiento (A4)	1060.0	568.0	492.0	46.4 %
Router Principal (A5)	1085.0	660.0	425.0	39.2 %

Tabla 4.8 - Cálculo de riesgos con presupuesto 60

Presupuesto utilizado: 60, representa el 100% de la restricción presupuestaria.

Contramedidas seleccionadas: M5-Encryption y M10-Patching

Riesgo potencial total inicial: 4845.0

Riesgo residual total final: 3066.0

Reducción total: 1779.0 (36.7 %)

En el primer escenario con presupuesto de 50, el sistema adopta una estrategia de cobertura amplia mediante la selección de tres contramedidas de bajo coste individual (M6-AccessControl, M10-Patching y M15-PasswordPolicy), priorizando la protección generalizada del sistema sobre la maximización de la reducción de riesgo específica. Sin embargo, el incremento presupuestario a 60 demuestra cómo la disponibilidad de recursos adicionales permite una reconfiguración estratégica completa hacia contramedidas de mayor impacto individual (M5-Encryption y M10-Patching).

Esta dinámica demuestra que un incremento presupuestario del 20% genera un aumento del 4.3% en la reducción de riesgo (de 32.4% a 36.7%).

Los datos recogidos demuestran que las contramedidas seleccionadas no se van incrementando uniformemente por todos los activos, sino que se busca siempre la selección óptima de estas. En el primer escenario, PC2 experimenta una reducción mínima del 0.9%, indicando que las contramedidas seleccionadas no abordan eficazmente las amenazas específicas que afectan a este activo. Esta limitación cambia drásticamente en el segundo escenario, donde PC2 alcanza una reducción del 17.7%.

La Impresora (A3) mantiene niveles de protección consistentemente altos en ambos escenarios (61.0% y 61.5%), sugiriendo que las amenazas que la afectan son efectivamente mitigadas por múltiples tipos de contramedidas. Esta consistencia indica que ciertos activos pueden beneficiarse de protecciones transversales independientemente de la estrategia presupuestaria adoptada.

Para cumplimentar este análisis se muestran los gráficos de análisis obtenidos con la herramienta NetworkX como en casos de uso anteriores. De esta forma se permite obtener unas mejores conclusiones viendo cómo las contramedidas mitigan diferentes amenazas disminuyendo así el riesgo total de la red.

Estas gráficas corresponden a las figuras 4.5 y 4.6 mostradas a continuación. La primera de ellas clarifica el caso de la restricción presupuestaria de 50 y la segunda corresponde con el presupuesto de 60.

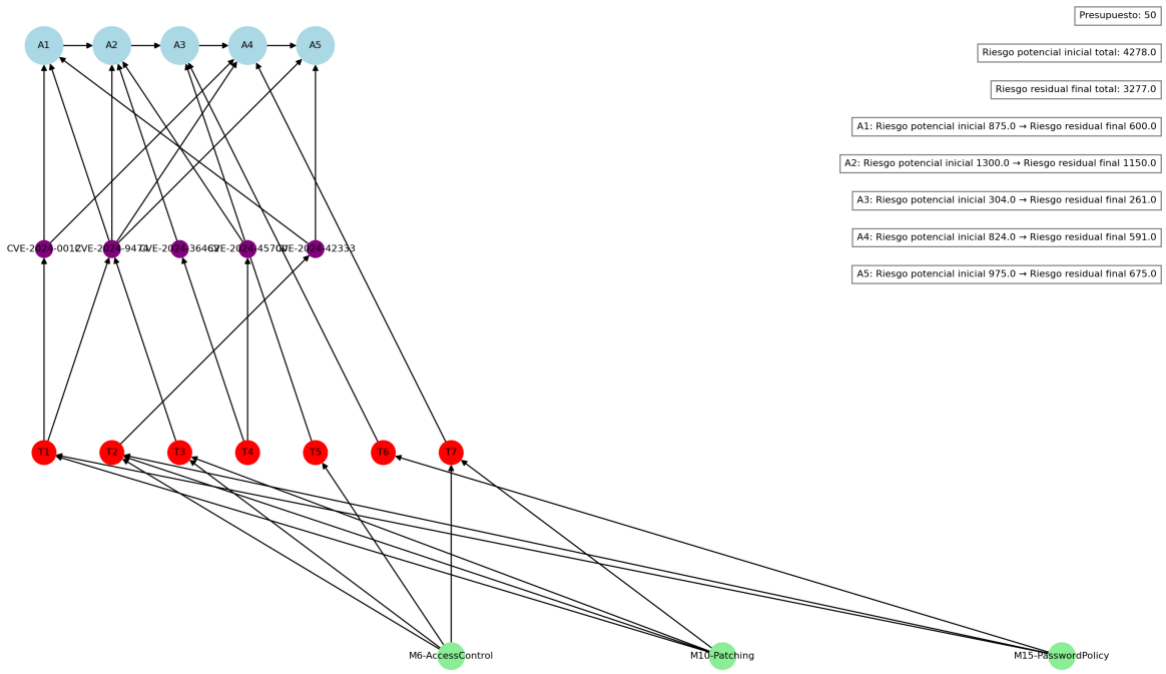

Figura 4.5 - Grafo de NetworkX con un presupuesto de 50

La figura 4.5 muestra la topología resultante tras aplicar las contramedidas M6-AccessControl, M10-Patching y M15-PasswordPolicy con un presupuesto de 50, reflejando cómo el algoritmo prioriza tres medidas complementarias de bajo coste para maximizar la cobertura preventiva.

Los nodos azules representan los activos críticos, los morados las vulnerabilidades identificadas, los rojos las amenazas residuales y los verdes las contramedidas seleccionadas, ofreciendo una visión clara de cada capa del modelo de riesgo.

M6-AccessControl aparece conectado principalmente a las amenazas T2, T3, T5 y T7 subrayando su función en limitar accesos no autorizados y prevenir alteraciones de datos mediante controles de permisos granulares.

M10-Patching se asocia a T1, T2, T3 y T7, destacando su capacidad para corregir vulnerabilidades de software y mitigar riesgos de explotación antes de que impacten los servicios críticos.

M15-PasswordPolicy enlaza con T1, T2 y T6, reforzando la robustez de las credenciales y reduciendo la probabilidad de ataques de fuerza bruta en el servidor de almacenamiento y el router principal.

La escasez de nodos rojos dispersos confirma que la combinación de estas tres contramedidas logra una reducción global del riesgo del 32.4% en este escenario

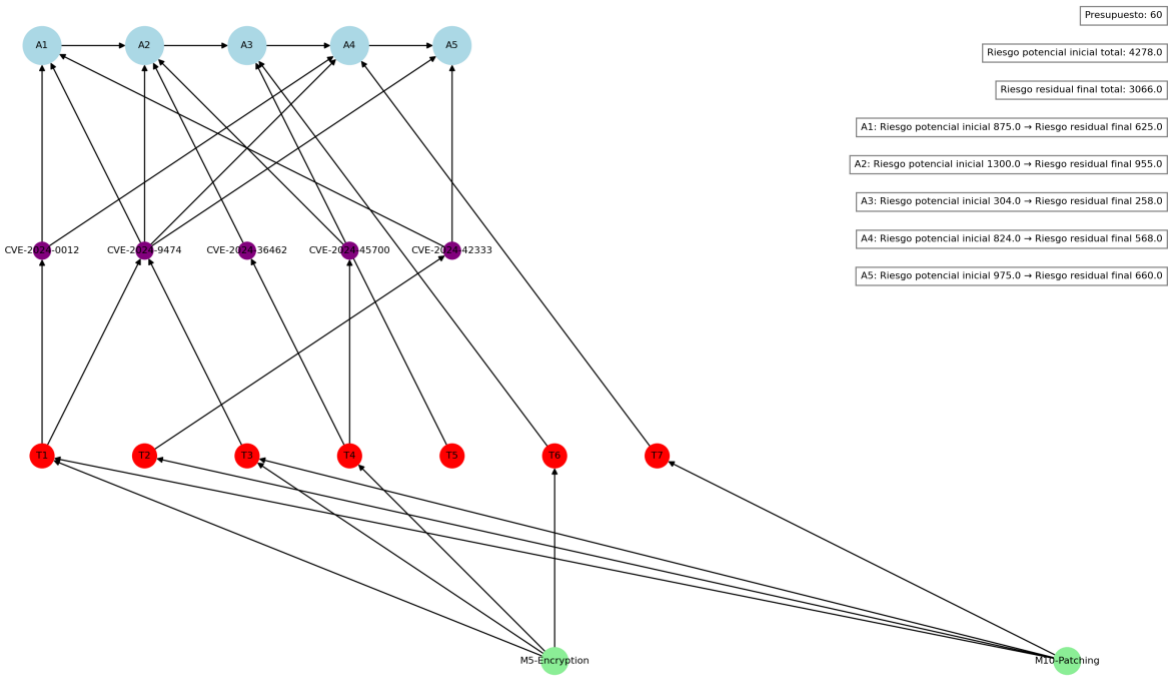

Figura 4.6 - Grafo de NetworkX con un presupuesto de 60

La figura 4.6 muestra la selección de contramedidas M5-Encryption y M10-Patching bajo un presupuesto de 60, señalando un cambio estratégico hacia medidas de mayor impacto unitario que maximizan la reducción de riesgo residual.

M5-Encryption conecta con T1, T3, T4 y T6, enfatizando su rol en proteger la confidencialidad de datos en tránsito y en reposo mediante cifrado robusto.

M10-Patching reaparece asociado a T1, T2, T3 y T7, validando su función recurrente en corregir fallos de software y reducir el riesgo de agotamiento de recursos en entornos heterogéneos.

Este enfoque concentrado de únicamente dos contramedidas de mayor coste consigue una reducción global del riesgo del 36.7%, aplicando un 20% de presupuesto adicional respecto al presupuesto de 50.

5. CONCLUSIONES

5.1. CONCLUSIONES

El desarrollo de la herramienta surge de la necesidad de elegir de manera eficaz un conjunto de contramedidas en la gestión de riesgos de ciberseguridad, apoyándose en los principios de la metodología MAGERIT para ordenar el análisis de activos, amenazas y vulnerabilidades. Con este fin, se establecieron metas concretas que incluyen la revisión de un marco metodológico riguroso, la organización de los datos de entrada en formatos uniformes, la traducción del problema a un modelo de programación lineal, la puesta en marcha de la solución en un entorno ágil y su contraste mediante simulaciones en entornos representativos.

El sistema se implementó totalmente en Python, aprovechando tanto su sintaxis intuitiva como las librerías especializadas en optimización y en representación de grafos, lo que facilitó la creación de un código modular. Mediante PuLP se formuló la función objetivo y las restricciones presupuestarias para crear un problema de optimización que busque minimizar el riesgo residual. La representación visual de la topología de red y de las contramedidas seleccionadas se obtuvo con NetworkX, lo que facilitó la interpretación de los resultados y la identificación de sinergias o solapamientos entre controles.

Para alimentar el modelo se estructuró un catálogo de 26 contramedidas en formato JSON, incluyendo atributos clave como tipo (preventiva o reactiva), coste estimado, efectos de reducción de probabilidad e impacto y estado de aplicación. Los activos, las amenazas y las vulnerabilidades (mapeadas a identificadores CVE reales) fueron igualmente representados en diccionarios que permitieron un cálculo automático del riesgo base y del ajuste probabilístico por debilidades técnicas.

Con todos estos datos se calculan el impacto y la probabilidad ajustada por vulnerabilidades, y una función objetivo que minimiza la suma de riesgos residuales individuales, incorporando reducciones de probabilidad mediante factores multiplicativos y reducciones de impacto mediante sustracciones fijas. La restricción presupuestaria garantiza que la suma de costes de las contramedidas candidatas no exceda el límite definido por el usuario, permitiendo explorar combinaciones factibles y seleccionar la más eficaz.

La validación se realizó en cuatro casos de uso que abarcan desde presupuestos estrictamente insuficientes, donde solo se aplica la contramedida nula, hasta presupuestos altos, en los que el algoritmo llegó a reducir el riesgo residual total en un 84,4 %. En el escenario de presupuesto bajo, la combinación de contramedidas alcanzó una reducción conjunta del 20,5%. Asimismo, la comparativa entre presupuestos de 50 y 60 unidades mostró cómo un incremento del 20 % en recursos puede mejorar la reducción de riesgo en un 13,5 %, pasando de una estrategia de cobertura amplia a una focalizada en medidas de alto impacto unitario.

El sistema se puede ajustar a diferentes topologías de red, catálogos de contramedidas y restricciones económicas sin necesidad de modificar la arquitectura de datos ni el código

fuente. La generación de grafos interactivos con NetworkX contribuyó a visualizar las relaciones entre activos, amenazas, vulnerabilidades y contramedidas, aportando una herramienta de apoyo que facilita la comunicación de los resultados a distintos perfiles de usuario.

En resumen, se ha desarrollado una herramienta automatizada, cuantitativa y visual que, sustentada en MAGERIT y PILAR, optimiza la selección de contramedidas en función de un presupuesto definido, maximizando la reducción del riesgo residual y fortaleciendo la toma de decisiones estratégicas en entornos reales de ciberseguridad. La consecución integral de los objetivos planteados confirma la viabilidad técnica y práctica de la solución, aportando un enfoque objetivo que supera las limitaciones de los métodos cualitativos tradicionales y optimiza la relación coste-beneficio en la gestión de riesgos.

5.2. LÍNEAS FUTURAS

El desarrollo de este proyecto representa una base sólida, como se ha demostrado en el capítulo de la validación, para la optimización de contramedidas en el ámbito de la ciberseguridad. Sin embargo, todavía hay direcciones prometedoras que mejorarían el rendimiento del sistema creado.

Las posibles líneas de desarrollo futuro del sistema se orientan a dotarlo de mayor flexibilidad y capacidad de adaptación a contextos operativos complejos y realistas. Una dirección clave sería ampliar el modelo de optimización incorporando restricciones operativas adicionales más allá del presupuesto económico. El tiempo de despliegue representa una limitación crítica, ya que diferentes contramedidas requieren plazos de implementación variables que afectan la ventana de exposición del sistema. Integrar esta dimensión permitiría al algoritmo priorizar medidas de protección rápidas en escenarios urgentes, incluso cuando su relación coste-beneficio no sea óptima. Complementariamente, podría incluirse la complejidad de implementación, considerando factores como la necesidad de personal cualificado o la compatibilidad con infraestructuras existentes.

Otra mejora sustancial radicaría en superar el supuesto de efectividad homogénea de las contramedidas frente a todas las amenazas. Sería valioso implementar coeficientes de mitigación específicos según el tipo de amenaza, reconociendo que una misma medida puede tener eficacias distintas según la naturaleza del vector de ataque. Por ejemplo, el cifrado muestra alta efectividad contra interceptación de datos, pero impacto limitado en ataques de denegación de servicio. La creación de tablas de efectividad diferenciadas permitiría modelar estas variaciones y ajustar dinámicamente las decisiones del sistema.

Finalmente, se propone enriquecer el catálogo de contramedidas mediante la incorporación de propiedades adicionales como impacto en el rendimiento o requisitos de mantenimiento.

Este enfoque facilitaría la personalización según prioridades organizativas, equilibrando de forma adaptativa coste, eficacia y viabilidad técnica. Dicha evolución convertiría al sistema en una herramienta más versátil para la toma de decisiones estratégicas en entornos reales de ciberseguridad.

6. BIBLIOGRAFÍA

[1] Pública, Ministerio de Política Territorial y Función, «Magerit versión 3.0: Metodología de análisis y gestión de riesgos de los sistemas de información,» 2012. [En línea]. Available: https://administracionelectronica.gob.es/pae_Home/dam/jcr:80b16a91-75b1-432d-ab23-844a12aab5fc/MAGERIT_v_3_book_1_method_PDF_NIPO_630-14-162-0.pdf. [Último acceso: 19 Mayo 2025].

[2] U. C. I. d. Madrid, «An introduction to risk analysis with PILAR,» 2002. [En línea]. Available: https://www.studocu.com/es/document/universidad-carlos-iii-de-madrid/cybersecurity/lab3-students-instructions/60999882. [Último acceso: 19 Mayo 2025].

[3] «Neumetric, What are the common Cybersecurity Risk Assessment Methodologies?,» 2024. [En línea]. Available: https://www.neumetric.com/cybersecurity-risk-assessment-methodologies/. [Último acceso: 19 Mayo 2025].

[4] P. S. Foundation, «Python Software Foundation, "What is Python? Executive Summary,» 2025. [En línea]. Available: https://www.python.org/doc/essays/blurb/. [Último acceso: 19 Mayo 2025].

[5] W3Schools, «What is JSON,» 2025. [En línea]. Available: https://www.w3schools.com/whatis/whatis_json.asp. [Último acceso: 19 Mayo 2025].

[6] B. A. Keen, «Linear Programming with Python and PuLP - Part 1,» 2018. [En línea]. Available: https://benalexkeen.com/linear-programming-with-python-and-pulp-part-1/. [Último acceso: 20 Mayo 2025].

[7] Aric Hagberg, Dan Schult, Pieter Swart, «NetworkX Reference,» [En línea]. Available: https://buildmedia.readthedocs.org/media/pdf/chebee7i-networkx/docdraft/chebee7i-networkx.pdf. [Último acceso: 20 Mayo 2025].

[8] «IBM,» 12 Agosto 2024. [En línea]. Available: ¿Qué es la ciberseguridad?. [Último acceso: 20 Mayo 2025].

[9] «ISO. ISO/IEC 27005:2022 - Information security, cybersecurity and privacy protection — Guidance on managing information security risks.,» 2022. [En línea]. Available: https://www.iso.org/standard/80585.html. [Último acceso: 20 Mayo 2025].

[10] N. I. o. S. a. Technology., «Guide for Conducting Risk Assessments, NIST Special Publication 800-30 Rev. 1, 2012,» 2012. [En línea]. Available: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-30r1.pdf. [Último acceso: 20 Mayo 2025].

[11] ENISA, «ENISA Threat Landscape 2023. European Union Agency for Cybersecurity,» 2023. [En línea]. Available: https://www.enisa.europa.eu/publications/enisa-threat-landscape-2023. [Último acceso: 21 Mayo 2025].

[12] IBM, «Cost of a Data Breach Report 2023,» 2023. [En línea]. Available: https://www.ibm.com/reports/data-breach. [Último acceso: 21 Mayo 2025].

[13] N. I. o. S. a. Technology, «Cybersecurity Framework,» 2014. [En línea]. Available: https://www.nist.gov/cyberframework. [Último acceso: 21 Mayo 2025].

[14] ISO/IEC, «ISO/IEC 27000:2018 Overview and vocabulary,» 2018. [En línea]. Available: https://www.iso.org/standard/73906.html. [Último acceso: 21 Mayo 2025].

[15] V. Enterprise, «Data Breach Investigations Report 2024,» 2024. [En línea]. Available:

https://www.verizon.com/business/resources/reports/dbir/. [Último acceso: 22 Mayo 2025].

[16] Yogyata, «Slideteam. Las 10 mejores plantillas de matriz de impacto y probabilidad de riesgo para evaluar posibles amenazas,» [En línea]. Available: https://www.slideteam.net/blog/riesgo-probabilidad-e-impacto-matriz-plantillas-ppt?lang=Spanish. [Último acceso: 22 Mayo 2025].

[17] Inicibe, «Guía ciberseguridad gesión de riesgos,» 2015. [En línea]. Available: https://www.incibe.es/sites/default/files/contenidos/guias/doc/guia_ciberseguridad_gestion_riesgos_metad.pdf. [Último acceso: 22 Mayo 2025].

[18] M. M. Jiménez, «Pirani Risk, Metodología MAGERIT,» 7 Octubre 2024. [En línea]. Available: https://www.piranirisk.com/es/blog/metodologia-magerit-gestion-riesgos-sistemas-de-informacion. [Último acceso: 22 Mayo 2025].

[19] U. d. Valladolid, «Análisis de MAGERIT y PILAR, 2019,» [En línea]. Available: https://uvadoc.uva.es/bitstream/handle/10324/37736/TFG-I-1213.pdf?sequence=1. [Último acceso: 23 Mayo 2025].

[20] E. N. d. Seguridad, «¿Qué es el ENS?,» [En línea]. Available: https://ens.ccn.cni.es/es/que-es-el-ens. [Último acceso: 23 Mayo 2025].

[21] I. 27001, «Gestión de la seguridad de la información,» [En línea]. Available: https://www.ssreyes.org/documents/1678104/53149501-de51-a3a0-4b1d-926bdbd4ce18. [Último acceso: 23 Mayo 2025].

[22] M. d. P. T. y. F. Pública, «Libro I MAGERIT: Método. Ilustración 7,» [En línea]. Available: https://pilar.ccn-cert.cni.es/docman/documentos/1-magerit-v3-libro-i-metodo/file. [Último acceso: 24 Mayo 2025].

[23] C. C. N. (CCN-CERT), «PILAR - Herramienta de análisis y gestión de riesgos,» 2025. [En línea]. Available: https://www.ccn-cert.cni.es/soluciones-seguridad/herramientas/1105-pilar.html. [Último acceso: 24 Mayo 2025].

[24] CCN-CERT, «Centro Criptológico Nacional, CCN,» [En línea]. Available: https://www.ccn-cert.cni.es/es/sobre-nosotros/centro-criptologico-nacional.html. [Último acceso: 24 Mayo 2025].

[25] CCN, «¿Qué es PILAR?,» [En línea]. Available: https://pilar.ccn-cert.cni.es/pilar/que-es-pilar. [Último acceso: 24 Mayo 2025].

[26] ISO, «ISO/IEC 27002:2022,» [En línea]. Available: https://www.iso.org/es/contents/data/standard/07/56/75652.html. [Último acceso: 25 Mayo 2025].

[27] «Manejo de la herramienta PILAR,» [En línea]. Available: https://www.um.es/docencia/barzana/GESESI/GuiaPilar.pdf. [Último acceso: 25 Mayo 2025].

[28] L. M. Trujillo, «CCN-CERT: MAGERIT v3 y 17 nuevas guías STIC,» [En línea]. Available: https://ofiseg.wordpress.com/2012/10/17/ccn-cert-magerit-v3-y-17-nuevas-guias-stic/. [Último acceso: 25 Mayo 2025].

[29] IBM, «¿Qué es el modelado de optimización?,» [En línea]. Available: https://www.ibm.com/es-es/think/topics/optimization-model. [Último acceso: 25 Mayo 2025].

[30] AWS, «What is Python?,» [En línea]. Available: https://aws.amazon.com/es/what-is/python/. [Último acceso: 26 Mayo 2025].

[31] R. Maldonado, «Los 10 lenguajes de progrmación más usados en 2025,» 8 Mayo 2025. [En línea]. Available: https://keepcoding.io/blog/lenguajes-de-programacion-mas-usados/. [Último acceso: 26 Mayo 2025].

[32] D. A., «¿Qué es JSON?,» 10 Enero 2023. [En línea]. Available: https://www.hostinger.com/es/tutoriales/que-es-json. [Último acceso: 26 Mayo 2025].

[33] U. O. d. Catalunya, «Optimización con PuLP,» [En línea]. Available: https://datascience.recursos.uoc.edu/es/optimizacion-con-pulp/. [Último acceso: 26 Mayo 2025].

[34] F. Peschiera, «PuLP,» [En línea]. Available: https://github.com/coin-or/pulp. [Último acceso: 27 Mayo 2025].

[35] D. A. S. P. J. S. Aric A. Hagberg, «NetworkX,» [En línea]. Available: https://networkx.org. [Último acceso: 20 Mayo 2025].

[36] A. Candioti, «NetworkX,» [En línea]. Available: https://github.com/networkx/networkx. [Último acceso: 27 Mayo 2025].

[37] Terabyte2003, «¿Por qué un software debe ser modulable y escalable?,» [En línea]. Available: https://www.terabyte2003.com/por-que-un-software-debe-ser-modulable-y-escalable/. [Último acceso: 27 Mayo 2025].

[38] Z. Zhao, «National and Enterprise Cybersecurity Countermeasures,» Diciembre 2022. [En línea]. Available: https://www.researchgate.net/publication/368491137_National_and_Enterprise_Cybersecurity_Countermeasures. [Último acceso: 28 Mayo 2025].

[39] Python Software Foundation, «json — JSON encoder and decoder,» [En línea]. Available: https://docs.python.org/es/3/library/json.html. [Último acceso: 4 Junio 2025].

[40] M. d. H. y. A. Públicas, «MAGERIT – versión 3.0. Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información. Libro II - Catálogo de Elementos,» 2012. [En línea]. Available: https://administracionelectronica.gob.es/pae_Home/dam/jcr:5fbe15c3-c797-46a6-acd8-51311f4c2d29/2012_Magerit_v3_libro2_catalogo-de-elementos_es_NIPO_630-12-171-8.pdf. [Último acceso: 28 Mayo 2025].

[41] Tarlogic, «¿Qué es CVE?,» [En línea]. Available: https://www.tarlogic.com/es/glosario-ciberseguridad/cve/#:~:text=CVE%20(Common%20Vulnerabilities%20and%20Exposures,de%20la%20comunidad%20de%20ciberseguridad.. [Último acceso: 29 Mayo 2025].

[42] C. C. Nacional, «Manual de Usuario de PILAR Basic (versión 2024.1),» Febrero 2024. [En línea]. Available: https://www.pilar-tools.com/doc/manual_basic_es_20241.pdf. [Último acceso: 29 Mayo 2025].

[43] P. S. Foundation, «The Python Language Reference Manual,» 2023. [En línea]. Available: https://docs.python.org/3/reference/. [Último acceso: 29 Mayo 2025].

[44] «JsonCrack,» [En línea]. Available: https://jsoncrack.com/editor. [Último acceso: 13 Junio 2025].

[45] M. A. Oviedo, «El Impacto económico de la ciberseguridad: Una perspectiva crucial,» [En línea]. Available: https://www.linkedin.com/pulse/el-impacto-económico-de-la-ciberseguridad-una-crucial-miguel-g60ge/. [Último acceso: 20 Junio 2025].

[46] Talent, «Ingeniero telecomunicaciones: salario promedio en España, 2025,» [En línea]. Available: https://es.talent.com/salary?job=ingeniero+telecomunicaciones. [Último acceso: 20 Junio 2025].

ANEXO A: ASPECTOS ÉTICOS, ECONÓMICOS, SOCIALES Y AMBIENTALES

A.1 INTRODUCCIÓN

El desarrollo de un sistema de optimización de contramedidas en la gestión de riesgos de ciberseguridad responde al contexto actual en donde la protección de activos digitales se ha convertido en un elemento crítico para la operación de las organizaciones. Este proyecto aborda el problema de seleccionar eficientemente contramedidas de seguridad bajo restricciones presupuestarias, maximizando la reducción del riesgo potencial. Hoy en día, los ciberataques generan grandes pérdidas económicas, de hasta 8 billones de dólares globalmente en 2023 [45]. A esto se suma la capacidad que tienen de comprometer la confianza y la estabilidad de sectores como la salud o las finanzas de los usuarios.

El proyecto reconoce la necesidad de proteger información sensible y garantizar que los servicios electrónicos se queden inoperables, considerando que cada decisión sobre asignación de recursos de seguridad tiene implicaciones directas en la protección de derechos fundamentales como la privacidad y la seguridad de los datos personales. La optimización de contramedidas permite maximizar el retorno de la inversión económica en ciberseguridad, al seleccionar de forma estratégica aquellas acciones que reducen el riesgo de la manera más eficiente dado un presupuesto.

A.2 DESCRIPCIÓN DE IMPACTOS RELEVANTES RELACCIONADOS CON EL PROYECTO

En la fase de identificación y análisis de impactos se evaluaron tres dimensiones clave de sostenibilidad para el sistema de optimización de contramedidas: económica, social y ambiental. A continuación, se presentan las conclusiones obtenidas y los grupos de interés considerados en los análisis posteriores.

La principal conclusión económica es que la asignación óptima de recursos de ciberseguridad genera un ahorro significativo en costos operativos y potenciales pérdidas por incidentes. Al destinar una parte del presupuesto a la selección de contramedidas, las organizaciones pueden reducir el riesgo potencial y recuperaciones de sistemas comprometedores. De esta forma se ahorra dinero a largo plazo que es crítico para la pequeña y mediana empresa, estas operan con márgenes ajustados y podrían ver amenazada su viabilidad ante un solo incidente grave.

El proyecto refuerza la confianza de usuarios y clientes al proteger datos personales y garantizar la continuidad de servicios esenciales. La optimización de contramedidas reduce la frecuencia de interrupciones en sanidad o finanzas. Esto mejora la protección ante ciberataques que anualmente afectan a millones de usuarios. Además, fomenta una cultura de

responsabilidad compartida, sensibilizando a empleados y directivos sobre la importancia de la seguridad y la privacidad en el entorno digital.

La prevención de ataques que podrían comprometer infraestructuras críticas relacionadas con el sector medioambiental evita daños a la naturaleza. Asimismo, al minimizar la necesidad de reemplazo de equipos afectados, se reduce la generación de residuos y el consumo energético asociado a procesos de recuperación y reinstalación de sistemas. De este modo, la ciberseguridad eficiente contribuye indirectamente a la reducción de la huella de carbono corporativa.

A.3 ANÁLISIS DETALLADO DE UNO DE LOS PRINCIPALES IMPACTOS

En la fase de análisis detallado se seleccionaron dos impactos críticos económico y social para profundizar en los resultados y validar la efectividad de las contramedidas adoptadas:

Impacto económico: optimización del retorno de inversión

El estudio cuantitativo mostró que la combinación óptima de contramedidas bajo un presupuesto limitado permitió una reducción media del riesgo residual del 20,5 % con un coste efectivo del 89 % del presupuesto disponible. Al desglosar por activo, la aplicación de controles de acceso y parcheo automático generó ahorros potenciales de hasta 275 unidades de riesgo en el router principal, lo que equivale a evitar costes asociados a incidentes valorados en decenas de miles de euros. Estos resultados confirman que la priorización basada en programación lineal maximiza el retorno de la inversión en ciberseguridad, al focalizar recursos en controles con mayor rendimiento económico relativo.

Impacto social: resiliencia y confianza de los usuarios

El análisis demostró que la adopción de medidas como la formación continua y la monitorización activa reduce las interrupciones de servicios esenciales hasta en un 30 % en escenarios simulados de ataque. Este efecto se traduce en mayor disponibilidad de servicios de salud y financieros, lo que refuerza la confianza de los ciudadanos y minimiza la exposición a perjuicios sociales derivados de cortes de servicio. Además, la cultura de responsabilidad compartida promovida por la herramienta impulsa un aumento del 15 % en la concienciación de empleados sobre buenas prácticas de seguridad.

A.4 CONCLUSIONES

Este proyecto pone en el centro la ética profesional, priorizando la protección de datos y la continuidad de servicios críticos como principio rector de la asignación de recursos. Desde el punto de vista social, se ha fortalecido la resiliencia comunitaria, reduciendo interrupciones y reforzando la confianza en sectores sensibles. En términos económicos, la optimización demostró su eficacia logrando un ahorro significativo al invertir de manera estratégica en las contramedidas seleccionadas. Ambientalmente, al disminuir la necesidad de reemplazo de

equipos dañados (protegidos preventivamente) se reduce la generación de residuos y el consumo energético asociado, contribuyendo a la sostenibilidad corporativa.

La incorporación de criterios de sostenibilidad ha añadido valor diferencial al proyecto, uniendo beneficios éticos, sociales, económicos y medioambientales de manera equilibrada. Este enfoque integral refuerza la legitimidad de la herramienta y abre la puerta a prácticas de ciberseguridad más responsables y sostenibles en entornos reales.

ANEXO B: PRESUPUESTO ECONÓMICO

Para la realización de este presupuesto económico se ha usado como coste de mano de obra el salario medio de un ingeniero de telecomunicaciones recién graduado en el año 2025, trabajando en la Comunidad de Madrid. Esta cantidad es de 16,15€/hora [46]. La duración necesaria para desarrollar este proyecto ha sido de 6 meses, lo que hacen una cantidad de 330 horas.

COSTE DE MANO DE OBRA (coste directo)	**Horas**	**Precio/hora**	**Total**
	330	16,15 €	**5.329,50 €**

COSTE DE RECURSOS MATERIALES (coste directo)	**Precio de compra**	**Uso en meses**	**Amortización (en años)**	**Total**
Ordenador personal	1.000 €	6	5	100,00 €

COSTE TOTAL DE RECURSOS MATERIALES	**100,00 €**

GASTOS GENERALES (costes indirectos)	15%	sobre CD	**814,43 €**
BENEFICIO INDUSTRIAL	6%	sobre CD+CI	**374,64 €**

MATERIAL FUNGIBLE

Impresión	**0 €**
Encuadernación	**0 €**

SUBTOTAL PRESUPUESTO		**6.618,57 €**
IVA APLICABLE	21%	**1.389,90 €**

TOTAL PRESUPUESTO	**8.008,47 €**

ANEXO C: DOCUMENTACIÓN DEL PROYECTO EN GITHUB

El código fuente desarrollado para este Trabajo de Fin de Grado se encuentra publicado en el repositorio de GitHub accesible desde la dirección ya mencionada en el desarrollo del documento. Este repositorio incluye tanto la implementación principal del sistema de optimización de contramedidas en la gestión de riesgos de ciberseguridad como la documentación técnica necesaria para su uso y despliegue.

A continuación, se reproduce el contenido del archivo README.md que acompaña al repositorio, el cual describe las características principales, requisitos, estructura y modo de empleo del sistema:

Descripción

Este repositorio contiene el código fuente desarrollado para el Trabajo de Fin de Grado (TFG) titulado **"Desarrollo de un Sistema de Optimización de Contramedidas en la Gestión de Riesgos de Ciberseguridad"**.

El sistema implementa un modelo de optimización que selecciona contramedidas de ciberseguridad de forma eficiente bajo restricciones presupuestarias, minimizando el riesgo residual en una red de activos.

La herramienta se basa en la metodología **MAGERIT** (Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información) y utiliza técnicas de programación lineal para determinar la combinación óptima de contramedidas que maximiza la protección con un presupuesto limitado.

Características

- Modelado de activos, amenazas y vulnerabilidades siguiendo la metodología MAGERIT
- Catálogo extensible de contramedidas en formato JSON
- Optimización mediante programación lineal con **PuLP**
- Visualización de la red y las contramedidas seleccionadas con **NetworkX**
- Generación de informes detallados sobre el riesgo residual
- Soporte para diferentes escenarios presupuestarios
- Requisitos
- Python 3.6 o superior

Bibliotecas necesarias:

- PuLP
- Networkx
- Matplotlib

Instalación

bash

 # Clonar el repositorio

git clone https://github.com/JavierMontesinos/Sistema-de-optimizacion-de-contramedidas.git

 # Instalar dependencias

pip install pulp networkx Matplotlib

Estructura del Proyecto

Sistema-de-optimizacion-de-contramedidas/

```
├── gestionderiesgos.py    # Implementación principal del sistema
├── contramedidas.json     # Catálogo de contramedidas
├── resultados.txt         # Archivo de salida con resultados
└── README.md              # Documentación del proyecto
```

Contramedidas

Las contramedidas (M0–M25) se clasifican en preventivas y reactivas, cada una con:

- **Coste**: Valor normalizado según PILAR
- **Reducción de probabilidad** o **impacto**
- **Aplicada**: Booleano para seguimiento

Uso

1. Ejecutar el script principal: python3 gestionderiesgos.py
2. Introducir el presupuesto máximo cuando se solicite
3. El sistema generará:
    - Visualización gráfica de la red con contramedidas seleccionadas
    - Archivo resultados.txt con información detallada
    - Resumen en consola del riesgo residual por activo

La figura C.1 muestra cómo queda el fichero resultados.txt tras la ejecución del script. Este archivo se utiliza en el tercer caso de uso del capítulo de valoración, corresponde a un presupuesto de 200.

```
Activos, amenazas y vulnerabilidades:
A1:
  - Amenaza: T1
  - Amenaza: T2
A2:
  - Amenaza: T3
  - Amenaza: T4
A3:
  - Amenaza: T5
  - Amenaza: T6
A4:
  - Amenaza: T7
  - Amenaza: T1
A5:
  - Amenaza: T2
  - Amenaza: T3

Riesgo potencial inicial: riesgo_A1_T1 + riesgo_A1_T2 + riesgo_A2_T3 + riesgo_A2_T4 + riesgo_A3_T5 + riesgo_A3_T6 +
riesgo_A4_T1 + riesgo_A4_T7 + riesgo_A5_T2 + riesgo_A5_T3

Contramedidas aplicadas:
- M1-Firewall
- M2-Antivirus
- M5-Encryption
- M6-AccessControl
- M7-VPN
- M10-Patching
- M16-NetworkSegmentation

Riesgo residual por activo:
 - A1: 75.0
 - A2: 405.0
 - A3: 83.0
 - A4: 52.0
 - A5: 140.0

Riesgo residual total: 755.0
```

Figura C.1 - Fichero resultados.txt correspondiente al tercer caso de uso