

UNIVERSIDAD POLITÉCNICA DE MADRID
Escuela Técnica Superior de Ingenieros de Telecomunicación



**Propuesta de un modelo de caracterización de
ciberataques para entornos de conciencia
cbersituacional**

TESIS DOCTORAL

Presentada para optar al título de Doctor por:

Carmen Sánchez Zas
Máster Universitario en Ingeniería de Telecomunicación

Madrid, 2024



UNIVERSIDAD POLITÉCNICA DE MADRID
Escuela Técnica Superior de Ingenieros de Telecomunicación

Doctorado en Ingeniería de Sistemas Telemáticos

Propuesta de un modelo de caracterización de ciberataques para entornos de conciencia cibersituacional

TESIS DOCTORAL

Presentada para optar al título de Doctor por:

Carmen Sánchez Zas

Máster Universitario en Ingeniería de Telecomunicación

Bajo la dirección de:

Dr. Víctor Abraham Villagrá González

Dr. Xavier Andrés Larriva Novo

Madrid, 2024

Título: Propuesta de un modelo de caracterización de ciberataques para entornos de conciencia cibersituacional

Autor: Carmen Sánchez Zas

Programa de Doctorado: Ingeniería de Sistemas Telemáticos

Dirección de Tesis:

Dr. Víctor Abraham Villagrá González, Profesor Titular Universitario, UPM (Director)

Dr. Xavier Andrés Larriva Novo, Profesor Ayudante Doctor, UPM (Director)

Tribunal de Tesis:

Fecha de Defensa de Tesis:

If you think technology can solve your security problems, then you don't understand the problems and you don't understand the technology.

Bruce Schneier

Agradecimientos

Quiero dedicar este espacio a dejar por escrito algo que he intentado hacer durante todo este proceso, pero especialmente, en las últimas semanas: reconocer y dar las gracias.

Creo firmemente en la necesidad de relacionarse con personas que nos inspiren a ser mejores y, por suerte, he podido contar con muchas, aunque algunas ya no estén.

Siempre hay que rodearse de personas que sean mejores y que sepan más que uno mismo. Por eso, quiero empezar dando las gracias a mis directores, Víctor y Xavi, por vuestro tiempo y consejos, por saber acompañarme hasta el final y por vuestra visión del potencial de esta investigación. Ha sido un camino duro pero muy bonito, estoy muy agradecida por haberlo empezado con vosotros y por todo lo que me habéis enseñado a lo largo de él.

Sin duda, también al resto del grupo de Redes y Servicios de Telecomunicación e Internet, por haber confiado en mí y darme la oportunidad de participar en proyectos que me han permitido descubrir y contribuir a este mundo tan interesante que es la ciberseguridad. Por ponerme las cosas lo más fácil que habéis podido, escucharme, animarme y, especialmente, por el sentimiento de grupo y compañerismo que se respira entre vosotros. El trabajo así se ve de otra manera. Sin todo ello, esta tesis no habría sido posible.

También creo que hay que tener a tu lado a quienes se emocionen más que tú cuando pasan cosas buenas pero que igualmente permanezcan ahí cuando se ponen difíciles, sea donde sea y estén donde estén. Es algo que he aprendido de mi familia, que son la base de todo lo que soy hoy. Quiero daros las gracias por tratar de ayudarme siempre, aunque tenga mi propio idioma, por la paciencia que tenéis y por animarme a dar lo mejor de mí misma y trabajar siempre para conseguir el próximo objetivo.

A mis amigos, que se han convertido más, si cabe, en un apoyo fundamental. Gracias por adaptarlos a mis tiempos, aceptar mis ausencias y despistes, por mostrar continuamente que estáis a mi lado y por darme ánimos de mil millones de maneras, algunas inimaginables. Por saber cuándo preguntar y cuándo dejar de hacerlo. Aunque a algunos os *sonara a chino* y otros hayáis podido darme una visión más cercana, cada uno habéis contribuido con vuestro granito de arena. Con vosotros al lado, todo es mucho mejor.

Dicen que uno no sabe lo que tiene hasta que lo pierde. Por suerte, yo no he tenido que hacerlo, me ha bastado con desarrollar esta Tesis Doctoral. Espero acordarme de ello siempre.

Por último, a todos los que en este periodo de tres años han tenido palabras de ánimo y han estado ahí, especialmente al final, esperándome. Esta Tesis Doctoral es un poquito de todos vosotros.

Infinitamente, gracias.

Abstract

Today, the constant development of artificial intelligence has a major impact on system security. New challenges are continuously arising and are being met by research currently focused on the concept of cyber situational awareness, which provides a global view of the security of a system.

Current cyber-attack trends show that are becoming more frequent and complex, using advanced techniques and zero-day attacks that make them considerably more difficult to detect. This raises the issue of the lack of knowledge about these cyber-attacks and their characterisation, making it possible to adapt responses to incidents and to consider the ability to automate them so that the associated risks can be mitigated in real time.

This is the main motivation behind the proposal of this Doctoral Thesis, a model for characterising cyber-attacks in cyber-situational awareness environments. To this end, first, an intrusion detection system based on heterogeneous data sources, such as physical and logical communication sensors, is proposed. In addition, a method for identifying MITRE ATT&CK techniques in traffic logs is presented, which allows extracting information such as attack patterns or possible recommended mitigations and favours the subsequent risk management and automatic recommendation of responses to the detected incidents.

Based on the analysis of the state of the art and the literature, the modules that compose the cybersituational awareness environment and enable this characterisation are proposed. Firstly, an intrusion detection system, composed of unsupervised machine learning models trained to learn the normal behaviour within the data generated by heterogeneous devices, and to generate alerts when the system detects anomalous behaviour according to the characteristics of the information or according to the timestamp in which they are received. The operation of this module allows the identification of possible threats to the protected system carried out not only through the network but also using technologies such as Wi-Fi, Bluetooth, radio frequency, mobile networks or user behaviour.

In addition to considering other sources, the most common cyber-attacks are carried out through networks, whether internal or external. Therefore, in the cyber-situational awareness environment, a module focused on the characterisation of this type of attack is needed. Using decision tree models, it identifies in traffic logs a set of MITRE ATT&CK techniques. This not only allows distinguishing the type of cyber-attack being carried out from the tactic but also allows deducing vital information for the characterisation, such as the possible attack patterns of the adversary, in which step they are and recommended countermeasures to mitigate the effect of cyber-attacks on a given system.

To complete the proposed environment, the information from the incident detection module and the characterisation of techniques in traffic records are collected in an ontology for dynamic risk management. This also has the capacity to be interoperable, transferring the results of other methodologies to a common standard, ITSRM, so that results can be compared or information can be exchanged regarding responses in previous situations. From this knowledge model, a set of recommendations against cyber-attacks is extracted through decision support.

In summary, this Doctoral Thesis presents the characterisation of cyber-attacks as a contribution to the security of an organisation's systems and assets. The different proposals introduced address this objective from different approaches, to obtain as a result a global vision and a more complete protection.

Resumen

En la actualidad, el desarrollo constante de la inteligencia artificial tiene un gran impacto en la seguridad de los sistemas. Continuamente surgen nuevos desafíos que se responden con investigaciones centradas en la actualidad en el concepto de conciencia cibersituacional, que permite tener una visión global de la seguridad de un sistema.

Las tendencias actuales sobre ciberataques muestran que cada vez son más frecuentes y complejos, utilizando técnicas avanzadas y ataques de día cero que los hacen considerablemente más difíciles de detectar. A partir de ahí, surge la problemática sobre la falta de conocimiento referente a estos ciberataques, su caracterización. Esto permitirá adaptar las respuestas a los incidentes y plantear la capacidad de automatizarlas para que en tiempo real se puedan mitigar los riesgos asociados.

Esta es la motivación principal de la propuesta de esta Tesis Doctoral, un modelo de caracterización de ciberataques en entornos de conciencia cibersituacional. Para ello, se plantea en primer lugar un sistema de identificación de intrusiones a partir de fuentes de datos heterogéneas, como pueden ser sensores de comunicaciones físicos y lógicos. Además, se propone un método de identificación de técnicas MITRE ATT&CK en registros de tráfico, que permiten extraer información como patrones de ataque o posibles mitigaciones recomendadas y que favorecen la posterior gestión de riesgos y recomendación automática de respuestas frente a los incidentes detectados.

A partir del análisis del estado del arte y la literatura, se plantean los módulos que componen el entorno de conciencia cibersituacional y permiten esta caracterización. En primer lugar se propone un sistema de detección de intrusiones compuesto por modelos de aprendizaje automático no supervisado entrenados para conocer el comportamiento normal dentro de los datos generados por dispositivos heterogéneos, y generar alertas cuando el sistema detecte comportamiento anómalo según las características de la información o según el sello temporal en el que se reciben. El funcionamiento de este módulo permite identificar posibles amenazas al sistema protegido que se lleven a cabo no solo a través de la red sino utilizando tecnologías Wi-Fi, *Bluetooth*, radio frecuencia, redes móviles o comportamiento de usuario.

Además de considerar otras fuentes, los ciberataques más comunes se llevan a cabo mediante las redes, ya sean internas o externas. Por ello, en el entorno de conciencia cibersituacional se propone un módulo centrado en la caracterización de este tipo de ataques. Mediante modelos de árbol de decisión, identifica en registros de tráfico un conjunto de técnicas de MITRE ATT&CK. Esto no sólo permite distinguir el tipo de ciberataque que se está llevando a cabo a partir de la táctica sino que se puede deducir información vital para la caracterización, como los posibles patrones de ataque del adversario, en qué paso se encuentran y contramedidas recomendadas para poder mitigar el efecto de los ciberataques sobre un determinado sistema.

Para completar el entorno propuesto, la información de módulo de detección de incidentes y la caracterización de técnicas en registros de tráfico se recoge en una ontología para la gestión dinámica de riesgos. Ésta además tiene la capacidad de ser interoperable, trasladando los resultados de otras metodologías a un estándar común, ITSRM, de forma que puedan

compararse los resultados o intercambiar información referente a las respuestas en situaciones anteriores. De este modelo de conocimiento se extrae un conjunto de recomendaciones frente a los ciberataques mediante el soporte a la toma de decisiones.

En resumen, en esta Tesis Doctoral se presenta la caracterización de ciberataques como contribución a la seguridad de los sistemas y activos de una organización. Las distintas propuestas introducidas abordan este objetivo desde distintos enfoques, para obtener como resultado una visión global y una protección más completa.

Tabla de Contenido

Agradecimientos	v
Abstract	vi
Resumen	viii
Lista de Figuras	xiv
Lista de Tablas	xxviii
Abreviaturas y acrónimos	xxii
1 Introducción	1
1.1 Introducción	1
1.2 Contexto	2
1.3 Motivación	3
1.4 Objetivos	5
1.5 Estructura de la memoria	6
2 Metodología de investigación	9
2.1 Identificación de hipótesis	9
2.2 Identificación de objetivos	10
2.3 Identificación de tareas	10
2.4 Propuesta de metodología	11
3 Caracterización de ciberataques: <i>Cyber Threat Hunting</i>	13
3.1 Introducción	13
3.2 Ciberseguridad y <i>Cyber Threat Hunting</i>	14
3.3 Sistemas de Detección de Intrusiones	15
3.4 Modelos de inteligencia de amenazas de MITRE	17
3.4.1 MITRE ATT&CK	17
3.4.2 MITRE CAPEC	25
3.5 Conclusiones	41
4 Inteligencia Artificial aplicada a la Ciberseguridad	43
4.1 Introducción	43
4.2 Ontologías	44
4.2.1 Lenguajes de definición	45
4.2.2 Reglas de inferencia de conocimiento	47
4.2.3 Razonadores semánticos	49

4.3	Aprendizaje automático supervisado	51
4.3.1	Algoritmos	51
4.3.2	Pre-procesado de datos	54
4.3.3	Métricas	59
4.4	Aprendizaje automático no supervisado	62
4.4.1	Algoritmos de <i>clustering</i>	62
4.4.2	Reducción dimensional	65
4.4.3	Métricas	68
4.5	Conclusiones	69
5	Gestión de riesgos	71
5.1	Introducción	71
5.2	Estándares y metodologías	72
5.2.1	E BIOS	72
5.2.2	MAGERIT	73
5.2.3	MONARC	74
5.2.4	ITSRM	75
5.2.5	CRAMM	76
5.2.6	Comparación de metodologías	77
5.3	Conclusiones	81
6	Arquitectura global del modelo propuesto	83
6.1	Diseño del modelo de arquitectura	83
6.1.1	Sistema de detección de intrusiones	84
6.1.2	Sistema de caracterización de TTPs	84
6.1.3	Ontología interoperable para la gestión de riesgos	85
7	Propuesta para la detección automática de ciberataques en registros heterogéneos	87
7.1	Introducción	88
7.2	Trabajos relacionados	88
7.3	Propuesta	91
7.4	Diseño de la propuesta	92
7.4.1	Conjunto de datos de entrenamiento	93
7.4.2	Generación de datos sintéticos	98
7.4.3	Pre-procesado	99
7.4.4	Entrenamiento y validación	103
7.4.5	Gestión en tiempo real	118
7.5	Resultados	118
7.5.1	Comparación de modelos	118
7.5.2	Ratio de detección	120
7.5.3	Visualización de <i>clusters</i>	123
7.5.4	Comparación con trabajos anteriores	124
7.6	Conclusiones	125

8 Propuesta para la caracterización de técnicas MITRE ATT&CK en ciberataques	127
8.1 Introducción	128
8.2 Trabajos relacionados	128
8.3 Propuesta	131
8.3.1 Alternativa: Decisión mediante reglas	131
8.3.2 Diseño de la metodología propuesta	132
8.4 Desarrollo de la solución	133
8.4.1 Pre-procesado del conjunto de datos UWF-ZeekDataFall22	133
8.4.2 Elección de los algoritmos y entrenamiento de los modelos	138
8.4.3 Ontología para la gestión de información relacionada	141
8.5 Resultados	144
8.5.1 Modelos de aprendizaje automático	144
8.5.2 Caso de uso	153
8.6 Conclusiones	158
9 Propuesta de una metodología basada en ontologías para la interoperabilidad de marcos dinámicos de gestión de riesgos	161
9.1 Introducción	162
9.2 Trabajos relacionados	163
9.3 Diseño de la solución	165
9.3.1 Diseño de la ontología	168
9.3.2 Diseño del gestor de la ontología	170
9.4 Validación	177
9.4.1 Metodología de validación	177
9.4.2 Validación de los aspectos funcionales	178
9.4.3 Caso de uso 1: MONARC y MAGERIT	179
9.4.4 Caso de uso 2: Escenario general	183
9.5 Conclusiones	196
10 Validación global	199
10.1 Metodología de validación del sistema	199
10.2 Caso de uso 1: Evaluación de la caracterización de ciberataques	201
10.2.1 No se produce la caracterización de ciberataques	202
10.2.2 Se produce la caracterización de ciberataques	205
10.3 Caso de uso 2: Evaluación del entorno de conciencia cibersituacional	211
10.3.1 Generación de datos heterogéneos	212
10.3.2 Identificación de anomalías Wi-Fi y <i>Bluetooth</i>	213
10.3.3 Caracterización de técnicas en registros de tráfico	214
10.3.4 Ontología y gestión dinámica de riesgos	215
10.3.5 Visualización de la información	222
10.4 Conclusiones	223
11 Conclusiones y líneas futuras	225
11.1 Conclusiones de la investigación realizada	225

11.2 Contribución al conocimiento	227
11.3 Líneas futuras	232
Referencias	235
A Estructura de los catálogos del entorno de conciencia cibersituacional	247

Listas de Figuras

1.1	Ataques de día cero registrados entre 2012 y 2022. Datos: [32]	4
2.1	Fases de la metodología propuesta	12
2.2	Resumen de la metodología de investigación	12
3.1	Caracterización de ciberataques	15
4.1	Términos utilizados en ontologías	45
4.2	Esquema de funcionamiento del algoritmo de árbol de decisión	52
4.3	Esquema de funcionamiento del algoritmo <i>Random Forest</i>	53
4.4	Esquema de funcionamiento del algoritmo XGBoost	54
4.5	Diagrama de Venn sobre la relación entre información mutua y entropía	57
4.6	Composición de la matriz de confusión	60
5.1	Esquema del proceso de análisis y gestión de riesgos EBIOS	73
5.2	Esquema del proceso de análisis y gestión de riesgos MAGERIT	74
5.3	Esquema del proceso de análisis y gestión de riesgos MONARC	75
5.4	Esquema del proceso de análisis y gestión de riesgos ITSRM	76
5.5	Esquema del proceso de análisis y gestión de riesgos CRAMM	77
6.1	Arquitectura global del modelo propuesto	84
7.1	Módulo IDS en la arquitectura global propuesta	87
7.2	Arquitectura de la propuesta	92
7.3	Proceso para el diseño de un IDS utilizando fuentes de datos heterogéneos	93
7.4	Módulo de generación de datos	99
7.5	Pre-procesado de datos de sensor de redes móviles	100
7.6	Pre-procesado de datos de sensor de radiofrecuencia	100
7.7	Pre-procesado de datos de sensor <i>Bluetooth</i>	101
7.8	Pre-procesado de datos de sensor Wi-Fi	101
7.9	Pre-procesado de datos de cortafuegos	101
7.10	Pre-procesado de datos de SIEM	102
7.11	Pre-procesado de datos de actividad (UEBA)	102
7.12	Pre-procesado de datos de buscador (UEBA)	102
7.13	Pre-procesado de datos de documentos (UEBA)	102
7.14	Pre-procesado de datos de red (UEBA)	103

7.15	Pre-procesado de datos de procesos (UEBA)	103
7.16	Pre-procesado de datos de <i>sockets</i> (UEBA)	103
7.17	Módulo de entrenamiento y validación	104
7.18	Métrica Silhouette - K-Means - Número de <i>clusters</i> - Redes móviles	106
7.19	Métrica Silhouette - K-Means - Medida de la distancia - Redes móviles	106
7.20	Coste - K-Means - Número de <i>clusters</i> - Radiofrecuencia	107
7.21	Coste - K-Means - Medida de la distancia - Radiofrecuencia	107
7.22	Coste - K-Means - Número de <i>clusters</i> - <i>Bluetooth</i>	108
7.23	Coste - K-Means - Medida de la distancia - <i>Bluetooth</i>	108
7.24	Coste - K-Means - Número de <i>clusters</i> - Wi-Fi	109
7.25	Métrica Silhouette - K-Means - Medida de la distancia - Wi-Fi	109
7.26	Coste - K-Means - Número de <i>clusters</i> - Cortafuegos	110
7.27	Coste - K-Means - Medida de la distancia - Cortafuegos	110
7.28	Coste - K-Means - Número de <i>clusters</i> - SIEM	111
7.29	Métrica Silhouette - K-Means - Medida de la distancia - SIEM	111
7.30	Coste - K-Means - Número de <i>clusters</i> - Actividad (UEBA)	112
7.31	Coste - K-Means - Número de <i>clusters</i> - Buscador (UEBA)	113
7.32	Métrica Silhouette - K-Means - Medida de la distancia - Buscador (UEBA) .	113
7.33	Métrica Silhouette - K-Means - Número de <i>clusters</i> - Documentos (UEBA) .	114
7.34	Coste - K-Means - Medida de la distancia - Documentos (UEBA)	114
7.35	Coste - K-Means - Número de <i>clusters</i> - Red (UEBA)	115
7.36	Coste - K-Means - Medida de la distancia - Red (UEBA)	115
7.37	Coste - K-Means - Número de <i>clusters</i> - Procesos (UEBA)	116
7.38	Coste - K-Means - Medida de la distancia - Procesos (UEBA)	116
7.39	Coste - K-Means - Número de <i>clusters</i> - <i>Sockets</i> (UEBA)	117
7.40	Coste - K-Means - Medida de la distancia - <i>Sockets</i> (UEBA)	117
7.41	Módulo de gestión en tiempo real	118
7.42	Clasificación de eventos - Redes móviles	120
7.43	Clasificación de eventos - Radiofrecuencia	120
7.44	Clasificación de eventos - <i>Bluetooth</i>	121
7.45	Clasificación de eventos - Wi-Fi	121
7.46	Clasificación de eventos - Cortafuegos	121
7.47	Clasificación de eventos - SIEM	122
7.48	Representación de los <i>clusters</i> de cada fuente	123
7.48	Representación de los <i>clusters</i> de cada fuente	124
8.1	Módulo de caracterización de TTPs en la arquitectura global propuesta . . .	127
8.2	Metodología propuesta para la caracterización de TTPs	132
8.3	Información mutua de las columnas del <i>dataset</i>	137
8.4	Matriz de correlación del conjunto UWF-ZeekDataFall22	137
8.5	Estructura parcial de árbol del modelo árbol de decisión	139
8.6	Estructura completa de árbol del primer estimador del modelo <i>Random Forest</i>	139
8.7	Estructura parcial de árbol del segundo estimador del modelo <i>Random Forest</i>	140
8.8	Estructura parcial de árbol del tercer estimador del modelo <i>Random Forest</i> .	140
8.9	Estructura de árbol del primer estimador del modelo XGBoost	141

8.10	Estructura de árbol del segundo estimador del modelo XGBoost	141
8.11	Ontología propuesta	142
8.12	Matriz de confusión de la identificación de técnicas para el árbol de decisión	146
8.13	Matriz de confusión de la identificación de tácticas para el árbol de decisión .	146
8.14	Matriz de confusión de la clasificación binaria para el árbol de decisión . .	147
8.15	Curva ROC del modelo árbol de decisión	147
8.16	Matriz de confusión de la identificación de técnicas para <i>Random Forest</i> . .	149
8.17	Matriz de confusión de la identificación de tácticas para <i>Random Forest</i> . .	149
8.18	Matriz de confusión de la clasificación binaria para <i>Random Forest</i>	150
8.19	Curva ROC del modelo <i>Random Forest</i>	150
8.20	Matriz de confusión de la identificación de técnicas para XGBoost	151
8.21	Matriz de confusión de la identificación de tácticas para XGBoost	152
8.22	Matriz de confusión de la clasificación binaria para XGBoost	152
8.23	Curva ROC del modelo XGBoost	153
8.24	Relaciones entre los individuos del caso de uso	155
8.25	Estado inicial de la ontología para el caso de uso	155
8.26	Creación de los incidentes en la ontología del caso de uso	156
8.27	Mitigaciones asociadas con cada incidente	156
8.28	Amenazas generadas por cada incidente	157
8.29	Activo relacionado con los incidentes	157
9.1	Módulo de gestión de riesgos en la arquitectura global propuesta	161
9.2	Escenario de la propuesta	167
9.3	Esquema de la ontología	168
9.4	Secuencia de procesos del sistema	171
9.5	Mapa de calor para el cálculo del riesgo en EBIOS	173
9.6	Mapa de calor para el cálculo del riesgo en MAGERIT	173
9.7	Mapa de calor para el cálculo del riesgo en MONARC	174
9.8	Mapa de calor para el cálculo del riesgo en ITSRM	174
9.9	Mapa de calor para el cálculo del riesgo en CRAMM	175
9.10	Mapa de calor para el cálculo del riesgo residual	176
9.11	Flujo de trabajo del gestor de la ontología	177
9.12	Comparación del cálculo de riesgo en el escenario de MONARC	181
9.13	Comparación del cálculo de riesgo en el escenario de MAGERIT	182
9.14	<i>Primary Asset 1</i> en la ontología	183
9.15	Vulnerabilidades en la ontología	184
9.16	Ejemplo de amenazas	185
9.17	Ejemplo del efecto de las amenazas sobre los activos	186
9.18	Ejemplo de contramedida	188
9.19	Ejemplo de escenario de riesgo	189
9.20	Ejemplo de la generación de amenazas a partir de los escenarios de riesgo .	190
9.21	Ejemplo de cálculos de riesgo para la amenaza <i>User Error</i>	192
9.22	Cálculo del riesgo potencial global	192
9.23	Metodologías para la amenaza de <i>User Error</i>	193
9.24	Mitigación 2 - <i>Destructive Attack</i>	194

9.25 Efecto de la contramedida sobre el riesgo de la amenaza <i>Destructive Attack</i>	195
9.26 Riesgos calculados en el sistema: Metodología EBIOS, Riesgo potencial ITSRM y Riesgo residual ITSRM	195
10.1 Activos del escenario de validación	200
10.2 Nivel de riesgo potencial global del sistema. Validación - Caso de uso 1.1	204
10.3 Nivel de riesgo residual global del sistema. Validación - Caso de uso 1.1	205
10.4 Nivel de riesgo potencial global del sistema. Validación - Caso de uso 1.2	207
10.5 Nivel de riesgo residual global del sistema. Validación - Caso de uso 1.2	211
10.6 Nivel de riesgo potencial global del sistema. Validación - Caso de uso 2	218
10.7 Nivel de riesgo residual global del sistema. Validación - Caso de uso 2	221
10.8 Nivel de riesgo residual global del sistema modificado. Validación - Caso de uso 2	222
10.9 Prototipo de consola de mando y control	223
11.1 Contribuciones y publicaciones en el marco de la propuesta	230

Lista de Tablas

5.1	Aspectos generales de las metodologías - Comparación	77
5.2	Taxonomía de activos - Comparación	78
5.3	Valoración de activos - Comparación	79
5.4	Catálogo de amenazas- Comparación	79
5.5	Catálogo de vulnerabilidades - Comparación	80
5.6	Cálculo del riesgo - Comparación	80
7.1	Resultados de los trabajos previos	90
7.2	Campos de los datos de los dispositivos redes móviles	93
7.3	Campos de los datos de los dispositivos radiofrecuencia	94
7.4	Campos de los datos de los dispositivos <i>Bluetooth</i>	94
7.5	Campos de los datos de los dispositivos Wi-Fi	94
7.6	Campos de los datos de los dispositivos SIEM	95
7.7	Campos de los datos de los cortafuegos	95
7.8	Campos de los datos de los dispositivos UEBA - Monitor de actividad	96
7.9	Campos de los datos de los dispositivos UEBA - Buscador	96
7.10	Campos de los datos de los dispositivos UEBA - Procesos	96
7.11	Campos de los datos de los dispositivos UEBA - <i>Sockets</i>	97
7.12	Campos de los datos de los dispositivos UEBA - Documentos	97
7.13	Campos de los datos de los dispositivos UEBA - Red	98
7.14	Hiperparámetros de los modelos no supervisados	105
7.15	Valores de los hiper-parámetros seleccionados para los datos de redes móviles	106
7.16	Valores de los hiper-parámetros seleccionados para los datos de radiofrecuencia	107
7.17	Valores de los hiper-parámetros seleccionados para los datos <i>Bluetooth</i>	108
7.18	Valores de los hiper-parámetros seleccionados para los datos Wi-Fi	109
7.19	Valores de los hiper-parámetros seleccionados para los datos del cortafuegos .	110
7.20	Valores de los hiper-parámetros seleccionados para los datos del SIEM	111
7.21	Valores de los hiper-parámetros seleccionados para los datos de actividad (UEBA)	112
7.22	Valores de los hiper-parámetros seleccionados para los datos de buscador (UEBA)	113
7.23	Valores de los hiper-parámetros seleccionados para los datos de documentos (UEBA)	114
7.24	Valores de los hiper-parámetros seleccionados para los datos de red (UEBA)	115
7.25	Valores de los hiper-parámetros seleccionados para los datos de procesos (UEBA)	116
7.26	Valores de los hiper-parámetros seleccionados para los datos de <i>sockets</i> (UEBA)	117

7.27 Comparación de los modelos entrenados para cada fuente	119
7.28 Exactitud en la detección de anomalías	122
7.29 Comparación de los trabajos previos con esta propuesta	125
 8.1 Resumen de los resultados de trabajos previos	131
8.2 Codificación de etiquetas	134
8.2 Codificación de etiquetas	135
8.3 Registros por cada técnica tras el balanceo de datos	136
8.4 Mitigaciones recomendadas por ATT&CK para cada técnica	143
8.5 CAPECs asociados con cada técnica	144
8.6 Exactitud del modelo árbol de decisión y Tiempo de ejecución	145
8.7 Informe de clasificación del modelo de árbol de decisión	145
8.8 Exactitud del modelo <i>Random Forest</i> y Tiempo de ejecución	148
8.9 Informe de clasificación del modelo <i>Random Forest</i>	148
8.10 Exactitud del modelo XGBoost y Tiempo de ejecución	151
8.11 Informe de clasificación del modelo XGBoost	151
8.12 Asociación de información en el caso de uso para T1046	153
8.13 Asociación de información en el caso de uso para T1210	154
8.14 Asociación de información en el caso de uso para T1595	154
8.15 Comparación de los modelos entrenados	158
8.16 Comparación de trabajos previos con esta propuesta	159
8.17 Resultados de la investigación [126] aplicando esta propuesta	160
 9.1 Catálogo de vulnerabilidades. Caso de uso 1	178
9.2 Catálogo de activos - MONARC (PA: Activo principal, SA: Activo secundario)	179
9.3 Amenaza DoS - MONARC	180
9.4 Catálogo de activos - MAGERIT	181
9.5 Amenaza DoS - MAGERIT	182
9.6 Catálogo de activos. Caso de uso 2	184
9.7 Catálogo de vulnerabilidades. Caso de uso 2	184
9.8 Catálogo de amenazas. Caso de uso 2	185
9.9 Catálogo de activos afectado por las amenazas. Caso de uso 2	186
9.10 Catálogo de contramedidas. Caso de uso 2	187
9.11 Escenarios de Riesgo. Caso de uso 2	188
9.12 Incidentes registrados. Caso de uso 2	189
9.13 Catálogo de activos afectado por la amenaza del escenario de riesgo. Caso de uso 2	191
9.14 Amenazas de ejemplo - Cálculo del riesgo	196
9.15 Cálculo de riesgo global	196
 10.1 Catálogo de activos. Validación	200
10.1 Catálogo de activos. Validación	201
10.2 Información del sistema. Validación - Caso de uso 1	201
10.3 Incidentes registrados. Validación - Caso de uso 1.1	202
10.4 Escenarios de riesgo. Validación - Caso de uso 1.1	203
10.5 Cálculos de riesgo potencial. Validación - Caso de uso 1.1	203

10.6 Catálogo de mitigaciones. Validación - Caso de uso 1.1	204
10.7 Cálculos de riesgo residual. Validación - Caso de uso 1.1	205
10.8 Catálogo de vulnerabilidades. Validación - Caso de uso 1.2	206
10.9 Incidentes caracterizados. Validación - Caso de uso 1.2	206
10.10 Escenarios de riesgo. Validación - Caso de uso 1.2	207
10.11 Cálculos de riesgo potencial. Validación - Caso de uso 1.2	208
10.12 Catálogo de mitigaciones. Validación - Caso de uso 1.2	209
10.12 Catálogo de mitigaciones. Validación - Caso de uso 1.2	210
10.13 Cálculos de riesgo residual. Validación - Caso de uso 1.2	210
10.14 Registros Wi-Fi capturados - Caso de uso 2	212
10.15 Registros <i>Bluetooth</i> capturados - Caso de uso 2	213
10.16 Registros de tráfico de red capturados - Caso de uso 2	213
10.17 Caracterización de registros de tráfico capturados - Caso de uso 2	215
10.18 Información del sistema. Validación - Caso de uso 2	215
10.19 Catálogo de activos. Validación - Caso de uso 2	216
10.20 Catálogo de vulnerabilidades. Validación - Caso de uso 2	216
10.21 Escenarios de riesgo. Validación - Caso de uso 2	217
10.22 Incidentes registrados. Validación - Caso de uso 2	218
10.23 Cálculos de riesgo potencial. Validación - Caso de uso 2	218
10.23 Cálculos de riesgo potencial. Validación - Caso de uso 2	219
10.24 Catálogo de mitigaciones. Validación - Caso de uso 2	219
10.25 Soporte a la toma de decisiones. Validación - Caso de uso 2	220
10.26 Niveles de riesgo residual. Validación - Caso de uso 2	220
10.27 Soporte a la toma de decisiones modificada. Validación - Caso de uso 2	221
10.28 Niveles de riesgo residual modificado. Validación - Caso de uso 2	222

Abreviaturas y acrónimos

- AAE** Adversarial Auto Encoder
- APT** Advanced Persistent Threat
- AUC** Area Under Curve
- CAPEC** Common Attack Pattern Enumeration and Classification
- CAR** Cyber Analytics Repository
- CPE** Common Platform Enumeration
- CRAMM** CCTA Risk Analysis and Management Method
- CSV** Comma-Separated Values
- CTH** Cyber Threat Hunting
- CTI** Cyber Threat Intelligence
- CVE** Common Vulnerabilities and Exposures
- CVSS** Common Vulnerability Scoring System
- CWE** Common Weakness Enumeration
- DoS** Denegación de servicio
- E BIOS** Expressions des Besoins et Identification des Objectifs de Sécurité
- FN** Falso Negativo
- FP** Falso Positivo
- GAN** Generative Adversarial Networks
- GMM** Gaussian Mixture Modelling
- HIDS** Host Intrusion Detection System
- HW** Hardware
- IA** Inteligencia Artificial
- IDS** Intrusion Detection System
- IM** Información Mutua

- ISO** International Organization for Standardization
- ITSRM** IT Security Risk Management Methodology
- KNN** K-Nearest Neighbours
- MAGERIT** Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información
- MONARC** Method for an Optimized aNALysis of Risks by Cases
- NIDS** Network Intrusion Detection System
- OWL** Ontology Web Language
- PCA** Principal Component Analysis
- PILAR** Procedimiento Informático-Lógico para el Análisis de Riesgos
- RDF(S)** Resource Description Framework (Schema)
- ROC** Receiver Operating Characteristic
- SMOTE** Synthetic Minority Over-sampling Technique
- SPIN** SPARQL Inferencing Notation
- SVM** Support Vector Machine
- SW** Software
- SWRL** Semantic Web Rule Language
- TF-IDF** Term Frequency-Inverse Document Frequency
- TFN** Tasa de Falsos Negativos
- TFP** Tasa de Falsos Positivos
- t-SNE** T-Distributed Stochastic Neighbor Embedding
- TTP** Tácticas, Técnicas y Procedimientos
- TVN** Tasa de Verdaderos Negativos
- TVP** Tasa de Verdaderos Positivos
- UEBA** User and Entity Behavior Analytics
- UMAP** Uniform Manifold Approximation and Projection
- UPM** Universidad Politécnica de Madrid
- URI** Uniform Resource Identifier

VN Verdadero Negativo

VP Verdadero Positivo

W3C World Wide Web Consortium

WSSSE Within Set Sum of Squared Error

XGBoost Gradient Boosted Decision Trees

Capítulo 1

Introducción

1.1 Introducción

La revolución digital ha situado a la ciberseguridad en el punto de mira de las nuevas investigaciones, siendo la base fundamental para la protección de los datos, sistemas y redes. Al mismo tiempo, las amenazas también han evolucionado, convirtiéndose cada vez en ataques más complejos y sofisticados que ponen a prueba la capacidad de los sistemas y procesos actuales para defenderse.

Por este motivo, a día de hoy, no es suficiente con detectar las amenazas de forma tradicional, sino que las nuevas herramientas como la Inteligencia Artificial (IA) son imprescindibles para mantener la confidencialidad, integridad y disponibilidad de cualquier infraestructura, en combinación con la adopción de una metodología de análisis y gestión de riesgos adecuada al entorno. Así se conforma una plataforma de conciencia cibersituacional, capaz de detectar y reaccionar frente a intrusiones, proporcionando una visión global de la ciberseguridad del sistema en tiempo real, como se propone en la investigación realizada para esta Tesis Doctoral.

Estos sistemas están diseñados para recopilar y analizar información de la seguridad del entorno en tiempo real, como registros de eventos o inteligencia de amenazas, con el objetivo de detectar ciberataques, calcular el nivel de riesgo y tomar decisiones. Las múltiples fuentes de información que se utilizan para obtener esta visión global a la vez dificultan los procesos de detección de intrusiones y gestión de riesgos. La interconexión y creciente dependencia de la tecnología hace que la diversidad de infraestructuras y tecnologías genere datos heterogéneos que deben ser procesados dentro de la misma plataforma, aplicando tecnologías avanzadas como el aprendizaje automático y la IA que complementen el conocimiento humano. Así, los procesos redundantes se delegan en las tecnologías, que realizan estas tareas de forma más eficiente, mientras que el trabajo de los expertos se focaliza en el análisis de las situaciones en función de los datos obtenidos a partir de la conciencia cibersituacional.

Esta Tesis Doctoral se centra en el modelado de la caracterización de ciberataques para entornos de conciencia cibersituacional, diseñado para enriquecer la detección de intrusiones en entornos físico-lógicos y los procesos de gestión de riesgos y soporte a la toma de decisiones. Está compuesto por tres módulos fundamentales: un sistema de detección de intrusiones

(*Intrusion Detection System* o IDS), un sistema de caracterización de tácticas y técnicas MITRE ATT&CK en registros de tráfico y una metodología interoperable de gestión de riesgos; que en conjunto proporcionan una visión completa y en contexto de los ciberataques recibidos, ofreciendo la posibilidad de responder ante ellos de manera efectiva y adaptada.

1.2 Contexto

Hoy en día, la ciberseguridad depende completamente de la tecnología y, a medida que ésta evoluciona, la complejidad de los ciberataques y el número de incidentes detectados todos los años incrementa, sobre pasando las estrategias convencionales de defensa como los sistemas de control de acceso o los antivirus, pero gracias a esta evolución también surgen nuevas propuestas [34]. En este contexto dinámico, la conciencia cibersituacional surge como un concepto fundamental, que va más allá de la simple detección de amenazas para ofrecer una comprensión profunda y contextualizada de estos entornos en constante cambio.

Las bases de este término ya aparecen en la literatura desde 1995, aunque ha ido evolucionando hasta lo que se conoce hoy en día como “conciencia cibersituacional” [53], haciendo referencia a la capacidad de comprender y evaluar de manera efectiva el entorno digital en tiempo real, identificando posibles amenazas, vulnerabilidades y anomalías. Esto permite anticipar posibles riesgos y adaptar rápidamente un sistema a los cambios en el entorno.

Desde la aparición de los ciberataques, este campo ha avanzado a gran velocidad. Originalmente, estos incidentes eran simples y puntuales, siguiendo motivaciones variadas. Sin embargo, con el desarrollo de las tecnologías, las amenazas se han extendido, convirtiéndose en ciberataques frecuentes y complejos con los que la sociedad se está habituando a convivir. Los ciberdelincuentes también tienden a estar más organizados y tienen a su disposición herramientas y habilidades avanzadas, siendo el robo de datos uno de los ataques más comunes [32]. Al convertirse los incidentes de ciberseguridad en una preocupación de ámbito global, que afecta tanto a individuos particulares como a grandes empresas, se pone de manifiesto la necesidad de soluciones más avanzadas y proactivas como la caracterización de estos ciberataques o *Cyber Threat Hunting* (CTH) para poder hacerles frente.

Los enfoques tradicionales para la detección de incidentes presentan limitaciones a día de hoy para identificar los ataques más sofisticados en entornos heterogéneos, como pueden ser las Amenazas Avanzadas Persistentes (*Advanced Persistent Threats* o APTs) [34]. Para responder a estos cambios, surgen las primeras plataformas de conciencia cibersituacional. Con el tiempo, han evolucionado hasta abordar los nuevos problemas que han ido surgiendo, como el tratamiento de datos masivos que se produce en entornos 5G o la toma automática de decisiones adaptada a los incidentes registrados. Apoyándose en tecnologías novedosas, principalmente la IA debido a su capacidad de extraer patrones en información que el ser humano pasa por alto, estos entornos se están convirtiendo en sistemas capaces de detectar anomalías en conjuntos de datos de manera mucho más eficiente.

Actualmente, en términos de ciberseguridad, no importa cómo de bueno sea un diseño, porque los atacantes encontrarán la manera de acceder. Por ello, los sistemas y sus gestores deben estar preparados para la defensa. El concepto de “Sistema Inmune Artificial” [34], basado en

IA, establece un sistema en la actualidad que es capaz de detectar ciberataques y responder ante ellos de forma rápida y con menor coste gracias a la implementación de la tecnología. Por eso, en un momento donde la capacidad de adaptación y respuesta rápida se vuelve crítica, la ventaja principal de las plataformas de conciencia cibersituacional es que no buscan sustituir las protecciones actuales, sino trabajar como un complemento, integrándose para proporcionar la mayor cantidad de información sobre la situación contextualizada y entendible a los analistas de ciberseguridad en la que basar la toma de decisiones.

1.3 Motivación

Tras analizar el contexto en el que se enmarca esta Tesis Doctoral, se han identificado una serie de retos y limitaciones en la ciberseguridad que afectan a día de hoy [31]. Todos ellos comparten una confianza excesiva en las soluciones automatizadas; las tareas de ciberseguridad no se deben delegar únicamente en la tecnología, sino que es imprescindible la supervisión humana. Además, con el auge de la IA se ha desarrollado una dependencia hacia los datos de calidad para obtener modelos eficientes, lo cual no es posible en todos los ámbitos.

Las últimas tendencias en ciberseguridad buscan una defensa autónoma, que se caracteriza por un enfoque proactivo y una respuesta inmediata. Esta estrategia destaca la importancia de los modelos de comportamiento de la red y especialmente de comportamiento de usuario, que permiten identificar nuevas amenazas, como atacantes internos. En este contexto, las investigaciones se centran en la automatización de la respuesta para anticiparse a los incidentes, contribuyendo a la toma de decisiones para mitigar los ciberataques detectados en tiempo real.

Del análisis de los registros sobre ciberataques en 2022 presentados por el Centro Criptológico Nacional (CCN) [32] se extrae que el cibercrimen está viviendo un auge del *malware* enfocado hacia el robo de información y el compromiso tanto de sistemas remotos como de datos. Los ataques de día cero, explotando vulnerabilidades hasta el momento desconocidas, son una de las principales preocupaciones de la ciberseguridad. Como se observa en la Figura 1.1, la tendencia de los ciberataques es ascendente, alcanzando el máximo en 2021, lo que se traduce en que los sistemas actuales no son capaces de identificarlos correctamente y permitiendo que los atacantes consigan su objetivo. Especialmente destacan los ataques que automáticamente cambian sus características (ataques polimórficos) por lo que no coinciden con ningún registro previo. Estos datos resaltan la necesidad de incluir la protección de los sistemas en todos los niveles de las organizaciones.

Por otra parte, destaca la urgencia por trabajar en la capacidad de monitorización de Tácticas, Técnicas y Procedimientos (TTPs), ya que este incremento de ciberataques de *phishing* y fraude a través de la red va asociado a la intención de los atacantes de obtener información. Las técnicas más observadas en los incidentes de 2022 refuerzan el requisito de proteger los sistemas, no simplemente detectar las amenazas. Estas fueron T1588, T1587, T1190, T1585, T1591, T1595, T1583, T1212 y T1133, que en concreto tienen los siguientes objetivos: obtención y desarrollo de capacidades, explotación de aplicaciones, creación de cuentas, obtención de información, escaneo de la red, adquisición de la infraestructura y uso de servicios remotos.



Figura 1.1: Ataques de día cero registrados entre 2012 y 2022. Datos: [32]

Para afrontar estos retos, la tecnología principal es la IA [32], aprovechando todas sus ramas, desde las ontologías hasta el aprendizaje automático. La primera capa de una estrategia defensiva siguen siendo los IDS, pero deben evolucionar para hacer frente a las limitaciones actuales. En la detección de posibles anomalías en sensores heterogéneos, datos que no encajan con los *datasets* de entrenamiento conocidos, requiere monitorizar y analizar grandes cantidades de información y es donde tiene cabida el aprendizaje automático.

Además, los IDS tienden a producir alertas falsas, dificultando la identificación de los incidentes reales, especialmente en entornos donde la cantidad de información es alta y muy variada, por lo que se establecen modelos capaces de identificar las técnicas empleadas por los atacantes en las redes de tráfico. Esto repercute en la caracterización de los ciberataques, definiendo la respuesta ante un incidente registrado. La efectividad de esta respuesta se evalúa mediante los procesos de gestión de riesgo, que analizan las amenazas a las que se enfrentan los activos de una organización y tratan de mitigarlas. Para que este proceso sea completo se debe trabajar con información muy diversa, supervisada por expertos. En ocasiones, la falta de una visión completa del entorno, de información externa -como la inteligencia de amenazas-, o de herramientas que faciliten la caracterización homogénea y el manejo de los riesgos -como las ontologías-, influye en que la identificación y mitigación de los riesgos no sea eficiente y completa. El uso de ontologías ganó relevancia con la llegada de la Web Semántica, cuyo objetivo es estructurar y formalizar la información para que pueda ser comprendida de manera inteligente por aplicaciones. La ventaja principal de esta herramienta radica en su capacidad de inferencia de conocimiento, extrayendo nueva información a partir de los datos recabados.

En resumen, los principales retos identificados surgen alrededor de la necesidad de conocer información sobre los ciberataques, es decir, caracterizarlos. Los avances deben ir enfocados hacia la detección de ataques de día cero, la respuesta automatizada de incidentes, la adaptabilidad a los comportamientos de los atacantes y la integración con las infraestructuras existentes. Esta evolución, sin embargo, no debe oponerse a la necesidad demostrada de

establecer una política de defensa en profundidad y de la colaboración e intercambio de información que pueda ayudar a otras organizaciones a preparar la defensa frente a nuevos ciberataques. Además, siendo un contexto en constante cambio, debe adaptarse dinámicamente para responder a las tácticas y técnicas que puedan surgir.

La necesidad de abordar estos desafíos motiva la propuesta de un modelo de caracterización de ciberataques en entornos de conciencia cibersituacional. La arquitectura de esta investigación se compone de un IDS diseñado para abordar los problemas de diversidad y volumen de datos, siendo entrenado con modelos de aprendizaje automático no supervisado para poder adaptarse al comportamiento habitual de los sensores de fuentes heterogéneas y limitar las falsas alertas en tiempo real. Además, al complementarlo con un sistema de caracterización de TTP en los registros de tráfico, la propuesta pretende aportar información vital para decidir si un registro de tráfico es o no anómalo, a partir de una mejor comprensión de la amenaza y permitiendo la recomendación de respuestas y estrategias de defensa adecuadas a los incidentes detectados para contrarrestar las técnicas identificadas.

Además, las contribuciones que se proponen en este trabajo de Tesis Doctoral han podido ser validadas en proyectos de investigación y desarrollo en el ámbito, como son el proyecto PLICA, de ámbito nacional para el Ministerio de Defensa; o ECYSAP, para la Comisión Europea, que permiten validar el modelo de caracterización de ciberataques propuesto.

En resumen, la investigación llevada a cabo trata de abordar los retos actuales de la ciberseguridad, representando un avance significativo, con el potencial de influir en la forma en que las organizaciones abordan y gestionan los riesgos.

1.4 Objetivos

El objetivo principal de esta Tesis Doctoral es el modelado de la caracterización de ciberataques para entornos de conciencia cibersituacional, basándose en una metodología de análisis y gestión de riesgos interoperable con los principales marcos de trabajo en este ámbito. Este sistema permite detectar incidentes de ciberseguridad procedentes de distintas fuentes, caracterizar en los registros de tráfico las técnicas utilizadas por los atacantes y, por último, realizar un proceso de análisis y gestión de riesgos con la recomendación final de contramedidas para responder ante estos ataques o protegerse ante las técnicas utilizadas.

Para ello, se han identificado un conjunto de objetivos que determinan las tareas a realizar y las principales contribuciones de la Tesis Doctoral:

- Objetivo 1: Definición de un *dataset* de registros procedentes de fuentes heterogéneas para la detección de incidentes de ciberseguridad.
- Objetivo 2: Diseño de un modelo de pre-procesado de datos.
- Objetivo 3: Desarrollo de un sistema de detección de intrusiones basado en algoritmos de aprendizaje automático.
- Objetivo 4: Diseño de un sistema de caracterización de tácticas y técnicas MITRE ATT&CK.

- Objetivo 5: Planteamiento de un sistema de recomendación de contramedidas frente a ciberataques basado en un módulo de soporte a la toma de decisiones.
- Objetivo 6: Propuesta de una ontología interoperable de análisis y gestión dinámica de riesgos aplicable a metodologías ampliamente aceptadas.
- Objetivo 7: Integración de los módulos individuales en un entorno global de conciencia cibersituacional.

1.5 Estructura de la memoria

La memoria se estructura en capítulos de la siguiente forma:

- El **Capítulo 1** realiza una introducción al entorno en el que se desarrolla la tesis, enfocándose en el contexto de donde surge la motivación y los principales objetivos de esta investigación.
- El **Capítulo 2** presenta la metodología seguida para el desarrollo de la tesis, definiendo las hipótesis a validar, las tareas para la consecución de los objetivos y las contribuciones obtenidas a partir del trabajo realizado.
- En el **Capítulo 3** se introduce la base de esta propuesta, la caracterización de ciberataques. En primer lugar se profundiza en el contexto, la ciberseguridad y el concepto de CTH, definiendo los componentes principales de la defensa de un sistema: la detección de intrusiones y la caracterización del comportamiento del usuario a través del marco MITRE.
- En el **Capítulo 4** se describe el marco teórico relacionado con los conceptos de IA aplicados en el ámbito de la ciberseguridad, poniendo el foco especialmente en las tecnologías en las que se basa esta investigación, como son las Ontologías y el Aprendizaje Automático Supervisado y No Supervisado. Aquí se presenta el estudio llevado a cabo para elegir las herramientas utilizadas para el desarrollo de esta Tesis Doctoral.
- El **Capítulo 5** introduce el estado del arte en relación con el Análisis y la Gestión de Riesgos. Incluye información sobre las metodologías y estándares más utilizados a nivel europeo, y un análisis de sus principales características en relación con la capacidad de interoperabilidad.
- En el **Capítulo 6** se presenta por primera vez la propuesta planteada en global a raíz del estudio del estado del arte, introduciendo cada módulo que será detallado en los siguientes capítulos.
- El **Capítulo 7** incluye el trabajo realizado para la detección de incidentes de seguridad en registros de fuentes heterogéneas. El análisis llevado a cabo engloba un estudio de los trabajos relacionados y de las posibles herramientas y tecnologías utilizadas. A continuación, se presenta la propuesta definida, su desarrollo y resultados y finalmente las conclusiones extraídas de esta investigación.
- En el **Capítulo 8** se desarrolla la investigación principal relacionada con la caracteri-

zación de ciberataques, la identificación de técnicas ATT&CK en registros de tráfico y la extracción de otra información relacionada con las TTPs como mitigaciones o debilidades. El trabajo realizado comienza con un estudio de la literatura previa, a partir de la que se identifica el problema y se formula una propuesta. A continuación, se presenta su desarrollo, resultados y finalmente las conclusiones extraídas.

- A lo largo del **Capítulo 9** se presenta una metodología para la gestión dinámica de riesgos basada en una ontología interoperable con información procedente de distintos marcos de gestión de riesgos. Ésta reacciona a los incidentes heterogéneos identificados y los ciberataques que se caracterizan para llevar a cabo los procesos de análisis, gestión y evaluación de riesgos, terminando con un soporte a la toma de decisiones para la recomendación de contramedidas que respondan ante ellos. Tras el estudio de las investigaciones anteriores, se diseña una ontología y un gestor que aceptan información de las metodologías analizadas en el Capítulo 5, las traduce a un marco interoperable para poder compararlas y realiza la recomendación de contramedidas en base a las condiciones del entorno. Este desarrollo se verifica con dos casos de uso y finalmente se extraen las conclusiones de la propuesta.
- En el **Capítulo 10** se define la metodología de validación del conjunto a través de varios casos de uso que ponen a prueba el valor de la caracterización definida y el funcionamiento del sistema global.
- Finalmente, el **Capítulo 11** recoge las conclusiones extraídas de la propuesta de caracterización de ciberataques presentada, destacando las principales contribuciones, y las líneas de investigación futuras que surgen de esta Tesis Doctoral.
- En el **Apéndice A** se detalla el formato y contenido de los catálogos que se utilizan como entrada en el entorno de conciencia cibersituacional para llevar a cabo los procesos de gestión de riesgos.

Capítulo 2

Metodología de investigación

Esta Tesis Doctoral se basa en el método científico [44] para el planteamiento de la metodología a seguir durante su desarrollo. En primer lugar, se estudia el contexto y se propone un conjunto de hipótesis que se validarán o refutarán a lo largo del desarrollo de la Tesis Doctoral. A continuación, se define el conjunto de objetivos que permitan comprobar estos supuestos, y se definirán en tareas. En conjunto la metodología se compone de los siguientes pasos: estudio teórico, planteamiento de hipótesis, experimentación y validación y análisis de los resultados.

2.1 Identificación de hipótesis

En primer lugar, se llevó a cabo un estudio teórico del estado del arte y la literatura desarrollada hasta la fecha. Ésto permite localizar áreas donde la contribución de esta investigación tuviese valor y aportase soluciones a problemas reales. Así, se plantean las hipótesis que se encuentran en la base del desarrollo:

- Hipótesis 1: Los IDS basados en aprendizaje automático supervisado han demostrado tener grandes resultados en el análisis de tráfico de red. Sin embargo, cuando se trata de sensores son datos heterogéneos y sin etiquetar, y no existen *datasets* que permitan el entrenamiento de estos modelos de manera fiable reflejando estos comportamientos, por lo que se podrían obtener buenos resultados con un conjunto de algoritmos no supervisados tratados correctamente.
- Hipótesis 2: El proceso de análisis y gestión de riesgos requiere de estudios exhaustivos previos del entorno. Mediante la caracterización del mismo, a través de la identificación de técnicas y tácticas en los registros de tráfico que generan incidentes en el sistema bajo estudio, se pueden obtener mejores resultados en la reducción de riesgos y aplicación de contramedidas.
- Hipótesis 3: La gestión global en un entorno de conciencia cibersituacional que aúne desde el proceso de detección de intrusiones hasta la recomendación de contramedidas permitirá obtener una visión global de los indicadores de la situación del sistema como el nivel de riesgo y adaptar las acciones al contexto, mejorando las respuestas ante estos incidentes aplicando contramedidas de acuerdo a los ciberataques y no estrategias

genéricas.

2.2 Identificación de objetivos

En el capítulo anterior se introdujeron los principales objetivos de la Tesis Doctoral, basados en la caracterización de ciberataques en entornos de conciencia cibersituacional. Éstos se plantean a partir de las hipótesis anteriores, con el objetivo de validarlas o refutarlas, pasando a las siguientes fases del método científico: el diseño de la solución, su desarrollo y experimentación, y validación y análisis de los resultados.

En relación con la primera hipótesis, surgen los Objetivos 1 - 3, centrados en la identificación de anomalías que puedan asociarse con un ciberataque entre información de fuentes heterogéneas. Para ello, se requiere el diseño de conjuntos de datos que representen la información generada por estas fuentes y que se utilizará para entrenar los modelos de aprendizaje automático. Estos datos, que dependen en gran medida de los dispositivos de origen, no están etiquetados, por lo que son idóneos para verificar la Hipótesis 1. Además, es necesario tratar estos datos para que el algoritmo sea capaz de procesarlos. Al proceder de fuentes distintas, cada conjunto de datos tiene unos atributos determinados y por lo tanto requiere un pre-procesado adaptado a ellos. Finalmente, se desarrollará el sistema de detección de intrusiones basado en algoritmos no supervisados que permitirá comprobar la primera hipótesis.

Por otra parte, el planteamiento de la segunda hipótesis conlleva objetivos que se definen desde dos perspectivas: la caracterización de las técnicas en registros de tráfico y la reducción de riesgos y aplicación adaptada y recomendada de contramedidas. Aquí surgen los Objetivos 4-6, que consisten en el diseño de un sistema que caracterice el tráfico según la matriz MITRE ATT&CK, el planteamiento de un sistema de recomendación de contramedidas en base a la información obtenida en el anterior y su asociación a través de una ontología de gestión dinámica de riesgos capaz de calcular el nivel de riesgo que implican los ciberataques recibidos y proponer mitigaciones adaptadas para responderlos gracias a su caracterización.

Por último, la Hipótesis 3 se plantea condicionada por la validación de las dos anteriores, ya que contempla la propuesta de esta Tesis Doctoral en conjunto. Para su validación se propone el Objetivo 7, que consiste en la integración de los módulos individuales en un entorno global de conciencia cibersituacional.

2.3 Identificación de tareas

La consecución de los objetivos se divide en distintas tareas a realizar, para llevar a cabo la investigación de forma ordenada y validar o descartar las hipótesis planteadas:

- Tarea 1: Estudio del estado del arte referente a los sistemas IDS.
- Tarea 2: Definición del escenario de sensores heterogéneos sobre los que trabaja el modelo de IDS propuesto en esta Tesis Doctoral. De aquí surge la primera hipótesis, y los objetivos asociados (Objetivo 1, Objetivo 2, Objetivo 3)
- Tarea 3: Caracterización y definición de un conjunto de datos de fuentes heterogéneas.

- Tarea 4: Planteamiento del modelo de IDS basado en algoritmos no supervisados.
- Tarea 5: Desarrollo y validación del modelo. En este punto se valida la Hipótesis 1 y sus objetivos.
- Tarea 6: Estudio del estado del arte referente a la identificación de tácticas y técnicas.
- Tarea 7: Planteamiento del escenario de caracterización de TTPs.
- Tarea 8: Análisis de las metodologías más adoptadas de análisis y gestión de riesgos. Tras este análisis, se plantean la segunda y la tercera hipótesis y los Objetivos 4, 5, 6 y 7.
- Tarea 9: Elección y pre-procesado del conjunto de datos de entrenamiento.
- Tarea 10: Definición del modelo de caracterización de técnicas.
- Tarea 11: Planteamiento del sistema de recomendación de contramedidas.
- Tarea 12: Desarrollo y validación del módulo. Aquí se validan los Objetivos 4 y 5.
- Tarea 13: Diseño de una ontología y su gestor para la aplicación de una metodología interoperable de análisis y gestión de riesgos.
- Tarea 14: Desarrollo y validación del sistema de ontologías para la gestión de riesgos. En este punto se valida el Objetivo 6 y la Hipótesis 2.
- Tarea 15: Validación de la plataforma de conciencia cibersituacional en conjunto. Tras la última tarea se valida la tercera hipótesis y el Objetivo 7.

2.4 Propuesta de metodología

Tras la presentación de las hipótesis, los objetivos y tareas definidas para la investigación de esta Tesis Doctoral y como se ha definido previamente en este capítulo, la metodología propuesta se compone de cuatro fases organizadas de forma secuencial, que se presentan en la Figura 2.1.

En el estudio teórico del contexto de la propuesta se llevarán a cabo las Tareas 1, 2, 6, 7 y 8, consistentes en el análisis del estado del arte y el planteamiento de escenarios sobre los que trabajar. Esto finaliza con la definición de las hipótesis, en la segunda fase de la metodología.

Para seguir, comienza la experimentación que permita validar o refutar estas hipótesis. Por ello, en paralelo comienzan las Tareas 3 y 9, mediante las que se definen los conjuntos de datos con los que se trabajará. A continuación, las Tareas 4, 10 y 13, que plantean la arquitectura de los distintos módulos, y finalmente la tarea 11, que plantea el sistema de recomendación de contramedidas según el modelo de caracterización establecido.

Para iniciar la fase de validación y análisis de resultados, se realizan las Tareas 5, 12 y 14. La primera finaliza con la validación de la primera hipótesis, mientras que las otras permiten el respaldo de la segunda. Con estas condiciones cumplidas, se lleva a cabo la Tarea 15, que permite la validación global de la propuesta de la Tesis Doctoral y la confirmación de la

última hipótesis. La validación de las premisas se llevará a cabo mediante la comparación de resultados con desarrollos anteriores en el caso de la Hipótesis 1, y mediante casos de uso que prueben los resultados mejorados en las Hipótesis 2 y 3.

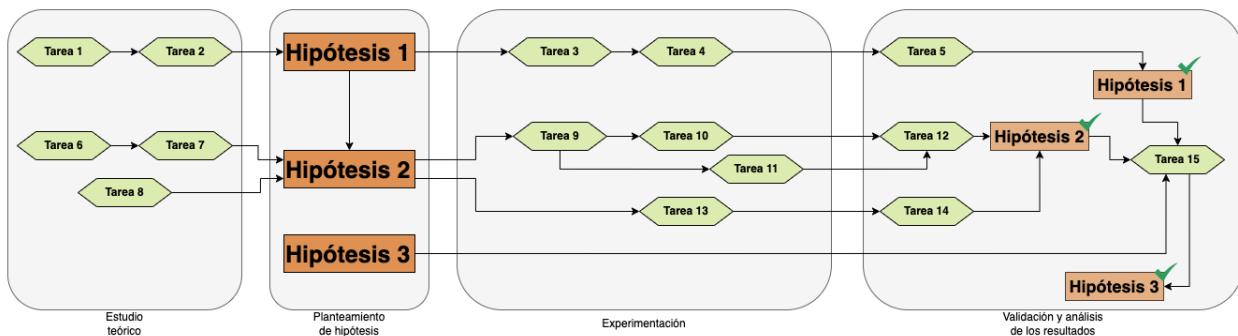


Figura 2.1: Fases de la metodología propuesta

En la Figura 2.2 se resume la metodología presentada, con la organización de tareas y objetivos en función de las hipótesis que se plantean:

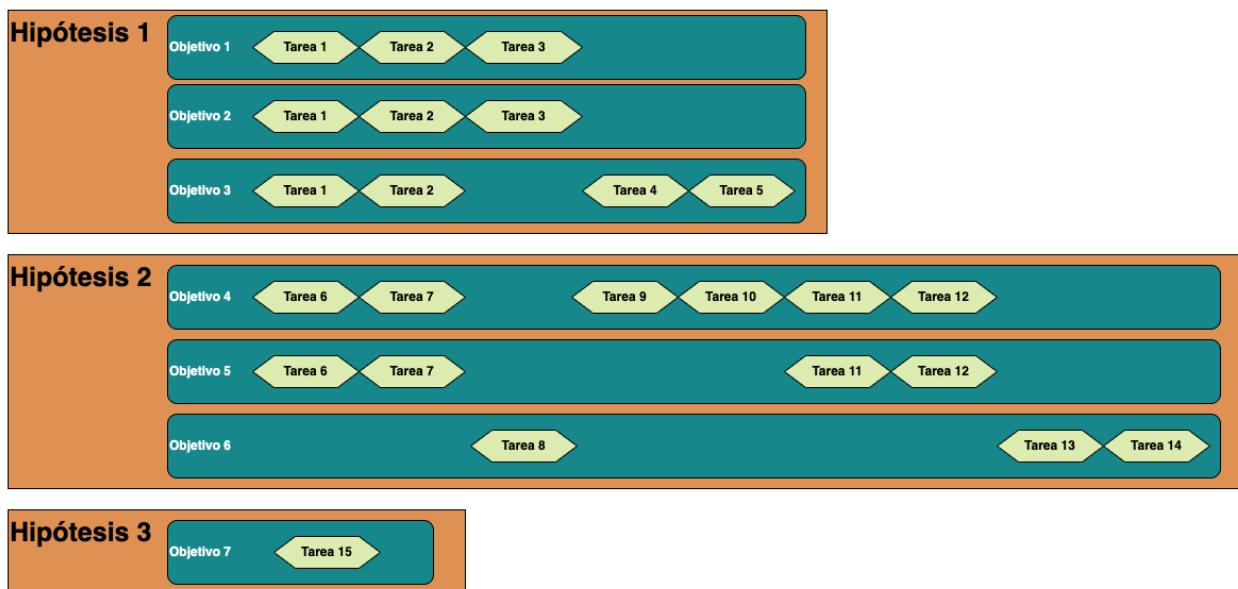


Figura 2.2: Resumen de la metodología de investigación

Capítulo 3

Caracterización de ciberataques: *Cyber Threat Hunting*

La caracterización de ciberataques es el concepto fundamental que reside en la base de los entornos de conciencia cibersituacional, ya que permite identificar y responder de manera proactiva a los incidentes de seguridad a partir de distintas tecnologías, objetivo fundamental de esta Tesis Doctoral. Este capítulo se centrará en el contexto de la ciberseguridad y la caracterización de los ciberataques. Primero, en la Sección 3.1 se presenta una introducción a este ámbito. A continuación, en la Sección 3.2, se define el concepto de CTH y el estado de la ciberseguridad a día de hoy. En la Sección 3.3 se presenta el concepto de Sistema de Detección de Intrusiones, uno de los elementos fundamentales del entorno modelado en esta propuesta. Más adelante, en la Sección 3.4 se detalla el marco de trabajo MITRE, en concreto dos de sus propuestas, ATT&CK y CAPEC, fundamentales para la caracterización de los ciberataques. Finalmente, en la Sección 3.5 se resumen las conclusiones obtenidas en el Capítulo.

3.1 Introducción

Las nuevas generaciones de ataques, que aplican *malware* mejorado, no pueden detectarse mediante técnicas tradicionales [34]. La IA en la ciberseguridad ofensiva se aplica en tareas de fuerza bruta, criptoanálisis o en el desarrollo de ciberataques avanzados, mientras que en el lado defensivo se centra en el desarrollo de herramientas capaces de detectarlos.

La ciberseguridad se encuentra en una batalla entre atacantes y defensores, que constantemente buscan mejorar sus herramientas [31]. La identificación de nuevas vulnerabilidades que explotar y técnicas de intrusión se responde con técnicas de anticipación, detección y respuesta renovadas.

Por tanto, la caracterización de ciberataques, que permite identificarlos y responder en tiempo real, está compuesta tanto de la detección de amenazas como de un análisis profundo del comportamiento del sistema y del atacante. Los incidentes actualmente utilizan tácticas de difícil detección, a las que únicamente puede hacer frente la IA en su búsqueda de patrones anómalos (actividades no autorizadas o inesperadas, franjas temporales extrañas o envío de

datos de gran volumen).

En este proceso, aparecen sistemas como los IDS basados en IA, capaces de detectar comportamientos anómalos, que son fundamentales para la protección de un sistema, y que gracias a la aplicación de estas tecnologías han incrementado su efectividad en la identificación de ciberataques en tiempo real. Por otro lado, se encuentra el trabajo de organizaciones, como MITRE, que recaban información sobre el comportamiento de los atacantes y permiten complementar esta detección de incidentes con datos fundamentales para la gestión de los riesgos generados y para los sistemas de respuesta ante estas amenazas. Estos últimos, gracias a la aplicación de herramientas como las ontologías y la inferencia de conocimiento, pueden razonar sobre toda esta información, recomendando acciones predefinidas y adecuadas al contexto para mitigar un ataque.

La ciberseguridad en la era moderna depende de la fuerte unión y enriquecimiento entre el conocimiento de expertos y los sistemas tradicionales, la detección de intrusiones, análisis de los datos y gestión de riesgos y respuesta automatizada, haciendo de la caracterización de ciberataques una parte fundamental de los entornos de conciencia cibersituacional. Éstos son capaces de tomar decisiones en tiempo real frente a los ataques detectados que se basan en datos y experiencias previas, mejorando la reducción de riesgos y mitigando las consecuencias de las amenazas.

3.2 Ciberseguridad y *Cyber Threat Hunting*

La ciberseguridad [34] es una práctica que se centra en proteger sistemas, redes y datos de posibles ciberamenazas. Para ello se aplican tecnologías como la criptografía o la IA para desarrollar sistemas de detección de intrusiones o antivirus.

Los nuevos avances y una sociedad cada vez más conectada generan grandes volúmenes de datos que son difíciles de tratar con aproximaciones tradicionales, mientras que gracias a la IA [31], encontrar patrones en ellos que puedan indicar ataques es una tarea automática, considerando como anomalía aquello alejado de un comportamiento normal.

Mientras que la detección de estas amenazas es un proceso reactivo, CTH es una práctica cuyo objetivo es identificar ciberamenazas de forma proactiva [89], para mejorar las medidas de seguridad y poder defenderse ante futuros ciberataques al integrarlo dentro de una propuesta más amplia, como es la de un entorno de conciencia cibersituacional presentada en esta Tesis Doctoral.

El concepto de CTH [21] se refiere a la búsqueda e identificación de ciberataques y ciberamenazas que no se han podido identificar mediante técnicas tradicionales como cortafuegos o antivirus. Esta práctica supone la detección de signos que indiquen APTs, para lo que se requiere acceso a los dispositivos y redes.

En la Figura 3.1 se muestran los procesos que permiten la caracterización de ciberataques. En este tipo de entornos se espera localizar técnicas de ataque en registros de tráfico a partir de una investigación. La información recopilada es analizada en conjunto con los resultados de la detección de intrusiones para encontrar la respuesta más adecuada ante estos ataques.

**Figura 3.1:** Caracterización de ciberataques

Esta caracterización de ciberamenazas se puede enmarcar en el contexto de la cibercontrainteligencia al integrarlo en los planes de ciberseguridad, enfocándose siempre en obtener información sobre el incidente para poder mitigar o neutralizar las amenazas inmediatas. Además, se puede entender este concepto dentro del análisis y monitorización de ciberamenazas, a nivel de gestión, ya que los datos recopilados para entender al atacante y sus TTPs permiten llevar a cabo un soporte a la toma de decisiones informadas.

La respuesta automática [31] es la capacidad de detectar una amenaza y responder a ella sin la participación de un usuario o administrador. Además, con la rápida expansión de las amenazas, la habilidad de reaccionar de forma automática es vital para reducir las consecuencias de los ciberataques. Sin embargo, entre los retos actuales de estos sistemas se encuentra su correcta configuración, ya que en caso de existir algún error, las respuestas pueden agravar el efecto de los incidentes.

Al detectar un comportamiento fuera de lo normal en el tráfico, el sistema puede analizar las medidas a realizar para protegerse. Esta defensa, además de ser automática, si se complementa con herramientas de IA, debe adaptarse a cada escenario y situación de amenaza. Ante la detección de cualquier anomalía, el soporte a la toma de decisiones puede recomendar en tiempo real medidas a ejecutar para mitigar su efecto o evitar que los ciberataques sigan avanzando.

3.3 Sistemas de Detección de Intrusiones

Un ciberataque [86] se define como un conjunto de eventos que comprometen los principios de los sistemas informáticos. Los métodos de seguridad más tradicionales se basan en información de ataques conocidos, pero con el desarrollo de la IA, los incidentes han ido ganando complejidad, hasta el punto de que los cortafuegos e IDS básicos no son capaces de identificar y eliminar los nuevos ataques, y deben complementarse con técnicas de aprendizaje automático para ofrecer una mayor protección sobre la infraestructura de una organización. Así, los ataques de día cero, que no se han visto anteriormente y que no se pueden comparar con la base de datos de ataques conocidos, se pueden identificar a partir de la definición de un comportamiento normal, y al detectar actividad que se aleja de esta definición, se genera una alarma.

A la hora de definir anomalías no basta con asumir que son una muestra que difiere de las demás. Especialmente cuanto más amplio es un conjunto de datos y más atributos contiene, es difícil establecer hasta qué punto la diferencia implica una anomalía [76].

La detección de intrusiones es una tarea adecuada para el aprendizaje automático desde dos posibles enfoques. Por un lado, si las anomalías son eventos raros, a partir de modelos estadísticos se pueden aprender los patrones del comportamiento normal y tratar el resto como anomalía. En algunos entornos, se encuentra un problema al entrenar con datos normales únicamente, ya que el modelo tiene dificultad para diferenciar los datos que no lo son, o en problemas de clasificación multi-etiqueta, donde la clase ‘anómala’ está compuesta de varias sub-clases. En contraposición se encuentran las técnicas de aprendizaje profundo, en especial la reconstrucción de errores para identificar anomalías, que construyen un *auto-encoder* para aprender de los datos el comportamiento normal. Ambas aproximaciones, además, deben lidiar con la falta de datos de entrenamiento y validación adecuados.

Según el enfoque que aplican a la detección de intrusiones, los IDS se clasifican en dos tipos [93]:

- IDS de Red (*Network IDS* o NIDS): son sistemas autónomos presentes en la misma red que monitorizan en busca de ciberataques, basándose principalmente en dos principios:
 - Firma: se detectan intrusiones al comparar la actividad con otros ataques conocidos, no siendo efectivo para detectar ataques de día cero.
 - Anomalía: se define un perfil básico que modela el comportamiento normal de la red, y a partir de ahí, cualquier desviación se considera un ciberataque.

Sin embargo, no pueden acceder al estado interno de los sistemas, lo que complica la detección en algunos casos. Además, las redes encriptadas, que cada vez son más comunes, también impiden a este tipo de IDS realizar su función correctamente

- IDS de *Host* (HIDS): son componentes software que se instalan en los dispositivos de la red que monitorizan, que suelen tener acceso a Internet o a una red interna. En este caso, recopilan mucha información del contexto, que permite entender mejor los procesos y actividades, volviéndose dispositivos más complejos que los NIDS.

Los NIDS de anomalía permiten identificar ataques tanto conocidos como de día cero, por lo que, en general, son los más recomendados. Sin embargo, la ventaja de los HIDS radica en que son capaces de identificar comportamiento anómalo dentro de los sistemas de una organización, evitando que el ataque se propague. En ambos casos, su efectividad se estima según la capacidad de identificar ciberataques a partir de datos normales y anómalos.

Los principales retos de los IDS a día de hoy consisten en lidiar con datos de entrenamiento desequilibrados y desactualizados, que no se adaptan a los ataques actuales. Además, deben ser capaces de identificar los ataques de día cero sin obtener tasas altas de falsos negativos y positivos, indicando que el modelo únicamente genera alertas cuando es necesario.

3.4 Modelos de inteligencia de amenazas de MITRE

MITRE [84] es una compañía que trabaja en avances de seguridad como asesor independiente relacionado con distintas materias: aeroespacial, inteligencia artificial, aviación y transporte, ciberseguridad, defensa e inteligencia, innovación gubernamental, salud, seguridad nacional y telecomunicaciones.

En el ámbito en el que se enmarca esta Tesis Doctoral, la ciberseguridad, la organización establece soluciones innovadoras para los retos que surgen con el avance de las tecnologías. Para ello, a lo largo de sus años de experiencia ha desarrollado marcos como ATT&CK, Engage, D3FEND o CALDERA, que proporcionan información esencial para analizar las intrusiones y perfeccionar las defensas frente a futuros ataques y vulnerabilidades.

MITRE también ha ido construyendo catálogos fundamentales para la caracterización de ciberataques y vulnerabilidades en activos, como *Common Weakness Enumeration* (CWE), *Common Vulnerabilities and Exposures* (CVE), *Common Attack Pattern Enumeration and Classification* (CAPEC) y ATT&CK. Estas dos últimas enfocan la caracterización de los ciberataques desde el comportamiento del atacante, realizando dos aproximaciones distintas que se detallan a continuación.

3.4.1 MITRE ATT&CK

En 2013, MITRE creó ATT&CK [85], centrado en la defensa de la red, para describir las fases operacionales del ciclo de vida de un adversario, tanto antes como después del ataque, detallando TTPs que se utilizan en APTs para lograr los objetivos del atacante. Esta base de conocimiento de tácticas y técnicas está fundamentada sobre observaciones reales del comportamiento de los atacantes y una taxonomía de posibles acciones a llevar a cabo.

En el contexto de la caracterización de ciberataques, ATT&CK permite desarrollar modelos de amenazas aplicables a distintas metodologías basados en un lenguaje común.

Se divide en dos partes, *Mobile*, que describe el comportamiento de los atacantes contra dispositivos móviles; y *Enterprise*, centrado en el comportamiento contra redes empresariales y nubes.

Las tácticas representan el “porqué” de una técnica o sub-técnica, el objetivo táctico del adversario, el motivo para realizar una acción. Por otro lado, las técnicas representan “cómo” el atacante consigue el objetivo táctico al realizar la acción. Las sub-técnicas son descripciones más específicas del comportamiento del adversario. Por último, los procedimientos son implementaciones concretas que un atacante usa para una técnica o sub-técnica, que no se incluyen en el ámbito de la propuesta.

La lista de tácticas de ATT&CK *Enterprise* está compuesta por:

- *Initial Access* (TA0001): El atacante trata de acceder a la red. Sus diez técnicas usan distintos vectores de entrada para conseguir su punto de apoyo dentro de la red.
- *Execution* (TA0002): El adversario intenta ejecutar código malicioso de forma local o remota. Son catorce técnicas que normalmente suelen acompañar otras tácticas que

busquen resultados más amplios.

- *Persistence* (TA0003): El atacante busca mantener su punto de acceso en casos de reinicio o cambios de credenciales a través de alguna de las veinte técnicas descritas.
- *Privilege Escalation* (TA0004): El adversario se enfoca en conseguir permisos de mayor nivel en una red o sistema. Puede acceder a una red sin permisos con las catorce técnicas incluidas pero para conseguir sus objetivos se requieren perfiles con permisos elevados.
- *Defense Evasion* (TA0005): El atacante trata de evitar ser detectado. Es el grupo más numeroso, con cuarenta y tres técnicas.
- *Credential Access* (TA0006): El adversario intenta obtener nombres de usuario y contraseñas mediante diecisiete técnicas.
- *Discovery* (TA0007): El atacante busca de conocer el entorno (sistema y red interna), lo que le permita orientarse antes de decidir cómo actuar, cómo podría entrar o qué puede controlar. Existen treinta y dos técnicas agrupadas en esta táctica.
- *Lateral Movement* (TA0008): El adversario se enfoca en moverse a través del entorno. Estas nueve técnicas buscan entrar y controlar sistemas remotos de la red.
- *Collection* (TA0009): El atacante trata de recoger información de interés para su objetivo. Son diecisiete técnicas que normalmente, se concatenan con una extracción de datos del sistema.
- *Exfiltration* (TA0010): El atacante intenta robar datos de la red a través de nueve técnicas distintas.
- *Command and Control* (TA0011): El adversario busca comunicarse con los sistemas comprometidos dentro de la red víctima y controlarlos utilizando alguna de las diecisiete técnicas clasificadas dentro.
- *Impact* (TA0040): El adversario se enfoca en manipular, interrumpir o destruir el sistema o los datos. Estas catorce técnicas buscan interrumpir la disponibilidad o comprometer la integridad manipulando procesos operacionales.
- *Resource Development* (TA0042): El adversario trata de establecer recursos que den soporte a sus operaciones. Estas ocho técnicas implican crear o comprometer recursos que apoyen sus objetivos.
- *Reconnaissance* (TA0043): El atacante intenta recoger información que usará en futuras operaciones. Las diez técnicas de este tipo implican acciones que activa o pasivamente permitan adaptar el ciberataque al objetivo.

En total ATT&CK *Enterprise* incluye una lista de 201 técnicas y 424 sub-técnicas. Debido a ello, en el modelo de caracterización propuesto en esta Tesis Doctoral se ha tenido que limitar la detección a las veintidós siguientes, presentes en el conjunto de datos elegido para trabajar. Sobre ellas, ATT&CK incluye información entre la que se encuentran las sub-técnicas, si existen; la táctica a la que se asocian; el procedimiento de detección y las posibles mitigaciones, que se listarán más adelante. Las técnicas seleccionadas son:

- *Network Service Discovery* (T1046): Los atacantes obtienen una lista de los servicios corriendo en equipos remotos o en dispositivos locales de la red.
 - Sub-técnicas: Ninguna.
 - Tácticas: *Discovery*.
 - Detección: a través del análisis de los servicios de la nube, comandos o tráfico de red.
 - Mitigaciones: M1042, M1031, M1030.
- *Command and Scripting Interpreter* (T1059): Los adversarios abusan de los intérpretes de comandos y *scripts* para ejecutar otros comandos o *scripts*.
 - Sub-técnicas: Nueve (T1059.001-T1059.009).
 - Tácticas: *Execution*.
 - Detección: a través del análisis de comandos, módulos, procesos o *scripts*.
 - Mitigaciones: M1049, M1040, M1045, M1042, M1038, M1026, M1021.
- *Application Layer Protocol* (T1071): Los adversarios se comunican usando la pila de protocolos OSI para evitar la detección al mezclarse con el resto del tráfico.
 - Sub-técnicas: Cuatro (T1071.001-T1071.004).
 - Tácticas: *Command and Control*.
 - Detección: a través del análisis de tráfico de red.
 - Mitigaciones: M1031.
- *Modify Registry* (T1112): Los atacantes interactúan con el registro Windows para ocultar información de configuración en las claves del registro o eliminar información.
 - Sub-técnicas: Ninguna.
 - Tácticas: *Defense Evasion*.
 - Defensas evitadas: Análisis forense del dispositivo.
 - Detección: a través del análisis de comandos, tráfico de red, procesos o el registro de Windows.
 - Mitigaciones: M1024.
- *External Remote Services* (T1133): Los adversarios aprovechan los servicios remotos externos para acceder a una red o permanecer en ella.
 - Sub-técnicas: Ninguna.
 - Tácticas: *Persistence e Initial Access*.
 - Detección: a través del análisis de registros de aplicación, inicio de sesión o tráfico de red.

- Mitigaciones: M1032, M1030, M1035, M1042.
- *Create Account* (T1136): Los atacantes crean una cuenta para mantener el acceso al sistema víctima.
 - Sub-técnicas: Tres (T1136.001-T1136.003).
 - Tácticas: *Persistence*.
 - Detección: a través del análisis de comandos, procesos o cuentas de usuario.
 - Mitigaciones: M1032, M1030, M1028, M1026.
- *Exploit Public-Facing Application* (T1190): Los atacantes pueden explotar debilidades en equipos con acceso a Internet para acceder a una red.
 - Sub-técnicas: Ninguna.
 - Tácticas: *Initial Access*.
 - Detección: a través del análisis de registros de aplicación y tráfico de red.
 - Mitigaciones: M1048, M1050, M1030, M1026, M1051, M1016.
- *Exploitation for Client Execution* (T1203): Los adversarios explotan vulnerabilidades de las aplicaciones para ejecutar código.
 - Sub-técnicas: Ninguna.
 - Tácticas: *Execution*.
 - Requisitos del sistema: servicio remoto accesible a través de la red.
 - Detección: a través del análisis de registros de aplicación y procesos.
 - Mitigaciones: M1048, M1050.
- *User Execution* (T1204): Cualquier adversario recurre a acciones específicas del usuario para conseguir la ejecución.
 - Sub-técnicas: Tres (T1204.001-T1204.003).
 - Tácticas: *Execution*.
 - Detección: a través del análisis de registros de aplicación, comandos, contenedores, archivos, imágenes, instancias, tráfico de red o procesos.
 - Mitigaciones: M1040, M1038, M1031, M1021, M1017.
- *Exploitation of Remote Services* (T1210): Los atacantes explotan servicios remotos para conseguir acceso no autorizado a sistemas internos dentro de una red.
 - Sub-técnicas: Ninguna.
 - Tácticas: *Lateral Movement*.
 - Permisos Requeridos: Usuario.

- Requisitos del sistema: software sin parches de seguridad o vulnerable. En algunos casos debe ser accesible remotamente a través de la red interna.
- Detección: a través del análisis de registros de aplicaciones o tráfico de red.
- Mitigaciones: M1048, M1042, M1050, M1030, M1026, M1019, M1051, M1016.
- *Server Software Component* (T1505): Los adversarios abusan de las características de los servidores para establecer un acceso persistente a los sistemas.
 - Sub-técnicas: Cinco (T1505.001-T1505.005).
 - Tácticas: *Persistence*.
 - Detección: a través del análisis de registros de aplicación, archivos, tráfico de red o procesos.
 - Mitigaciones: M1047, M1045, M1042, M1026, M1024, M1018.
- *Event Triggered Execution* (T1546): Los atacantes pueden establecer persistencia o elevar privilegios utilizando mecanismos del sistema que desencadenan la ejecución basándose en eventos específicos.
 - Sub-técnicas: Dieciséis (T1546.001-T1546.016).
 - Tácticas: *Privilege Escalation* y *Persistence*.
 - Detección: a través del análisis de servicios de nube, comandos, archivos, módulos, procesos, registro de Windows o WMI.
 - Mitigaciones: no se puede mitigar con controles preventivos, ya que se basa en abusos de las características del sistema.
- *Boot or Logon Autostart Execution* (T1547): Los adversarios pueden configurar un sistema para ejecutar un programa al inicio.
 - Sub-técnicas: Catorce (T1547.001-T1547.010, T1547.012-T1547.015).
 - Tácticas: *Privilege Escalation* y *Persistence*.
 - Permisos Requeridos: Administrador, Usuario, *root*.
 - Detección: a través del análisis de comandos, *drivers*, archivos, *kernel*, módulos, procesos o el registro de Windows.
 - Mitigaciones: no se puede mitigar con controles preventivos, ya que se basa en abusos de las características del sistema.
- *Abuse Elevation Control Mechanism* (T1548): Los adversarios pueden evitar los mecanismos de control de la elevación de privilegios.
 - Sub-técnicas: Cinco (T1548.001-T1548.005).
 - Tácticas: *Privilege Escalation* y *Defense Evasion*.
 - Permisos Requeridos: Administrador, Usuario.

- Detección: a través del análisis de comandos, archivos, procesos, cuentas de usuario o el registro de Windows.
- Mitigaciones: M1047, M1038, M1028, M1026, M1022, M1052, M1018.
- *Adversary-in-the-Middle* (T1557): Los atacantes se colocan entre dos o más dispositivos de una red para llevar a cabo comportamientos como *sniffing* o manipulación de datos.
 - Sub-técnicas: Tres (T1557.001-T1557.003).
 - Tácticas: *Credential Access* y *Collection*.
 - Detección: a través del análisis registros de aplicaciones, tráfico de red, servicios o el registro de Windows.
 - Mitigaciones: M1042, M1041, M1037, M1035, M1031, M1030, M1017.
- *Phishing* (T1566): Los adversarios envían mensajes engañosos y falsos para conseguir acceso al sistema víctima.
 - Sub-técnicas: Cuatro (T1566.001-T1566.004).
 - Tácticas: *Initial Access*.
 - Detección: a través del análisis de registros de aplicación, archivos o tráfico de red.
 - Mitigaciones: M1049, M1031, M1021, M1054, M1017.
- *Non-Standard Port* (T1571): los atacantes utilizan un protocolo y un puerto que, por defecto, no están asociados.
 - Sub-técnicas: Ninguna.
 - Tácticas: *Command and Control*.
 - Detección: a través de tráfico de red.
 - Mitigaciones: M1030, M1031.
- *Develop Capabilities* (T1587): El adversario construye sus capacidades.
 - Sub-técnicas: Cuatro (T1587.001-T1587.004).
 - Tácticas: *Resource Development*.
 - Detección: a través del escaneo de Internet o repositorios de *malware*.
 - Mitigaciones: M1056.
- *Gather Victim Identity Information* (T1589): Los adversarios recogen información de la identidad de la víctima.
 - Sub-técnicas: Tres (T1589.001-T1589.003).
 - Tácticas: *Reconnaissance*.
 - Detección: a través del análisis del tráfico de la red.

- Mitigaciones: M1056.
- *Gather Victim Network Information* (T1590): Los atacantes recogen información de la red víctima.
 - Sub-técnicas: Seis (T1590.001-T1590.006).
 - Tácticas: *Reconnaissance*.
 - Detección: está asociado a falsos positivos, ya que estas actividades son comunes, siendo difícil de detectar en esta fase. La detección se debe llevar a cabo en procedimientos de *Initial Access*.
 - Mitigaciones: M1056.
- *Gather Victim Host Information* (T1592): Los adversarios recogen información del dispositivo de la víctima.
 - Sub-técnicas: Cuatro (T1592.001-T1592.004).
 - Tácticas: *Reconnaissance*.
 - Detección: a través del escaneo de Internet.
 - Mitigaciones: M1056.
- *Active Scanning* (T1595): El atacante ejecuta escaneos activos de reconocimiento para obtener información.
 - Sub-técnicas: Tres (T1595.001-T1595.003).
 - Tácticas: *Reconnaissance*.
 - Detección: a través del análisis del tráfico de la red.
 - Mitigaciones: M1056.

Otra parte fundamental del marco ATT&CK, como se ha visto, es la capacidad de asociar TTPs con un conjunto de mitigaciones para hacerles frente y proteger los sistemas de los ciberataques. En total, cuenta con un catálogo de cuarenta y tres mitigaciones que representan conceptos y tipos de tecnología adecuados para prevenir una técnica o sub-técnica de ser ejecutada con éxito. A continuación se describen aquellas que tienen relación con las técnicas presentadas:

- M1016 - *Vulnerability Scanning*: Encontrar vulnerabilidades potencialmente explotables en el software para poder remediarlas.
- M1017 - *User Training*: Entrenar a los usuarios para que identifiquen intentos de acceso o manipulación, reduciendo el riesgo de sufrir ataques de ingeniería social como el *phishing* u otras técnicas que implican la interacción del usuario.
- M1018 - *User Account Management*: Gestionar la creación, modificación, uso y permisos asociados a las cuentas de usuario.
- M1019 - *Threat Intelligence Program*: Los programas de inteligencia de amenazas ayudan

a que las organizaciones generen su propia información y sigan tendencias para informar sobre las prioridades defensivas a la hora de mitigar el riesgo.

- M1021 - *Restrict Web-Based Content*: Restringir el uso de determinados sitios web, bloquear descargas y adjuntos, extensiones de navegador o JavaScript.
- M1022 - *Restrict File and Directory Permissions*: Restringir el acceso definiendo permisos de directorio y archivo no específicos a usuarios o cuentas privilegiadas.
- M1024 - *Restrict Registry Permissions*: Restringir la capacidad de modificar ciertas claves en el registro Windows.
- M1026 - *Privileged Account Management*: Gestionar la creación, modificación, uso y permisos asociados a cuentas con privilegios, como SYSTEM y root.
- M1028 - *Operating System Configuration*: Realizar cambios en la configuración en relación con el sistema operativo o una característica común suya para protegerlo frente a técnicas.
- M1030 - *Network Segmentation*: Los arquitectos dividen la red para aislar sistemas críticos, funciones o recursos. La segmentación física y lógica previene del acceso a sistemas e información sensibles. Mediante el uso de un DMZ se contienen los servicios de Internet que no se deberían exponerse desde la red interna. Configurar nubes privadas virtuales para aislar sistemas *cloud* críticos.
- M1031 - *Network Intrusion Prevention*: Utilizar firmas en la detección de intrusiones para bloquear tráfico de red.
- M1032 - *Multi-factor Authentication*: Utilizar dos o más pruebas para autenticarse en un sistema.
- M1035 - *Limit Access to Resource Over Network*: Prevenir los accesos a compartición de archivos y acceso remoto a los sistemas y servicios innecesarios.
- M1037 - *Filter Network Traffic*: Utilizar aplicaciones de red para filtrar el tráfico entrante y saliente, aplicando controles en base al protocolo.
- M1038 - *Execution Prevention*: Bloquear ejecuciones de código en un sistema mediante control de aplicaciones y bloqueo de *scripts*.
- M1040 - *Behavior Prevention on Endpoint*: Utiliza capacidades para prevenir patrones sospechosos de que ocurran en los sistemas finales.
- M1041 - *Encrypt Sensitive Information*: Proteger información sensible con codificación robusta.
- M1042 - *Disable or Remove Feature or Program*: Eliminar o denegar el acceso a software innecesario y potencialmente vulnerable.
- M1045 - *Code Signing*: Mejorar la integridad de aplicación con verificación de firma digital para evitar que se ejecute código no fiable.
- M1047 - *Audit*: Llevar a cabo auditorías o escáneres de sistemas, permisos, software y

configuraciones no seguras para identificar vulnerabilidades potenciales.

- M1048 - *Application Isolation and Sandboxing*: Restringir la ejecución de código a un entorno virtual en un sistema final o en tránsito hacia él.
- M1049 - *Antivirus/Antimalware*: Utiliza firmas para detectar software malicioso.
- M1050 - *Exploit Protection*: Utilizar capacidades para detectar y bloquear condiciones que puedan conducir o ser indicativo de la explotación de software.
- M1051 - *Update Software*: Llevar a cabo actualizaciones regulares de software para reducir el riesgo de explotación.
- M1052 - *User Account Control*: Configura el control de cuentas de usuarios en Windows para mitigar el riesgo de que los adversarios obtengan acceso a un proceso con privilegios elevados.
- M1054 - *Software Configuration*: Implementar cambios en la configuración de software para mitigar los riesgos asociados a su forma de operar.
- M1056 - *Pre-Compromise*: Cualquier actividad de mitigación que ocurra antes de que un adversario consiga acceso inicial, como técnicas de *Reconnaissance* o *Resource Development*.

Para la caracterización de ciberataques y poder trabajar a partir de ellos en una gestión eficaz del riesgo, proponiendo contramedidas y mitigaciones para hacerles frente, es imprescindible conocer los métodos que utilizan los atacantes para llevarlos a cabo. A partir de los datos obtenidos de ATT&CK, una vez que se identifica la técnica o técnicas utilizadas, es posible ofrecer una respuesta adaptada al incidente.

3.4.2 MITRE CAPEC

CAPEC [28] proporciona un catálogo público de patrones de ataque comunes con el objetivo de explicar cómo los atacantes explotan las vulnerabilidades de los activos. Se centra en la seguridad de la aplicación, describiendo atributos y técnicas utilizadas para explotar debilidades conocidas.

Un patrón de ataque es una descripción de los atributos y aproximaciones comunes para explotar una vulnerabilidad, incluyendo los retos a los que deben hacer frente y cómo se solucionan. Cada patrón de ataque captura el conocimiento relacionado con el diseño y ejecución de una parte específica de un ciberataque, proporcionando guías para disminuir su efectividad.

Los patrones que pertenecen al listado CAPEC [29] se relacionan con las listas de CWEs y CVEs para definir las vulnerabilidades explotadas. Mediante los CVEs se conocen instancias específicas de una debilidad (CWE) que se ha podido explotar, y a través de los datos de CAPEC se sabe cómo se ha explotado la debilidad (CWE). De esta forma, se podría entender CAPEC como un método para compensar un CWE al ejecutar un ataque.

Las entradas CAPEC normalmente contienen un flujo de ejecución que define los pasos del atacante para llevar a cabo una intrusión. Además tienen asociado un identificador numérico,

una descripción del patrón de ataque, la lista de debilidades relacionadas y de las consecuencias en el caso de que el patrón se materialice en la explotación de una debilidad, y, finalmente, un conjunto de ejemplos de estos patrones de ataque.

Los CAPECs se puede organizar en función del mecanismo de ataque, según el ámbito de la ciberseguridad que se ataca; o mediante el dominio del ataque, que agrupa ciberataques con métodos similares. Dentro de la información de cada CAPEC aparecen relaciones con otras entradas, como patrones más generales o específicos, o aquellos con los que se continuaría para completar un ataque o una explotación de una debilidad.

Muchos de los patrones de ataque definidos en CAPEC se ejecutan mediante técnicas ATT&CK específicas. Al unir ambas perspectivas se obtiene un conocimiento contextual del patrón de ataque en el ciclo de vida de un atacante. Del total de CAPECs disponibles en la página oficial, en 2024 existen 177 relacionados directamente con técnicas ATT&CK [30]. A continuación se detallan las relacionadas con las técnicas presentadas en el apartado anterior:

- **CAPEC-19: *Embedding Scripts within Scripts***: El atacante aprovecha la capacidad de ejecutar su propio *script* insertando otros *scripts* que el software ejecutará debido a alguna vulnerabilidad.
 - Probabilidad: Alta.
 - Severidad: Muy Alta.
 - Dominio de Ataque: Software.
 - Mecanismo de ataque: Inyectar elementos inesperados.
 - Flujo de ejecución:
 - * Explorar: Registrar todos los puntos de entrada en los que aparece un *script* en el lado del cliente.
 - * Experimentar: Sondear los posibles puntos de entrada para la vulnerabilidad XSS.
 - * Explotar: Robar credenciales, identificación y contenido de la página. Navegación Forzosa. Suplantación de contenido.
 - Pre-requisitos: Software objetivo debe poder ejecutar *scripts* y proporcionar al atacante privilegios de lectura/escritura.
 - Habilidades requeridas: Nivel Medio-Bajo. Ejecución de *scripts* remotos y recogida de la salida. Cargar *script* malicioso para abrir un directorio.
 - Consecuencias: Confidencialidad, Integridad y Disponibilidad (Ejecución de comandos no autorizados); Confidencialidad, Control de Acceso y Autorización (Obtención de privilegios).
 - Mitigaciones: Utilizar navegadores que no permitan *scripts* en el lado del cliente. Asegurar el contenido entregado al cliente. Validar todo el contenido remoto de entrada y salida. Desactivar lenguajes de *scripting* como JavaScript en el navegador.

- Implementar *tokens* de sesión. Mantener el software actualizado.
- CWEs Relacionados: CWE-284.
 - TTPs identificados: T1027.009, **T1546.004**, **T1546.016**.
 - **CAPEC-94: *Adversary in the Middle* (AiTM)**: El atacante amenaza la comunicación entre dos componentes y altera u obtiene datos de las transacciones.
 - Probabilidad: Alta.
 - Severidad: Muy Alta.
 - Dominio de Ataque: Software, Comunicaciones.
 - Mecanismo de ataque: Debilitar el control de acceso.
 - Flujo de ejecución:
 - * Explorar: Determinar el mecanismo de comunicación.
 - * Experimentar: Colocarse entre los objetivos.
 - * Explotar: Utilizar de forma malintencionada los datos interceptados.
 - Pre-requisitos: Existencia de dos componentes que se comunican, el atacante es capaz de identificar el mecanismo de comunicación y escucharla. Los objetivos no utilizan autenticación fuerte y la comunicación no está encriptada.
 - Habilidades requeridas: Nivel Medio. El ataque podría complicarse con la utilización de criptografía.
 - Consecuencias: Integridad (modificación de datos); Confidencialidad (lectura de los datos); Confidencialidad, Control de Acceso y Autorización (Conseguir privilegios).
 - Mitigaciones: Asegurar que las claves públicas se firman por una autoridad certificada, encriptar la comunicación, utilizar autenticación fuerte e intercambiar claves públicas a través de un canal seguro.
 - CWEs Relacionados: CWE-300, CWE-290, CWE-593, CWE-287, CWE-294.
 - TTPs identificados: **T1557**.
 - **CAPEC-98: *Phising***: Técnica de ingeniería social donde el atacante suplanta una entidad legítima para engañar a la víctima para que revele información confidencial.
 - Probabilidad: Alta.
 - Severidad: Muy Alta.
 - Dominio de Ataque: Ingeniería Social.
 - Mecanismo de ataque: Participar en actividades engañosas.
 - Flujo de ejecución:
 - * Explorar: Obtener nombre de dominio y certificados para suplantar un sitio

web o explorar páginas legítimas y crear un duplicado.

- * Explotar: Convencer al usuario de introducir información sensible en el sitio web del adversario y utilizar estos credenciales robados para acceder en el sitio legítimo.
- Pre-requisitos: Forma de contactar con la víctima, acertar la entidad de la víctima y suplantarla, encontrar un motivo suficiente para que la víctima actúe, replicar la página original de manera casi exacta, incluyendo la URL de la entidad.
- Habilidades requeridas: Nivel Medio. Conocimiento básico de páginas web.
- Recursos requeridos: Herramientas de desarrollo de páginas web.
- Consecuencias: Integridad (modificación de datos); Confidencialidad (lectura de los datos); Confidencialidad, Control de Acceso y Autorización (Conseguir privilegios).
- Mitigaciones: No seguir ningún enlace recibido por correo electrónico y no introducir credenciales en la página ni responder a los correos que soliciten información sensible.
- CWEs Relacionados: CWE-451.
- TTPs identificados: **T1566**, T1598.
- **CAPEC-114: Authentication Abuse:** El atacante obtiene acceso no autorizado a un activo a partir de conocimiento de una debilidad del mecanismo de autenticación o explotando un defecto en la implementación. En este tipo de ataques, el mecanismo de autenticación funciona, pero una secuencia de eventos provoca que proporcione acceso al atacante.
 - Severidad: Media.
 - Dominio de Ataque: Software, Hardware.
 - Mecanismo de ataque: Debilitar el control de acceso.
 - Pre-requisitos: Un mecanismo de autenticación a base de contraseñas, certificados de seguridad, etc. con defectos
 - Recursos requeridos: Aplicación cliente, línea de comandos o lenguaje de programación capaz de interactuar con el mecanismo de autenticación.
 - CWEs Relacionados: CWE-287, CWE-1244.
 - TTPs identificados: **T1548**.
- **CAPEC-115: Authentication Bypass:** El atacante consigue acceder a un activo con los privilegios de un usuario evadiendo el mecanismo de autenticación. Puede acceder a datos protegidos sin haberse autenticado.
 - Severidad: Media.
 - Dominio de Ataque: Software.

- Mecanismo de ataque: Debilitar el control de acceso.
 - Pre-requisitos: Mecanismo de autenticación a base de contraseñas, certificados de seguridad, etc.
 - Recursos requeridos: Aplicación cliente, como un buscador o un lenguaje de programación que interactúe con el sistema.
 - CWEs Relacionados: CWE-287.
 - TTPs identificados: **T1548**.
- **CAPEC-122: Privilege Abuse:** El adversario es capaz de explotar características del objetivo reservadas para administradores, pero que son expuestas a cuentas con privilegios bajos.
 - Probabilidad: Alta.
 - Severidad: Media.
 - Dominio de Ataque: Software, Hardware.
 - Mecanismo de ataque: Debilitar el control de acceso.
 - Pre-requisitos: El objetivo debe tener los mecanismos de control de acceso mal configurados, de forma que los datos sensibles sean accesibles a usuarios que no deberían tener la capacidad. El adversario debe acceder al objetivo con una cuenta con pocos privilegios.
 - Habilidades requeridas: Nivel Bajo. Únicamente se requiere al atacante que cuente con una cuenta con privilegios limitados.
 - Recursos requeridos: Ninguno.
 - Consecuencias: Integridad (modificación de datos); Confidencialidad (lectura de los datos); Autorización (ejecución no autorizada de comandos y obtención de privilegios); Control de Acceso y Autorización (Sobrepasar mecanismos de protección).
 - Mitigaciones: Configurar los privilegios de forma que la funcionalidad del administrado no se exponga a cuentas no autorizadas.
 - CWEs Relacionados: CWE-269, CWE-732, CWE-1317.
 - TTPs identificados: **T1548**.
 - **CAPEC-132: Symlink Attack:** El atacante coloca un enlace simbólico de forma que el usuario o aplicación objetivo accede al *endpoint* del enlace, pensando que accede a un archivo con el mismo nombre.
 - Probabilidad: Baja.
 - Severidad: Alta.
 - Dominio de Ataque: Software.
 - Mecanismo de ataque: Participar en actividades engañosas.

- Flujo de ejecución:
 - * Explorar: Identificar el objetivo.
 - * Experimentar: Tratar de crear *symlinks* de distintos archivos.
 - * Explotar: La aplicación objetivo opera sobre el *symlink* de datos sensibles creado.
 - Pre-requisitos: La aplicación objetivo debe llevar a cabo actividades sobre un archivo sin comprobar si es un enlace simbólico o no. El atacante debe predecir el nombre del archivo que modifica y crear este *symlink*.
 - Habilidades requeridas: Nivel Bajo - Crear *symlinks*. Nivel Alto - Identificar archivos y crear el enlace en la ventana de operación.
 - Recursos requeridos: Ninguno.
 - Consecuencias: Integridad (modificación de datos); Confidencialidad (lectura de los datos); Autorización (ejecución no autorizada de comandos); *Accountability*, Autenticación, Autorización y No repudio (obtención de privilegios); Control de Acceso y Autorización (sobrepassar mecanismos de protección); Disponibilidad (ejecución no segura).
 - Mitigaciones: en el diseño, comprobar si se lleva a cabo la creación de archivos y, en ese caso, verificar que se comprueba si es un *symlink*. En implementación, utilizar nombres aleatorios para los archivos temporales con permisos restringidos.
 - CWEs Relacionados: CWE-59.
 - TTPs identificados: **T1547.009**.
- **CAPEC-141: Cache Poisoning:** El atacante explota una funcionalidad de la caché que causa que ayude a los atacantes cuando éste coloca información o material dañino en la caché.
 - Probabilidad: Alta.
 - Severidad: Alta.
 - Dominio de Ataque: Software.
 - Mecanismo de ataque: Manipular recursos del sistema.
 - Flujo de ejecución:
 - * Explorar: Identificar y explorar las cachés.
 - * Experimentar: Provocar que se capturen datos específicos.
 - * Explotar: Redirigir a los usuarios hacia sitios web maliciosos.
 - Pre-requisitos: Capacidad para modificar valores almacenados en la caché. La aplicación objetivo no debe detectar modificaciones ilícitas y debe confiar en los valores de la caché para los cálculos.

- Habilidades requeridas: Nivel Medio. Modificar cachés.
 - Mitigaciones: en la configuración, eliminar la captura en el lado del cliente. Para la implementación, escuchar las respuestas de las consultas en la red y enviar notificación cuando cambia una entrada.
 - CWEs Relacionados: CWE-348, CWE-345, CWE-349, CWE-346.
 - TTPs identificados: **T1557.002**.
- **CAPEC-163: Spear Phishing:** El adversario ataca a un grupo o usuario específico con un ataque de *Phishing* (CAPEC-98) adaptado a su categoría para evitar que se detecte.
 - Probabilidad: Alta.
 - Severidad: Alta.
 - Dominio de Ataque: Ingeniería Social.
 - Mecanismo de ataque: Participar en actividades engañosas.
 - Flujo de ejecución:
 - * Explorar: Obtener información útil del contexto del usuario u organización objetivo.
 - * Experimentar: Existen tres ejecuciones opcionales, que consisten en obtener nombres de dominio y certificados para suplantar un sitio legítimo, explorar páginas web legítimas y crear un duplicado, o construir variantes de la página web con información específica del usuario.
 - * Explotar: Convencer al usuario de introducir información sensible en el sitio web del adversario y utilizar estos credenciales robados para acceder en el sitio legítimo.
 - Pre-requisitos: Ninguno.
 - Habilidades requeridas: Nivel Medio. Requieren información específica de la víctima.
 - Recursos requeridos: Capacidad de comunicarse con la víctima y una plataforma para que el objetivo introduzca sus datos.
 - Consecuencias: Integridad (modificación de datos); Confidencialidad (lectura de los datos); *Accountability*, Autenticación, Autorización y No repudio (Obtención de privilegios).
 - Mitigaciones: No seguir ningún enlace recibido por correo electrónico y no introducir credenciales en la página ni responder a los correos que soliciten información sensible.
 - CWEs Relacionados: CWE-451.
 - TTPs identificados: T1534, **T1566.001**, **T1566.002**, **T1566.003**, T1598.001, T1598.002, T1598.003.

- **CAPEC-169: *Footprinting***: El adversario realiza actividades de exploración para identificar propiedades del sistema objetivo.
 - Probabilidad: Alta.
 - Severidad: Muy Baja.
 - Dominio de Ataque: Software, Comunicaciones.
 - Mecanismo de ataque: Recoger y analizar información.
 - Flujo de ejecución:
 - * Explorar: Analizar información de la página y código fuente, utilizando tareas automáticas para obtener tanta información como es posible sobre el sistema y la organización.
 - Pre-requisitos: La aplicación objetivo debe publicitar de manera voluntaria o involuntaria información del sistema.
 - Habilidades requeridas: Nivel Bajo. Envío de solicitudes HTTP y ejecución de la herramienta de escaneo.
 - Recursos requeridos: Herramientas para recoger información de la víctima. Escaneos de puertos y redes y analizadores de respuestas para determinar la versión e información de configuración.
 - Consecuencias: Confidencialidad (lectura de los datos).
 - Mitigaciones: Mantener el sistema actualizado, apagar puertos y servicios innecesarios, cambiar contraseñas por defecto, encriptar contraseñas para proteger datos sensibles, evitar incluir información que pueda afectar a la seguridad como planes de negocio o documentos propietarios.
 - CWEs Relacionados: CWE-200.
 - TTPs identificados: T1217, **T1592**, **T1595**.
- **CAPEC-203: *Manipulate Registry Information***: Un adversario explota una debilidad en autorización para modificar el contenido de un registro. Esto le permite ocultar información de configuración o eliminar indicadores de compromiso.
 - Severidad: Media.
 - Dominio de Ataque: Software.
 - Mecanismo de ataque: Manipular recursos del sistema.
 - Pre-requisitos: La aplicación objetivo debe depender de valores almacenados en un registro. El atacante requiere métodos para elevar los permisos y poder acceder y modificar los registros.
 - Habilidades requeridas: Nivel Alto. Se necesita credenciales privilegiadas o una herramienta a medida de acceso remoto.

- Recursos requeridos: Ninguno.
- Mitigaciones: Asegurar los permisos de los registros, emplear una postura defensiva robusta y por capas, implementar métodos de identificación robustos.
- CWEs Relacionados: CWE-15.
- TTPs identificados: **T1112**, T1647.
- **CAPEC-233: Privilege Escalation:** El atacante explota una debilidad que le permite elevar sus privilegios y ejecutar acciones que no tiene autorizadas.
 - Dominio de Ataque: Software, Hardware.
 - Mecanismo de ataque: Debilitar el control de acceso.
 - CWEs Relacionados: CWE-269, CWE-1264, CWE-1311.
 - TTPs identificados: **T1548**.
- **CAPEC-270: Modification of Registry Run Keys:** El atacante añade una nueva entrada en el registro de Windows para que una determinada aplicación se ejecute al iniciar sesión.
 - Probabilidad : Media.
 - Severidad: Media.
 - Dominio de Ataque: Software.
 - Mecanismo de ataque: Manipular recursos de sistema.
 - Flujo de ejecución:
 - * Explorar: Determinar el sistema objetivo (Windows).
 - * Experimentar: Conseguir acceso al sistema.
 - * Explotar: Modificar el registro de Windows.
 - Pre-requisitos: Acceso previo al sistema de forma física o lógica.
 - Consecuencias: Integridad (Modificación de datos y obtención de privilegios).
 - Mitigaciones: Identificar programas que se pueden utilizar para adquirir información de procesos y bloquearlos mediante una política de restricción de software.
 - CWEs Relacionados: CWE-15.
 - TTPs identificados: **T1547.001**, **T1547.014**.
- **CAPEC-300: Port Scanning:** El adversario utiliza una combinación de técnicas para determinar el estado de los puertos de un objetivo remoto. Cualquier servicio o aplicación compatible con TCP o UDP tendrá un puerto abierto para las comunicaciones sobre la red.
 - Severidad: Baja.

- Dominio de Ataque: Software, Comunicaciones.
 - Mecanismo de ataque: Recoger y analizar información.
 - Pre-requisitos: Acceso lógico a la red objetivo.
 - Recursos requeridos: Herramienta de escaneo de redes, inyección de paquetes o programación de *sockets*. Para ver la respuesta se necesitan *sniffers*.
 - Consecuencias: Confidencialidad, Control de Acceso y Autorización (eludir el mecanismo de protección, ocultar actividades).
 - CWEs Relacionados: CWE-200.
 - TTPs identificados: **T1046**.
- **CAPEC-309: Network Topology Mapping:** El adversario ejecuta actividades de escaneo para mapear los nodos, dispositivos y rutas de una red. Este tipo de reconocimiento se lleva a cabo en los primeros pasos de un ataque frente a una red externa
 - Severidad: Baja.
 - Dominio de Ataque: Software.
 - Mecanismo de ataque: Recoger y analizar información.
 - Pre-requisitos: Ninguno.
 - Recursos requeridos: La habilidad de enviar y recibir datos de un objetivo (*probing*) y entender el protocolo para analizar un canal de comunicación existente (escucha pasiva).
 - Consecuencias: Confidencialidad.
 - CWEs Relacionados: CWE-200.
 - TTPs identificados: T1016, T1049, **T1590**.
 - **CAPEC-407: Pretexting:** El atacante utiliza un pretexto para solicitar información a la víctima, o le manipula para conseguir que realice una acción que le beneficia. Crean un escenario inventado en el que el rol del adversario es convencer a la víctima para que revele información o lleve a cabo alguna acción
 - Probabilidad: Media.
 - Severidad: Baja.
 - Dominio de Ataque: Software, Ingeniería Social.
 - Mecanismo de ataque: Participar en actividades engañosas, Recoger y analizar información.
 - Pre-requisitos: El atacante debe poseer los medios y el conocimiento para comunicarse con la víctima y ofrecerle un pretexto que influencie las acciones de un objetivo específico.

- Habilidades requeridas: Nivel Bajo. Habilidades de comunicación e interpersonales.
- Consecuencias: Confidencialidad.
- Mitigaciones: Entrenamiento de ciberseguridad para evitar ataques de ingeniería social.
- CWEs Relacionados: Ninguno.
- TTPs identificados: **T1589**.
- **CAPEC-541: Application Fingerprinting:** El atacante realiza actividades de recogida de huellas para determinar el tipo o versión de una aplicación del sistema remoto.
 - Severidad: Baja.
 - Dominio de Ataque: Software.
 - Mecanismo de ataque: Recoger y analizar información.
 - Pre-requisitos: Ninguno
 - CWEs Relacionados: CWE-204, CWE-205, CWE-208.
 - TTPs identificados: **T1592.002**.
- **CAPEC-542: Targeted Malware:** El adversario desarrolla *malware* que aprovecha una vulnerabilidad conocida a partir de información recogida de la tecnología del entorno.
 - Dominio de Ataque: Software.
 - Mecanismo de ataque: Inyectar elementos inesperados.
 - CWEs Relacionados: Este es un caso excepcional, donde un CAPEC crea un CWE, al contrario que en la mayoría de los casos.
 - TTPs identificados: **T1587.001**, T1027.
- **CAPEC-552: Install Rootkit:** El adversario explota una debilidad en la autenticación para instalar *malware* que altera la funcionalidad y la información proporcionada por la llamada a la API. Los *rootkits* se utilizan para ocultar la presencia de programas, archivos, conexiones de red, servicios y otros componentes del sistema.
 - Probabilidad: Media.
 - Severidad: Alta.
 - Dominio de Ataque: Software.
 - Mecanismo de ataque: Inyectar elementos inesperados.
 - Mitigaciones: Prevenir el acceso a cuentas privilegiadas para instalar *rootkits*.
 - CWEs Relacionados: CWE-284.
 - TTPs identificados: T1014, T1542.003, **T1547.006**.

- **CAPEC-555: *Remote Services with Stolen Credentials***: Este patrón involucra un atacante que usa credenciales robados para aprovechar servicios remotos y acceder al sistema.
 - Severidad: Muy Alta.
 - Dominio de Ataque: Software.
 - Mecanismo de ataque: Debilitar el control de acceso.
 - Mitigaciones: Desactivar servicios remotos como SSH y establecer reglas de cortafuegos que bloquen este tipo de tráfico. Limitar a los usuarios con acceso remoto. Eliminar el grupo de administradores locales de los permisos para acceder a RDP. Limitar los permisos de usuario. Utilizar autenticación multi-factor para los inicios de sesión remotos.
 - CWEs Relacionados: CWE-522, CWE-308, CWE-309, CWE-294, CWE-263, CWE-262, CWE-521.
 - TTPs identificados: T1021, T1114.002, **T1133**.
- **CAPEC-556: *Replace File Extension Handlers***: Cuando se abre un archivo, el buscador determina con qué aplicación se abre. Algunas aplicaciones pueden modificar los gestores de archivo a través de una extensión de archivo para utilizar un programa arbitrario al abrir ese archivo.
 - Dominio de Ataque: Software.
 - Mecanismo de ataque: Inyectar elementos inesperados.
 - Mitigaciones: Inspeccionar el registro en busca de cambios. Limitar los privilegios de las cuentas de usuario para cambiar los gestores de archivo con autorización de un administrador.
 - CWEs Relacionados: CWE-284.
 - TTPs identificados: **T1546.001**.
- **CAPEC-558: *Replace Trusted Executable***: El atacante explota una debilidad en la gestión de privilegios o el control de acceso para reemplazar un ejecutable por una versión maliciosa que permite la ejecución de otro *malware*.
 - Probabilidad: Baja.
 - Severidad: Alta.
 - Dominio de Ataque: Software.
 - Mecanismo de ataque: Inyectar elementos inesperados.
 - CWEs Relacionados: CWE-284.
 - TTPs identificados: **T1505.005**, **T1546.008**.
- **CAPEC-564: *Run Software at Logon***: Los sistemas permiten ejecutar *scripts* al

inicio. Si el atacante puede acceder a estos *scripts*, puede insertar código que le ayude a mantener la persistencia o moverse lateralmente.

- Dominio de Ataque: Software.
- Mecanismo de ataque: Inyectar elementos inesperados.
- Mitigaciones: Restringir el acceso de estructura a los *scripts* de inicio.
- CWEs Relacionados: CWE-284.
- TTPs identificados: T1037, T1543.001, T1543.004, **T1547**.
- **CAPEC-579: Replace Winlogon Helper DLL:** Winlogon es una parte de Windows que lleva a cabo acciones de inicio de sesión. Al modificar la clave de registro, se puede provocar que cargue un DLL al arrancar.
 - Dominio de Ataque: Software.
 - Mecanismo de ataque: Inyectar elementos inesperados.
 - Mitigaciones: Cambios en las entradas de registro que no correlacionan con software conocido son sospechosas.
 - CWEs Relacionados: CWE-15.
 - TTPs identificados: **T1547.004**.
- **CAPEC-640: Inclusion of Code in Existing Process:** El adversario aprovecha un error en la verificación de la integridad de un proceso en ejecución para ejecutar código arbitrario.
 - Probabilidad: Baja.
 - Severidad: Alta.
 - Dominio de Ataque: Software.
 - Mecanismo de ataque: Inyectar elementos inesperados.
 - Flujo de ejecución:
 - * Explorar: Determinar el proceso objetivo con suficientes privilegios para introducir el código.
 - * Experimentar: Probar a incluir código simple con salidas conocidas para comprobar si funciona.
 - * Explotar: Incluir código arbitrario en un proceso existente.
 - Pre-requisitos: La aplicación objetivo falla al verificar la integridad de un proceso en ejecución.
 - Habilidades requeridas: Nivel Alto. Requiere conocimientos para cargar el código malicioso en el espacio de memoria de un proceso activo y la habilidad para que el proceso ejecute el código.

- Consecuencias: Integridad (Ejecutar comandos no autorizados); Confidencialidad (Lectura de datos).
- Mitigaciones: Prevenir software malicioso de cargar a partir de una política de acceso, restringir la localización del software utilizado, aprovechar la seguridad de los módulos del kernel proporcionando control de acceso avanzado, monitorizar determinadas llamadas API y procesos o comandos.
- CWEs Relacionados: CWE-114, CWE-829.
- TTPs identificados: **T1505.005**, T1574.006, T1574.013, T1620.
- **CAPEC-642: Replace Binaries:** El atacante conoce algunos binarios que se ejecutan regularmente. Si no están protegidos correctamente, podría reemplazarlos con *malware*.
 - Severidad: Alta.
 - Dominio de Ataque: Software.
 - Mecanismo de ataque: Debilitar el control de acceso.
 - Pre-requisitos: Capacidad de colocar binario malicioso en la máquina objetivo.
 - Mitigaciones: Asegurar que los binarios frecuentemente utilizados tienen permisos correctos. Definir políticas de servicio que restringen la elevación de privilegios. Herramientas de auditoría.
 - CWEs Relacionados: CWE-732.
 - TTPs identificados: **T1505.005**, T1554, T1574.005.
- **CAPEC-650: Upload a Web Shell to a Web Server:** Al explotar permisos insuficientes, es posible subir una consola web a un servidor para ejecuciones remotas.
 - Severidad: Alta.
 - Dominio de Ataque: Software.
 - Mecanismo de ataque: Debilitar el control de acceso.
 - Pre-requisitos: El servidor web es susceptible a algún *exploit* de aplicaciones web que permite subir esta consola.
 - Consecuencias: Confidencialidad (Lectura de datos); Confidencialidad, Control de Acceso y Autorización (Obtención de privilegios); Confidencialidad, Integridad y Disponibilidad (Ejecutar comandos no autorizados).
 - Mitigaciones: Asegurar que el servidor web está actualizado y los permisos de los directorios del servidor donde se pueden ejecutar archivos.
 - CWEs Relacionados: CWE-287, CWE-553.
 - TTPs identificados: **T1505.003**.
- **CAPEC-654: Credential Prompt Impersonation:** El adversario, a partir de una aplicación maliciosa instalada previamente, suplanta una solicitud de credenciales para

conseguir la información de un usuario.

- Probabilidad: Media.
 - Severidad: Alta.
 - Dominio de Ataque: Software.
 - Mecanismo de ataque: Participar en actividades engañosas.
 - Flujo de ejecución:
 - * Explorar: Determinar tareas adecuadas en las que el usuario proporcione credenciales para explotar.
 - * Explotar: Suplantar la tarea legítima para conseguir credenciales.
 - Pre-requisitos: El adversario necesita tener acceso previo al sistema objetivo, donde debe existir una tarea legítima que pueda suplantar.
 - Habilidades requeridas: Nivel Bajo. Una vez conseguido el acceso al sistema, la suplantación no supone dificultad.
 - Recursos requeridos: *Malware* que comprometa el sistema inicialmente y otro *malware* para suplantar la solicitud de credenciales legítima.
 - Consecuencias: Control de Acceso y Autorización (Conseguir privilegios).
 - Mitigaciones: Evitar instalar aplicaciones maliciosas en el dispositivo, sospechar de aplicaciones con el permiso GET_TASKS, que permite consultar la lista de tareas en ejecución.
 - CWEs Relacionados: CWE-1021.
 - TTPs identificados: T1056, **T1548.004**.
- **CAPEC-697: *DHCP Spoofing***: Un atacante se enmascara como un servidor DHCP legítimo escuchando el tráfico de este protocolo con el objetivo de redirigir el tráfico de red y denegar el servicio DHCP.
 - Probabilidad: Bajo.
 - Severidad: Alta.
 - Dominio de Ataque: Software, Hardware, Ingeniería Social.
 - Mecanismo de ataque: Participar en actividades engañosas.
 - Flujo de ejecución:
 - * Explorar: Determinar el contrato DHCP existente.
 - * Experimentar: Capturar el mensaje DHCP DISCOVER.
 - * Explotar: Comprometer el acceso a la red y recoger actividad de la red.
 - Pre-requisitos: Tener acceso a una máquina en la LAN objetivo que puede mandar

ofertas DHCP al objetivo.

- Habilidades requeridas: Nivel Medio. El adversario debe identificar objetivos potenciales para la escucha DHCP y generar configuraciones de red para obtener resultados deseados.
 - Recursos requeridos: Acceso a la máquina en la LAN que no tiene tráfico DHCP asegurado.
 - Consecuencias: Confidencialidad y Control de Acceso (Lectura de datos); Integridad y Control de Acceso (Modificación de datos y ejecución de comandos no autorizados); Disponibilidad (Consumo de recursos).
 - Mitigaciones: en el diseño, reenvío forzado de MAC, y en la implementación, seguridad en los puertos y NIDS.
 - CWEs Relacionados: CWE-923.
 - TTPs identificados: **T1557.003**.
- **CAPEC-698: *Install Malicious Extension***: El atacante directamente instala o engaña a un usuario para instalar una extensión maliciosa en software confiable, provocando impactos técnicos negativos.
 - Probabilidad: Media.
 - Severidad: Alta.
 - Dominio de Ataque: Software.
 - Mecanismo de ataque: Inyectar elementos inesperados.
 - Flujo de ejecución:
 - * Explorar: Identificar el software objetivo.
 - * Experimentar: Crear extensiones maliciosas que se puedan instalar en el software identificado.
 - * Explotar: Instalar la extensión maliciosa.
 - Pre-requisitos: Generar *malware* basado en un tipo de software o sistema objetivo. Comprometer la máquina objetivo previamente.
 - Habilidades requeridas: Nivel Medio. Capacidad de crear extensiones maliciosas capaces de explotar determinado software y capacidad de explotar el sistema objetivo.
 - Consecuencias: Confidencialidad y Control de Acceso (Lectura de datos); Integridad y Control de Acceso (Modificación de datos); Autorización y Control de Acceso (Ejecución de comandos no autorizados, alterar la lógica de ejecución y obtención de privilegios).
 - Mitigaciones: Instalar solo extensiones verificadas, confirmando que son legítimas. Mantener el software actualizado. Cerrar sesiones en navegadores web al terminar

para evitar que las extensiones se ejecuten en segundo plano.

- CWEs Relacionados: CWE-507, CWE-829.
- TTPs identificados: T1769, **T1505.004**.

La información que proporcionan los CAPECs relacionados con posibles técnicas identificadas permite que la caracterización de los ciberataques sea completa, pudiendo relacionarlos con otros conceptos como CWEs y CVEs que permiten que los procesos de gestión de riesgos se adapten mejor al entorno. Además, las contramedidas y mitigaciones propuestas en base a las recomendaciones de MITRE en ambos marcos permiten que las respuestas a los ciberataques sean adecuadas al incidente registrado y no medidas genéricas.

3.5 Conclusiones

Proteger las redes y los dispositivos es uno de los principales enfoques de la investigación en ciberseguridad, puesto que los ataques continúan evolucionando y surgiendo nuevas técnicas constantemente. Los ataques de día cero son extremadamente difíciles de identificar ya que no coinciden con reglas definidas a raíz de incidentes anteriores. Para ello, es fundamental la caracterización de estos ciberataques, extrayendo información mediante técnicas de CTH que permiten identificar incidentes que no se pueden detectar con otros métodos y aportando datos para un soporte a la toma de decisiones informado.

A pesar de las ventajas que presentan los HIDS, los NIDS de anomalías se adaptan mejor a los datos disponibles para el entorno analizado en esta investigación, estableciendo un comportamiento normal y definiendo como anomalía aquello que se aleje lo suficiente. Sin embargo, los *datasets* actuales no contienen ataques modernos ni están adaptados a los entornos de fuentes heterogéneas, como el propuesto en esta Tesis Doctoral, por lo que el entrenamiento del modelo debe tener en cuenta esta limitación, a partir de modelos no supervisados o generando conjuntos de datos sintéticos.

Para la caracterización de ciberataques, la identificación de las tácticas y las técnicas implica conocer en qué momento se encuentra el incidente y con qué herramientas se está amenazando el sistema, lo que facilita su mitigación. Además, gracias a esta información, se enriquecen los procesos de gestión y evaluación de riesgos, por lo que la propuesta de esta Tesis Doctoral se apoya como base teórica en ATT&CK y CAPEC. Éstos ofrecen una información fundamental, ya que el sistema evoluciona de detectar un incidente a conocer las técnicas con las que se puede estar llevando a cabo, las debilidades que puede explotar, las consecuencias que tiene, cómo responder ante ello e, incluso, si va asociado a algún otro tipo de incidente.

A lo largo del capítulo se ha realizado una descripción de los fundamentos teóricos de los que surge la propuesta para el entorno de conciencia cibersituacional, enfocando los principales retos de la ciberseguridad y la caracterización de ciberataques, los distintos tipos de IDS y analizando el trabajo de MITRE en la recopilación de información que permita llevar a cabo este proceso.

Capítulo 4

Inteligencia Artificial aplicada a la Ciberseguridad

La IA está presente en muchos aspectos, convirtiéndose en una herramienta cada vez más utilizada. Especialmente desde que se aplica en la generación de ciberataques, estas tecnologías deben también emplearse mecanismos de defensa para los sistemas atacados. Cada vez más, los entornos de conciencia cibersituacional, como el que se presenta en esta Tesis Doctoral, se basan en IA para delegar varias de las funciones que se realizan, como la detección de intrusiones o el modelado de la base de conocimiento que permita llevar a cabo un soporte a la toma de decisiones ajustado a la situación. Este capítulo se centrará en presentar estos aspectos, que son la base de la propuesta definida. Primero, en la Sección 4.1 se presenta el contexto general de la IA aplicada en este ámbito. A continuación, en la Sección 4.2 se detalla una de las ramas de esta tecnología, las ontologías, y, de la misma forma, en la Sección 4.3 se presentará el aprendizaje automático supervisado y en la Sección 4.4 el aprendizaje automático no supervisado. Finalmente, en la Sección 4.5 se resumen las conclusiones obtenidas en el capítulo.

4.1 Introducción

La IA como campo científico abarca la creación de sistemas capaces de realizar tareas lógicas, como el reconocimiento de patrones, la resolución de problemas o la toma de decisiones. APLICADA a la ciberseguridad, se utiliza principalmente para mejorar la detección de ciberataques y la respuesta ante estas amenazas, automatizando respuestas frente a las anomalías [76].

Entre las ramas que destacan de la IA en ciberseguridad, se encuentran el aprendizaje automático y las ontologías. El primero se centra en el desarrollo de algoritmos y modelos para realizar tareas específicas a partir de unos datos de entrada, como puede ser la detección de ciberataques; mientras que las ontologías son una representación formal y estructurada del conocimiento de un dominio, a partir del que se puede generar nuevo conocimiento y, por tanto, llevar a cabo razonamientos e inferencias que deriven en una gestión adecuada del riesgo y como soporte a la toma de decisiones [147].

Las ontologías en nuestro campo son una técnica que forma parte del ámbito de la gestión del conocimiento y la IA, cuyo objetivo es la representación formal de conceptos, sus significados y sus relaciones. En el caso específico de la ciberseguridad, el valor añadido de las ontologías como base para la gestión de riesgos consiste en representar información procedente de distintas fuentes, realizar traducciones semánticas de manera automática y, sobre todo, ser capaz de generar conocimiento. En la Sección 4.2 se detallarán sus propiedades, y en el Capítulo 9 se utilizará como base para el desarrollo del sistema de gestión de riesgos dinámico e interoperable que forma parte del entorno de conciencia cibersituacional para la caracterización de ciberataques presentado en esta Tesis Doctoral.

Por otro lado, la aplicación del aprendizaje automático a día de hoy en la seguridad de la red está muy extendida, especialmente en las tareas de clasificación de incidentes o identificación de relaciones, mejorando cualquier tecnología aplicada con anterioridad. Esta rama científica [95] busca extraer información de un conjunto de datos a partir de su estructura, analizando su comportamiento y extrayendo relaciones genéricas. A partir de ahí, se crea un modelo capaz de establecer conclusiones sobre nuevos datos de entrada. En función del tipo de entrenamiento de este algoritmo, se dividen en: aprendizaje supervisado, si los datos de entrada están etiquetados; no supervisado, cuando el *dataset* no presenta información de la clase a la que pertenecen los datos; o aprendizaje por refuerzo, que utiliza recompensas y castigos para dirigir hacia las mejores acciones a tomar [69].

En esta investigación se utilizan los fundamentos técnicos detallados en la Sección 4.3 para la caracterización de tácticas y técnicas en registros de tráfico, como se muestra en el Capítulo 8, mientras que la información relacionada con los modelos no supervisados que se introduce en la Sección 4.4 se lleva a la práctica en la detección de ciberataques en entornos heterogéneos presentada en el Capítulo 7, siendo ambas parte fundamental del sistema de caracterización propuesto en la Tesis Doctoral.

4.2 Ontologías

Una de las definiciones más ampliamente aceptadas y detalladas de ontología es “una especificación explícita y formal de una conceptualización compartida” [58]. Analizando en detalle esta descripción, se considera explícita ya que establece de manera clara los conceptos, propiedades, relaciones, funciones, taxonomías, axiomas y restricciones o reglas que incluye el dominio estudiado. Además, al definirse a través de un lenguaje que las máquinas pueden interpretar, es formal. Como modelo abstracto que representa de manera simple la estructura y eventos del dominio, se define como conceptualización, que se comparte ya que representa información acordada por grupos de profesionales especializados.

Así, varios agentes inteligentes podrían compartir y reutilizar conocimiento a través de una ontología, con el objetivo de abordar desafíos de heterogeneidad surgidos en distintos niveles y lenguajes de especificación, como la variación de términos o formalismos o la falta de estandarización en los protocolos de intercambio de conocimiento y desajustes semánticos en las bases de conocimiento [58].

Las ontologías más comunes están formadas por un conjunto de clases, con sus categorías y

sub-categorías, sus propiedades y las conexiones entre ellos; y las reglas de inferencia, que permiten establecer las restricciones en los objetos de la taxonomía.

La utilización de ontologías en ciberseguridad implica un análisis completo y sistemático del entorno, para conseguir una abstracción y poder representar el conocimiento de ese dominio en su estructura.

4.2.1 Lenguajes de definición

Los lenguajes orientados a la definición de las ontologías se clasifican según la terminología que utilizan y cómo la utilizan [88].

Principalmente, los términos que aparecen son conceptos o clases, sus ejemplares y propiedades, y las relaciones definidas entre las clases:

- Clase: conceptos que se formalizan en la ontología, teniendo en cuenta el dominio de aplicación.
- Individuo: instancias de clases que modelan un concepto concreto, con valores determinados para sus propiedades que lo distinguen de los demás individuos.
- Relación: interacción entre dos clases del dominio, que favorecen la generación de nuevo conocimiento a partir de las reglas.

En la Figura 4.1 se representan gráficamente estos conceptos, ilustrados con un ejemplo del ámbito de la ciberseguridad.

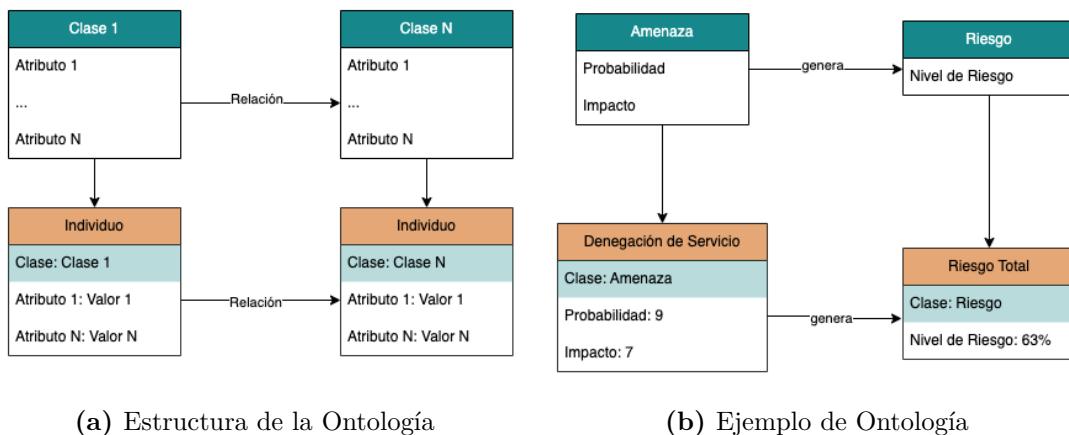


Figura 4.1: Términos utilizados en ontologías

Los lenguajes aplicados a la definición de ontologías se pueden clasificar en [101]:

- Lenguajes previos al desarrollo de la Web Semántica, que se basan en la lógica de primer orden, utilizando cuantificadores y variables para formular afirmaciones y relaciones de los individuos.
- Lenguajes de la Web Semántica, que se basan en *eXtensible Markup Language* (XML). Éste no se puede considerar un lenguaje de definición de ontologías porque presenta la

sintaxis de documentos estructurados, sin incluir restricciones semánticas. Entre ellos destacan RDF, RDFS, OWL y OWL2.

RDF y RDF-Schema

Resource Description Framework (RDF) [106] es un lenguaje de definición de ontologías definido por el *World Wide Web Consortium* (W3C) que actualmente es uno de los más utilizados en el conocimiento de la Web Semántica. Esto se debe a que fue el primer lenguaje formal en este ámbito, con una semántica que es interpretable por máquinas.

La estructura que sigue este lenguaje se basa en tripletes ‘sujeto - predicado - objeto’ que definen relaciones entre sujeto y objeto a través de la propiedad que representa el predicado. Estos tres componentes poseen un identificador único, *Uniform Resource Identifier* (URI).

Al incorporar una semántica formal, permite inferir nuevo conocimiento a partir de estos tripletes, aunque por ser el primer lenguaje de este tipo, la capacidad de razonamiento y la expresividad que proporcionan son muy limitados.

Por ese motivo surge *Resource Description Framework Schema* (RDFS) como extensión de RDF, con una sintaxis similar y compatible con el lenguaje anterior pero capaz de definir nuevas restricciones.

OWL y OWL2

Ontology Web Language (OWL) [91] es una Recomendación del W3C de 2004 para la creación e intercambio de ontologías. Surge posteriormente a RDF y RDFS, por lo que se diseñó con compatibilidad hacia ellos y con XML, pudiendo ser procesado por herramientas previas para cualquiera de estos lenguajes, ya que también define tripletes ‘sujeto - predicado - objeto’.

OWL se basa en lógicas de descripción, clasificando los datos mediante la definición de propiedades y relaciones entre estas clases e individuos. Este lenguaje está orientado a objetos a partir de una sintaxis abstracta para declarar clases, propiedades e instancias.

Las clases por lo general se nombran con mayúscula inicial (Riesgo) o mixtas en nombres complejos (AmenazaDeliberada). Por otro lado, las propiedades se clasifican en dos tipos: propiedades de objeto, que representan las relaciones entre clases; y propiedades de datos, haciendo referencia a los atributos de las clases. En cualquier caso, se nombran con minúscula inicial (impacto, genera), o mayúsculas mixtas si el concepto se define con varias palabras (nivelDeRiesgo, protegeFrente). Finalmente, los individuos se nombran como las clases, con mayúscula inicial (Incidente_1) o mixtas (AdministradorDeSistema_1).

Existen tres versiones de OWL, ya que su estructura en capas permite adaptar la complejidad y el nivel de expresividad a los requisitos del dominio que se modela:

- OWL Lite: es la versión más reducida y sencilla, un subconjunto de las construcciones definidas, con restricciones.
- OWL *Description Logic* (OWL DL): en esta versión se dispone del nivel máximo de expresividad, alcanzando conclusiones según la información existente. Incluye todas las construcciones definidas en OWL, con restricciones.

- OWL *Full*: es la versión más amplia, con total expresividad y libertad sintáctica. Es compatible con RDF/RDFS, por lo que puede aplicar construcciones tanto de OWL como de RDF(S), pero no garantiza la capacidad de alcanzar un razonamiento debido a esta libertad.

En 2012 el W3C revisó este lenguaje de definición de ontologías, extendiéndolo para generar OWL2. Esta nueva versión extiende la sintaxis y la semántica de OWL, aportando mayor expresividad e incorporando otros formatos como la sintaxis Manchester o Turtle.

En el caso de OWL2, existen tres sub-lenguajes, versiones reducidas para mejorar la eficacia del razonamiento:

- OWL2 *Existencial Language* (OWL2 EL): contiene una gran cantidad de propiedades y clases. Permite llevar a cabo razonamientos básicos.
- OWL2 *Query Language* (OWL2 QL): enfocado al tratamiento de grandes cantidades de instancias o individuos, ejecutando de forma eficiente el razonamiento sobre los datos, pero con limitaciones en la expresividad.
- OWL2 *Rule Language* (OWL RL): ofrece una capacidad de razonamiento escalable, utilizando reglas y mayor expresividad.

Las diferencias principales entre OWL y OWL2 radican en la capacidad que proporciona esta última para aumentar la expresividad, con nueva sintaxis y más razonadores soportados.

La aplicación principal de OWL es la representación de información que debe ser procesada por sistemas, no únicamente ser presentada, y tiene asociados lenguajes de inferencia de conocimiento.

4.2.2 Reglas de inferencia de conocimiento

Además de la capacidad para definir la representación del dominio, es necesario que las ontologías sean capaces de reaccionar y adaptar su información a los eventos mediante lenguajes de especificación del comportamiento. Éstos deben expresar restricciones mayores que las incluidas en los lenguajes de definición de las ontologías.

A continuación se van a presentar los lenguajes de inferencia de conocimiento compatibles con OWL que mejor se adaptan a los entornos de gestión de riesgos y caracterización de ciberataques, como el propuesto en esta Tesis Doctoral.

Semantic Web Rule Language (SWRL)

El lenguaje de reglas de la Web Semántica (*Semantic Web Rule Language*, SWRL) [133] está basado en otros lenguajes como OWL y se creó para incrementar la expresividad de sus restricciones y reglas. Surge al integrar OWL DL y OWL Lite con sub-lenguajes de definición de reglas no compatibles con OWL, como RuleML.

Una de las principales adiciones sobre OWL son las relaciones entre propiedades compuestas, como por ejemplo entre *padre*, *hijo* y *tío*.

La semántica de estas reglas está formada por un antecedente y un consecuente, ambos compuestos por átomos, de forma que cuando los datos existentes en la ontología cumplen el antecedente, se infiere el conocimiento nuevo definido por el consecuente, para que esas condiciones también se puedan verificar:

$$\text{antecedente} \rightarrow \text{consecuente} \Rightarrow \text{átomo}_{a1} \wedge \dots \wedge \text{átomo}_{an} \rightarrow \text{átomo}_{c1} \wedge \dots \wedge \text{átomo}_{cn}$$

Los átomos pueden representar los distintos componentes de una ontología:

- Átomo de clase, como por ejemplo *Activo* ($?a$), donde *Activo* es la clase y $?a$ la variable que representa a los individuos de esa clase.
- Átomo de propiedad individual, como *afecta*($?incidente, ?a$). En este caso, el átomo representa una propiedad de objeto entre dos individuos, *afecta*, y los dos individuos relacionados se asocian a las variables $?incidente$ y $?a$.
- Átomo de propiedad con valor de datos, como *impacto*($?amenaza, 3$). Este tipo de átomo representa atributos de los individuos. De nuevo, acepta dos argumentos, la variable del individuo y el valor que toma el atributo en ese individuo concreto.

Un ejemplo de regla sería la siguiente, donde se estima que dado un activo al que afecta un incidente, este activo será amenazado con un valor de impacto de 3:

$$\text{Activo} (?a) \wedge \text{Incidente} (?i) \wedge \text{afecta} (?i, ?a) \rightarrow \text{Amenaza} (?t) \wedge \text{afecta} (?t, ?a) \wedge \text{impacto} (?t, 3)$$

Las ventajas de SWRL radican en la capacidad de definir condiciones complejas, utilizando el operador AND, con un alto nivel de abstracción y expresividad. SWRL es integrable fácilmente con las ontologías OWL. Como inconvenientes de este lenguaje de definición de reglas se encuentran la imposibilidad de utilizar operadores como NOT y OR, que limita las definiciones de los átomos.

SPARQL *Inferencing Notation* (SPIN)

SPARQL *Inferencing Notation*, SPIN, [130] combina conceptos de lenguajes orientados a objetos, lenguajes de consulta y sistemas basados en reglas para definir el comportamiento de los datos.

SPIN proporciona, mediante el estándar de W3C SPARQL, un marco de trabajo para aprovechar el rendimiento y expresividad de SPARQL con el objetivo de superar en estos campos a otras recomendaciones del W3C. A través de este lenguaje, las reglas pueden incluir cálculos de valores para una propiedad en base a otras propiedades o ejecutar un conjunto de reglas bajo determinadas condiciones. Una de las bases de SPIN es unir definiciones de clases con consultas SPARQL para establecer reglas y formalizar el comportamiento esperado de estas clases.

La implementación de estas reglas utiliza construcciones típicas de SPARQL como UPDATE, INSERT, DELETE o CONSTRUCT, además de permitir la validación de los requisitos de formato en los datos y comprobar las limitaciones mediante la construcción ASK.

El beneficio principal [129] de la sintaxis de SPIN es poder almacenar las consultas y reglas junto al modelo. Las reglas se construyen a base de tripletes ‘sujeto - predicado - objeto’ en

formato consulta:

```
CONSTRUCT {
# Tripletes de individuos o relaciones
# construidas a partir de la regla
} WHERE {
# Tripletes de condición.
# Cuando se cumplen, las definiciones
# de la parte superior se construyen }
```

Los tripletes de la regla o la condición, como en el caso de SWRL, se forman siguiendo el patrón `?variable_sujeto relación ?variable_objeto` o `?variable_sujeto atributo ?valor_objeto`.

En el caso de las consultas SELECT, que permiten verificar la información almacenada en el modelo de dominio, la estructura es la siguiente:

```
SELECT # Lista de variables
WHERE {
# Tripletes de condición.
}
```

Estas consultas y reglas pueden complementarse con construcciones como FILTER, OPTIONAL, UNION o BIND, que permiten operar sobre los valores de las propiedades y los atributos o crear nuevos valores a partir de los antiguos. Como lenguaje de consultas, también permite el uso de agregaciones como AVG, COUNT, MIN, MAX, SUM, DISTINCT, NOT y modificadores de los resultados como LIMIT, OFFSET o ORDER BY.

Las ventajas principales de SPIN son la expresividad que proporciona, definiendo las reglas de forma clara, incluso en el caso de consultas complejas. Además, se basa en un estándar integrado con lenguajes de definición como OWL y RDF, que facilita su implementación y permite la inferencia no solo de nuevas relaciones, sino de nuevos datos a partir de los existentes. Sin embargo, el inconveniente principal que presenta es que, para volúmenes de datos elevados, el rendimiento de las consultas puede verse afectado, requiriendo una optimización y mantenimiento de las reglas y restricciones definidas para una ontología.

4.2.3 Razonadores semánticos

Los lenguajes analizados hasta el momento permiten representar el modelo de dominio de una ontología y definir restricciones e inferencias sobre esta información, formando la base del conocimiento. Esta información se complementa con los razonamientos semánticos, que permiten generar conocimiento implícito en los datos definidos explícitamente. Este motor de razonamiento o motor de reglas es capaz de extraer consecuencias lógicas a partir de la base de conocimiento y se complementa con las reglas de inferencia definidas en lenguajes como SWRL o SPARQL.

Existen distintos razonadores que conviven dentro de la Web Semántica, diferenciándose según el lenguaje de reglas soportado y el mecanismo que utilizan para llevar a cabo este

razonamiento. En este aspecto, se dividen principalmente en estos tipos de razonamiento [42]:

- Razonamiento deductivo: es un razonamiento que depende de los hechos. Cuando aparece un dato en la base de conocimiento, a partir de las reglas definidas se extraen nueva información hasta alcanzar el objetivo de la regla. Parte de afirmaciones generales buscando conclusiones específicas a partir de reglas lógicas y axiomas.
- Razonamiento inductivo: es un razonamiento que depende de los objetivos. A partir de una hipótesis inicial se construye la cadena de razonamiento en sentido inverso. Tratan de generalizar desde ejemplos específicos y obtener unos patrones generales, que pueden no localizarse. Permiten encontrar patrones en los datos, acercándose al funcionamiento del aprendizaje automático.
- Razonamiento mixto: Estos razonadores tratan de aprovechar las ventajas de los anteriores para mejorar la eficacia de las inferencias, buscando un equilibrio entre precisión y capacidad de descubrimiento.

A continuación se describen los razonadores deductivos más utilizados que son compatibles con OWL.

Pellet

Este razonador [98] basado en OWL 2 utiliza algoritmos de lógica descriptiva y razonamiento deductivo. De manera optimizada se utiliza para validar ontologías, comprobar la clasificación y consistencia de clases y responder a consultas SPARQL o ejecutar reglas SWRL. Al estar definido con el lenguaje de programación Java, se puede utilizar con librerías como Jena o OWL API, o con la librería de Python, OwlReady2. Este razonador permite verificar que la ontología no contiene en la base de conocimiento información contradictoria y otros procesos de depuración de ontologías, solución de errores y detección de inconsistencias, especialmente en ontologías de tamaño mediano, proponiendo un razonamiento incremental.

HermiT

HermiT [55] es un razonador semántico basado en OWL y OWL2 DL desarrollado por la universidad de Oxford. Además de verificar la consistencia de la ontología, utiliza técnicas de razonamiento deductivo basada en lógica de la descripción para localizar relaciones entre clases. Es compatible con toda la lógica de OWL2 realizando clasificación de propiedades de objeto y de datos, al contrario que el resto de razonadores, y además soporta consultas SPARQL y reglas SWRL. HermiT también está construido sobre Java, es el razonador por defecto compatible con la librería de Python más utilizada para el manejo de ontologías, OwlReady2. A partir de una serie de optimizaciones, como el cálculo de “hipertabla”, trata de ser eficiente en tiempo de ejecución y uso de memoria en su aplicación sobre ontologías buscando modelos para satisfacer una fórmula lógica a partir de árboles de búsqueda. En la clasificación de ontologías complejas, mejora los resultados de los anteriores razonadores.

FaCT++

Este razonador [50] usa algoritmos basados en la lógica de descripción y teoría de modelos para realizar inferencias y verificar la consistencia del modelo. Utiliza cálculo de tablas sobre OWL DL y OWL2 DL, y es compatible con la API de OWL.

4.3 Aprendizaje automático supervisado

Como se ha mencionado anteriormente, estos algoritmos supervisados permiten clasificar o realizar tareas de regresión si son entrenados sobre conjuntos de datos que ya contienen una etiqueta.

En el entorno de la ciberseguridad, la tarea principal de los modelos de aprendizaje automático es la clasificación, asignando etiquetas dentro de un rango definido a los datos de entrada. Para ello, según los datos de entrada, que deben procesarse adecuadamente para que el modelo sea capaz de utilizarlos, debe elegirse un algoritmo supervisado adecuado y adaptarlo a la tarea de clasificación objetivo según los datos pre-procesados de entrada. Finalmente, el rendimiento de estos modelos se evalúa según una serie de métricas. Al tratar con datos etiquetados, se conoce una verdad objetiva que permite analizar qué tal se adapta el algoritmo optimizado a los datos para llevar a cabo la asignación de clases.

4.3.1 Algoritmos

Los algoritmos de aprendizaje automático [96] son un conjunto de reglas matemáticas que permiten a un sistema realizar una tarea, ajustándose automáticamente para mejorar los resultados al exponerse a información.

En función de la aproximación que realiza al problema, existen algoritmos -de regresión lineal, logística o árboles de decisión-, o redes neuronales. A continuación se van a definir los más utilizados en tareas de caracterización por sus propiedades.

Árboles de decisión

Los modelos en forma de árbol [52] representan en cada nodo una prueba sobre uno de los atributos, donde cada rama que sale de él es un posible resultado de esa comprobación. En la raíz se encuentra el atributo con mayor ganancia de información, el siguiente nivel se determina por el siguiente más alto y así sucesivamente. De forma recursiva, el algoritmo [18] divide los datos en subconjuntos según las características más importantes en cada nodo del árbol. El esquema de este algoritmo se muestra en la Figura 4.2.

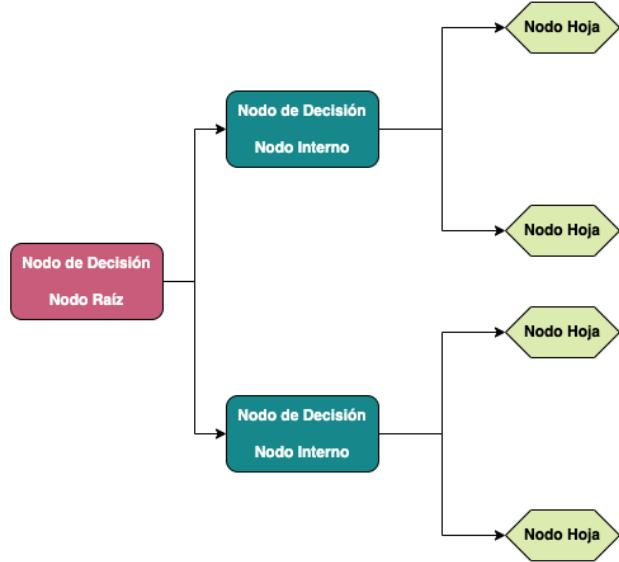


Figura 4.2: Esquema de funcionamiento del algoritmo de árbol de decisión

Para configurar correctamente un árbol de decisión es necesario establecer sus hiper-parámetros [116]:

- Profundidad máxima del árbol (*max_depth*): indicador del número de decisiones que tomará el modelo antes de alcanzar una predicción.
- Estado aleatorio (*random_state*): controla el nivel de aleatoriedad del estimador, ya que los atributos se intercambian aleatoriamente en cada división.
- *Criterion*: función que evalúa la calidad de una división.
- *Splitter*: estrategia seguida para elegir la división en cada nodo.
- *min_samples_split*: número mínimo de muestras necesarias para realizar una división en un nodo.
- *min_samples_leaf*: número mínimo de muestras requeridas para encontrar un nodo hoja.
- *max_features*: número de atributos o características consideradas para buscar la mejor división.
- *class_weight*: pesos asociados a cada clase.

Random Forest

Este algoritmo [96] está compuesto de muchos árboles de decisión, donde cada uno se crea a partir de un subconjunto de las características. Para obtener la predicción final, cada árbol vota según su información, y el algoritmo computa todos los resultados hasta obtener la clasificación más adecuada (la más votada) [18], obteniendo así buenos resultados frente al sobre-ajuste de los datos. Esto se debe a que varios modelos no correlacionados trabajan

mejor en grupo. En la Figura 4.3 se muestra el esquema de este algoritmo.

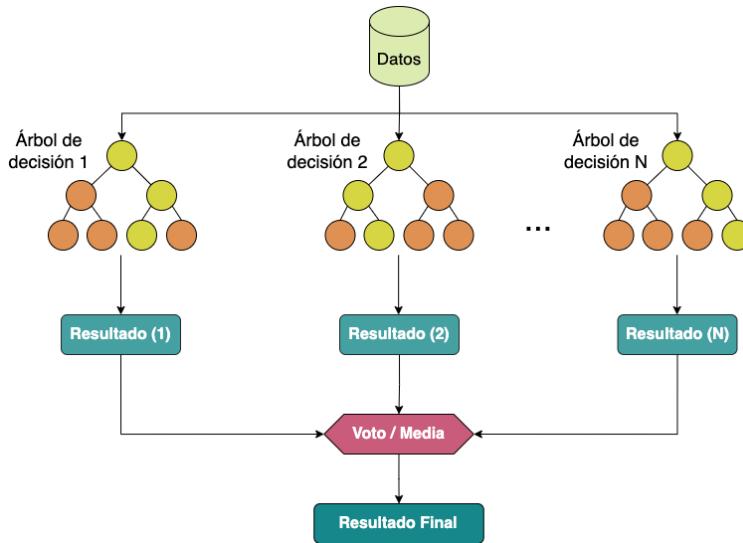


Figura 4.3: Esquema de funcionamiento del algoritmo *Random Forest*

En este caso, al partir de un conjunto de árboles de decisión, los hiper-parámetros que definen el algoritmo son los siguientes [120]:

- *Criterion*: función que mide la calidad de una división.
- *max_depth*: profundidad máxima del árbol.
- *n_estimators*: número de estimadores (árboles de decisión) en el algoritmo *Random Forest*.
- *min_samples_split*: número mínimo de muestras requeridas para dividir un nodo interno.
- *min_samples_leaf*: número mínimo de muestras necesarias para definir un nodo hoja.
- *max_features*: número de atributos máximo considerado para buscar la mejor división.
- *n_jobs*: número de trabajos ejecutados en paralelo.
- *random_state*: Controla tanto la aleatoriedad del muestreo de los datos de entrada utilizados para construir los árboles como el muestreo de las características a considerar al buscar la mejor división en cada nodo.
- *class_weight*: pesos asociados a cada clase del conjunto de datos.

Gradient boosted decision trees (XGBoost)

Este método [144] comienza con clasificadores débiles y mejora sus resultados a partir de un procesado secuencial con una función de pérdidas para minimizar el error en cada iteración, obteniendo finalmente un modelo robusto.

Utiliza un número elevado de árboles [18] a partir de segmentos del *dataset*, partiendo del

atributo con más información. Al combinarse, consiguen delimitar el camino más preciso a partir de los datos (Figura 4.4).

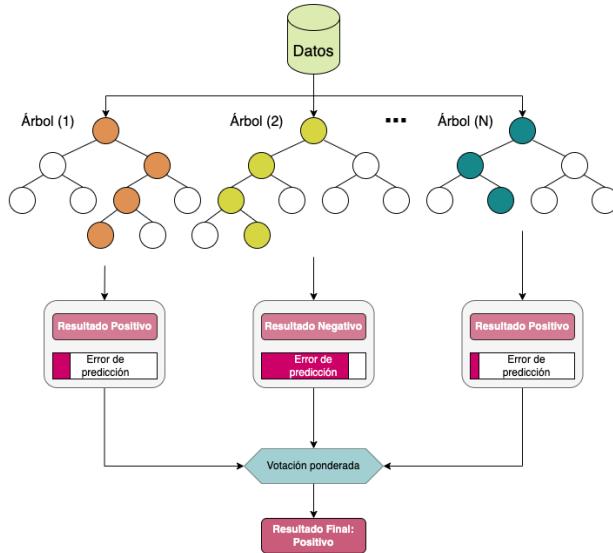


Figura 4.4: Esquema de funcionamiento del algoritmo XGBoost

Los modelos de XGBoost se configuran mediante los siguientes hiper-parámetros [117]:

- *n_estimators*: número de etapas de mejora.
- *Criterion*: función para medir la calidad de la división.
- *loss*: función de pérdida que se optimiza.
- *learning_rate*: valor en que se reduce la contribución de cada árbol.
- *min_samples_split*: número mínimo de muestras requeridas para dividir un nodo interno.
- *min_samples_leaf*: número mínimo de muestras necesarias para definir un nodo hoja.
- *random_state*: controla la semilla aleatoria que se proporciona a cada árbol en cada iteración, además de la permutación aleatoria de las características en cada división.
- *max_features*: número de atributos máximo considerado para buscar la mejor división.

4.3.2 Pre-procesado de datos

El objetivo principal del pre-procesado de los datos es conseguir que la información que contienen pueda ser accesible [102]. Cuando se aplican en tareas de clasificación con aprendizaje automático, si los datos son impuros (están incompletos, son inconsistentes o existe ruido) pueden conllevar predicciones inexactas.

Al pre-procesar los datos, se obtiene un *dataset* con datos de calidad. Este proceso incluye todas las técnicas de análisis de datos que permiten que los algoritmos puedan obtener mejor

información a partir de ellos: recogida e integración de los datos, limpieza, transformación o reducción.

En el entorno de la Ciberseguridad, debido al riesgo que conlleva trabajar directamente con ataques, la recogida e integración de los datos suelen ser procesos que se realizan previamente al entrenamiento, con el objetivo de generar un *dataset* que pueda ser validado en distintos entornos y aplicable a varios algoritmos.

Como parte del pre-procesado, se deben eliminar aquellos datos que impiden que el modelo pueda aprender de ellos, como los campos nulos o los *outliers*. Cada contexto y conjunto de datos requiere una limpieza distinta, eliminando estos registros o sustituyéndolos por valores como la media de los valores de la característica.

Los algoritmos de aprendizaje automático, como modelos matemáticos, aceptan entradas únicamente numéricas, pero los conjuntos de datos en ocasiones presentan características con información textual o lógico (Verdadero/Falso), que deben transformarse en enteros. El manejo de estos casos puede ser sencillo y realizarse de forma manual si el número de etiquetas es reducido, pero requiere de funciones más complejas como la codificación de etiquetas para atributos de los datos que toman muchos valores distintos, para que esta transformación se realice de forma automática [119].

Finalmente, una de las tareas principales para optimizar el rendimiento de los modelos es la reducción de datos en base a la información que aportan al modelo las distintas características y la obtención de muestras nuevas. Las herramientas que permiten esta optimización son el balanceo de datos, la información mutua y la correlación entre los datos, que se detallan a continuación.

Balanceo de datos

El desequilibrio en los datos se produce cuando en el conjunto de datos las clases no están igualmente representadas.

Esta situación puede considerarse algo común, ya que en los entornos reales no existe una igualdad absoluta entre los datos; sin embargo, esta situación se convierte en un problema cuando las diferencias entre el tamaño de las clases es muy grande, como ocurre con los ciberataques. Si estos datos desbalanceados se utilizan en entrenamientos de modelos y redes neuronales, estos algoritmos tienden a ignorar las clases con menor número de muestras, sobre-ajustándose a las más frecuentes.

La solución óptima para esta situación sería la recolección de datos nuevos para las clases menos representadas. Sin embargo, esto no siempre es posible, por lo que en estas ocasiones el *dataset* se re-muestrea, añadiendo datos a las clases menos frecuentes o eliminando de las más pobladas.

Para llevar a cabo estos procesos, existen distintos enfoques:

- Para la reducción de muestras (*undersampling*), la técnica más utilizada es la eliminación aleatoria de muestras en una o varias clases.
- Para la obtención de muestras nuevas (*oversampling*), similar al caso anterior, un

procedimiento ampliamente utilizado es la réplica aleatoria de muestras de las clases minoritarias.

Sin embargo, la solución más eficiente, y por tanto la más aplicada, es la generación sintética de muestras mediante la técnica *Synthetic Minority Over-sampling Technique* (SMOTE) [35]. Este algoritmo selecciona muestras similares en términos de distancia, las conecta y crea nuevas instancias a partir de los puntos que unen las muestras originales.

Según la estructura de los datos, las técnicas se pueden combinar, por ejemplo: si no existe equilibrio entre las clases, pero hay clases con un número aceptable de instancias para entrenar un modelo, las clases con más volumen de datos se sub-muestrean, eliminando datos, y las menos representadas se sobre-muestrean, generando nueva información a partir de la existente. Por otra parte, es imprescindible tener en cuenta las limitaciones y consecuencias que implican estos procesos: la generación aleatoria supone crear registros duplicados que, en exceso, pueden llevar al sobre-ajuste del modelo, mientras que el sub-muestreo implica pérdida de información, perjudicando el proceso de aprendizaje del algoritmo.

En los casos para los que el balanceo de datos es recomendable, los beneficios que proporciona deben compararse con las consecuencias de su aplicación y, teniendo como referencia el rendimiento del modelo en todos los casos, buscar el equilibrio óptimo entre las clases.

Información mutua

La información mutua en el pre-procesado de datos permite identificar qué características tienen mayor relación con la variable objetivo, y por tanto son más relevantes a la hora de hacer predicciones.

La entropía y la información mutua [139] son conceptos básicos de la teoría de la información. La entropía (H) mide la incertidumbre de una variable aleatoria en relación con la probabilidad de ocurrencia de un evento. Una entropía elevada implica que cada evento tiene la misma probabilidad de ocurrencia, mientras que la entropía baja implica que cada evento tiene una probabilidad distinta. La entropía de una variable discreta x , cuya probabilidad de masa es $p(x(i)) = \Pr\{x = x(i)\}, x(i) \in x$ se representa en la Ecuación 4.1:

$$H(x) = - \sum_{i=1}^n p(x(i)) \cdot \log_2(p(x(i))) \quad (4.1)$$

Siendo x e y dos variables aleatorias discretas, la entropía conjunta de x e y con una probabilidad de masa conjunta $p(x(i), y(j))$, es la suma de la incertidumbre de las dos variables. Esta entropía conjunta se calcula siguiendo la Ecuación 4.2:

$$H(\{x, y\}) = - \sum_{i=1}^n \sum_{j=1}^n p(x(i), y(j)) \cdot \log_2(p(x(i), y(j))) \quad (4.2)$$

Esta ecuación cumple que $\max(H(x), H(y)) \leq H(\{x, y\}) \leq H(x) + H(y)$.

Por otro lado, la entropía condicional mide la incertidumbre restante de x cuando el valor de y

es conocido. El valor mínimo que puede tomar es 0, cuando las variables son estadísticamente dependientes, y el máximo se da cuando son independientes. Esta entropía condicional se define en la Ecuación 4.3:

$$H(x|y) = \sum_{j=1}^n p(y(j)) \cdot H(x|y = y(j)) = H(\{x, y\}) - H(y) \quad (4.3)$$

Aquí, $0 < H(x|y) < H(x)$ y $H(x|y) = y(j)$ es la entropía de todas las $x(i)$ asociadas con $y = y(j)$.

A partir de estas descripciones, se define la información mutua como la cantidad de información que una variable posee sobre otra. En el ámbito de la selección de características, este cálculo permite medir la relevancia de un conjunto de atributos de los datos según las clases de salida. De manera formal, la información mutua (IM o I) se presenta en la Ecuación 4.4:

$$I(x; y) = \sum_{i=1}^n \sum_{j=1}^n p(x(i), y(j)) \cdot \log\left(\frac{p(x(i), y(j))}{p(x(i)) \cdot p(y(j))}\right) \quad (4.4)$$

La IM alcanza el valor de 0 cuando las dos variables son estadísticamente independientes. La relación lineal con las entropías presentadas se muestra en la Figura 4.5:

$$I(x; y) = H(x) - H(x|y) = H(y) - H(y|x) = H(x) + H(y) - H(x, y)$$

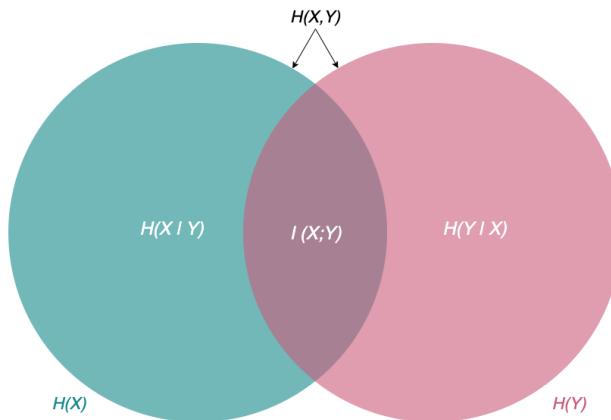


Figura 4.5: Diagrama de Venn sobre la relación entre información mutua y entropía

Por último, si se considera una tercera variable aleatoria discreta z , su interacción con las otras dos variables x e y se mide a partir de la información mutua condicional (Ecuación 4.5).

$$I(x; y|z) = \sum_{i=1}^n p(z(i)) \cdot I(x; y|z = z(i)) \quad (4.5)$$

Esta formulación matemática permite medir la IM de x e y en el contexto de $z = z(i)$. Llevado a la selección de características, permite establecer la información de dos variables en el contexto de la tercera, no entre las tres.

Análisis de la correlación

La correlación [62] entre atributos de los datos de entrada mide la redundancia entre la información que aportan a la decisión de un modelo. Si están completamente correlacionados, son datos redundantes porque proporcionan la misma información.

La correlación entre las características de entrada y la categoría objetivo se puede considerar fuerte, débil o no correlacionado. Si los datos de entrada están correctamente seleccionados, las características deberían estar fuertemente correlacionadas con esta categoría y débilmente o no correlacionadas entre ellas. Es decir, los datos no deben ser redundantes.

La aproximación para medir la correlación de estos datos se puede realizar a partir de un enfoque lineal o basado en la teoría de la información. La información mutua presentada anteriormente utiliza el segundo enfoque, y en esta sección se plantea una selección de características a partir de un análisis lineal de la correlación.

El coeficiente de correlación r mide este valor entre distintas categorías de la siguiente forma, siendo x e y dos características, \bar{x} y \bar{y} los valores medios de estos atributos, y s_x y s_y las desviaciones estándar de sus valores (Ecuación 4.6).

$$\begin{aligned} \bar{x} &= \frac{1}{n} \sum_{i=1}^n x_i & , \quad \bar{y} &= \frac{1}{n} \sum_{i=1}^n y_i & ; \quad s_x &= \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1}} & , \quad s_y &= \sqrt{\frac{\sum_{i=1}^n (y_i - \bar{y})^2}{n-1}} \\ r_{xy} &= \frac{\sum_{i=1}^n (x_i - \bar{x}) \cdot (y_i - \bar{y})}{(n-1) \cdot s_x \cdot s_y} \end{aligned} \quad (4.6)$$

Siendo n el número de muestras, el valor del coeficiente de correlación varía entre -1 y 1 . Cuanto mayor es su valor absoluto, más correlacionadas están las características x e y , siendo $r_{xy} = 0$ cuando son atributos independientes.

Dado un conjunto S de N características F_i , y siendo C la categoría del problema de clasificación que se aborda, se plantean dos tipos de correlación entre los datos de entrada:

- Para cualquier característica $F_i \in S$, la correlación entre F_i y C se denomina Correlación-C de la característica F_i . Esta medida influye sobre el rendimiento de la clasificación, cuanto mayor es, mejor clasificación se llevará a cabo.
- Para cualquier característica $F_i \in S$, la correlación entre F_i y F_j , ($j \neq i$) se denomina Correlación-F de la característica F_i . Un conjunto de datos ideal debería contener características correlacionadas con la etiqueta y no correlacionadas entre sí, por lo que este valor permite identificar información redundante tras estudiar la Correlación-C. A partir de este cálculo se obtiene la matriz de correlación R (Ecuación 4.7).

$$R = \begin{pmatrix} 1 & r_{12} & r_{13} & \dots & r_{1n} \\ & 1 & r_{23} & \dots & r_{2n} \\ & & \ddots & r_{ij} & \vdots \\ & & & 1 & r_{(n-1)n} \\ & & & & 1 \end{pmatrix} \quad (4.7)$$

En la matriz, los valores r_{ij} representan el valor absoluto de los coeficientes de correlación-F entre las características i y j . Al definir un umbral δ mayor que 0, cuando un coeficiente r_{ij} supera este valor, se elimina la característica anterior, para evitar redundancia en los datos.

4.3.3 Métricas

La ventaja principal de los modelos de aprendizaje supervisado es la presencia de una etiqueta que permite clasificar cada una de las muestras. Esta característica permite que la evaluación de los modelos sea directa, comparando las etiquetas obtenidas por los algoritmos con las clases reales. Así, para evaluar el rendimiento de los distintos modelos supervisados utilizados a lo largo de esta Tesis Doctoral, se utilizarán las siguientes métricas.

Matriz de confusión

Las matrices de confusión [25] resumen en una tabla el éxito en la predicción del modelo. Cada uno de los ejes representa la clase que ha predicho el modelo frente a los valores reales. Esta métrica permite identificar patrones en los fallos cometidos al clasificar etiquetas similares y como base para las siguientes métricas.

Como se puede ver en la Figura 4.6, en la diagonal de la matriz, de color azul oscuro, se encuentran los valores clasificados correctamente, y en el resto de la matriz se pueden cuantificar los errores en la asignación de etiquetas, en color azul claro. Como ejemplo, para la clase $N - 1$, los valores etiquetados correctamente serán los Verdaderos Positivos (VP), en rosa. El resto de valores de la columna ‘Clase $N - 1$ ’, en color verde, son etiquetados como datos de esa clase sin pertenecer a ella, y por tanto se consideran falsos positivos (FP), mientras que los valores de la fila ‘Clase $N - 1$ ’, en naranja, son falsos negativos (FN), deberían etiquetarse en esa clase y, sin embargo, se han clasificado en las demás. Así, para la clase $N - 1$, todos los campos azules (oscuro y claro) son verdaderos negativos (VN).

	Clase 1	Clase 2	...	Clase N-1	Clase N
Clase 1	VN	VN	VN	FP	VN
Clase 2	VN	VN	VN	FP	VN
...	VN	VN	VN	FP	VN
Clase N-1	FN	FN	FN	VP	FN
Clase N	VN	VN	VN	FP	VN

Figura 4.6: Composición de la matriz de confusión

Índices positivos y negativos

Del análisis de la matriz de confusión, como se ha mencionado anteriormente, se obtienen algunas métricas [25]. Los VP representan las predicciones correctas del modelo en la clase N correspondiente, mientras que los VN representan aquellas muestras que el modelo no ha clasificado dentro de una clase N a la que no pertenecen. En ambos casos, el modelo acierta en la asignación de las etiquetas.

En el otro extremo, un FP representa aquellas muestras que se asignan a una clase N a la que no pertenecen, mientras que los FN se refieren a las muestras de la clase N a las que se asigna otra etiqueta. Ambos son errores en la clasificación igualmente.

Exactitud (Accuracy)

Esta métrica [25] es una de las más frecuentes para la evaluación de modelos. Se evalúa el número de predicciones correctas sobre el número total de muestras, obteniendo la Ecuación 4.8.

$$\text{Exactitud} = \frac{VP + VN}{VP + VN + FP + FN} \quad (4.8)$$

Sensibilidad (recall) y Precisión (precision)

La Tasa de Verdaderos Positivos (TVP), *recall* o sensibilidad [25], mide la proporción de muestras de la clase N identificadas correctamente sobre todas las muestras que pertenecen realmente a esa clase (Ecuación 4.9).

$$Sensibilidad = \frac{VP}{VP + FN} \quad (4.9)$$

La precisión [25] representa el porcentaje de identificaciones positivas de muestras de la clase N sobre el total de clasificaciones positivas, es decir las muestras que pertenecen a esa clase correctamente identificadas y las que se han asociado a esa clase de forma equivocada (Ecuación 4.10).

$$\text{Precisión} = \frac{VP}{VP + FP} \quad (4.10)$$

Otra de las métricas más utilizadas es *F1-Score*, que representa la media entre la precisión y la sensibilidad (Ecuación 4.11).

$$\text{F1-Score} = \frac{2 \times \text{Precisión} \times \text{Sensibilidad}}{\text{Precisión} + \text{Sensibilidad}} \quad (4.11)$$

Otras métricas

A partir de la matriz de confusión, se pueden obtener otras métricas como la Especificidad o Tasa de Verdaderos Negativos (TVN), que representa la proporción de muestras negativas correctamente identificadas, es decir aquellas que no pertenecen a la clase N y que no se asociaron a esta etiqueta entre las muestras etiquetadas erróneamente como clase N y las no asignadas a esta clase de forma correcta (Ecuación 4.12).

$$TVN = \frac{VN}{VN + FP} \quad (4.12)$$

También se puede definir la Tasa de Falsos Positivos (TFP) o Error de Tipo 1, que representa los fallos en los que el modelo predice una muestra como positiva cuando es negativa, es decir, asocia la muestra a la clase N cuando no pertenece a ella (Ecuación 4.13).

$$TFP = \frac{FP}{FP + VN} \quad (4.13)$$

Por último, la Tasa de Falsos Negativos (TFN) o Error de Tipo 2, representa los casos en los que el modelo predice erróneamente una clase negativa, es decir, las muestras de la clase N que no se han etiquetado en esa clase (Ecuación 4.14).

$$TFN = \frac{FN}{FN + VP} \quad (4.14)$$

Área bajo la curva - Curva ROC

La métrica del área bajo la curva o *Area Under Curve* (AUC) [25] representa la probabilidad de que un clasificador tenga más confianza en que una muestra positiva sea realmente positiva

que de que una muestra negativa sea negativa. Cuanto mayor es este valor, el modelo identifica mejor las distintas clases, siendo 1 el valor ideal.

A partir de ahí, la métrica *Receiver Operating Characteristic* (ROC) [25] evalúa la calidad del modelo de aprendizaje automático, la capacidad de distinguir entre las clases.

Así, la Curva AUC-ROC representa en el eje X los valores de TFP y en el eje Y los valores de TVP. Los resultados esperados deben localizar la curva en la esquina superior izquierda, lo que maximiza el valor AUC con mayor TVP y menor TFP.

En los modelos multi-clase, las etiquetas deben transformarse en datos binarios, obteniendo una curva para cada etiqueta.

La evaluación mediante esta métrica puede aproximarse desde dos perspectivas: ‘*micro-averaging*’ implica que los valores de VP, VN, FP y FN se consideran separadamente pero sin distinguir entre las clases, mientras que ‘*macro-averaging*’ realiza una agrupación entre todas las clases.

4.4 Aprendizaje automático no supervisado

Dada la amplia variedad de algoritmos y procesos aplicables en el aprendizaje automático no supervisado, en esta sección se van a presentar únicamente las distintas técnicas y métricas empleadas en el desarrollo de la Tesis Doctoral, y sus principales alternativas.

4.4.1 Algoritmos de *clustering*

Es una de las mayores familias de aprendizaje no supervisado, cuyo objetivo principal es agrupar datos no etiquetados para elaborar conjuntos o grupos (*clusters*) [83]. Cada uno de estos conjuntos de datos son una colección de información que se parecen entre ellos, según determinadas métricas. Esta definición implica que registros de distintos *clusters* tendrán propiedades que los diferencian, por lo que esta agregación/segmentación dependerá del método por el que se analizan los datos. Cualquier variación en las métricas implicará distinto número de grupos y de diferente tamaño.

K-Means

Este método de *clustering* [108] es uno de los más simples. La base de K-Means depende de encontrar correctamente los centroides, referencias con las que se van a comparar el resto de muestras, agregándose al grupo del centroide con el que tengan más similitud. Cada *cluster* tendrá un único centroide, que se considerará el centro de masas.

Principalmente se utilizan dos métodos para calcular la distancia entre cada punto y los centroides [24] (Ecuaciones 4.15 y 4.16):

- La similitud de coseno entre dos puntos se calcula a partir del ángulo que forman los vectores resultantes. Siendo X una matriz de dimensiones $m \times n$ que se puede descomponer en m vectores fila de dimensiones $1 \times n$ (x_1, \dots, x_m), la similitud de coseno

ente los vectores x_s y x_t será:

$$d = 1 - \frac{x_s x'_t}{\sqrt{(x_s x'_s)(x_t x'_t)}} \quad (4.15)$$

- La distancia euclídea entre los puntos a y b se calcula como:

$$d = \sqrt{\sum_{j=1}^k (a_j - b_j)^2} \quad (4.16)$$

K-Means es un proceso iterativo para ajustar mediante convergencia los *clusters* [125]. Este método se refleja en el Algoritmo 1.

Algoritmo 1 K-Means

- 1: Definir el número de *clusters* (k).
 - 2: Elegir aleatoriamente k puntos del *dataset* D como centroides.
 - 3: $\forall x \in D \rightarrow$ Comprobar la similitud de coseno o la distancia euclídea con los centroides.
 - 4: $\forall x \in D \rightarrow x \in \text{cluster}$ más cercano.
 - 5: Media del *cluster* = nuevo centroide.
 - 6: Repetir los pasos 3, 4 y 5 hasta que la variación de los nuevos centroides y los de la iteración anterior sea menor que una tolerancia predefinida.
-

Como se puede observar, es necesario estudiar previamente a la aplicación del algoritmo el número de *clusters* que se van a formar. La importancia de esta decisión radica en que si la configuración es errónea, el proceso no se realizará correctamente, por exceso o por defecto del número de grupos. Igualmente es relevante llevar a cabo un proceso de normalización de los datos, ya que K-Means utiliza distancias para generar los *clusters*, y puede resultar en mayor peso para los datos con valores numéricos mayores, que no permite un proceso de agrupamiento adecuado.

Entre las ventajas de este algoritmo destacan la facilidad de implementación, la capacidad de escalado y adaptabilidad, así como, la garantía de convergencia; mientras que como aspectos negativos, resaltan la necesidad de elegir el número de *clusters* manualmente o la dependencia hacia los valores que se elijan inicialmente como centroides.

Clustering jerárquico

Este método de generación de *clusters* [108] se basa en la formación de grupos de manera secuencial. Este procedimiento se puede abordar desde dos perspectivas: aglomeración o división. El primero (Algoritmo 2) asume que inicialmente cada dato forma un *cluster* y, en cada iteración, los grupos que se parecen, se unen hasta que se alcance el número indicado de grupos (k).

Algoritmo 2 *Clustering* Jerárquico por Aglomeración

- 1: $\forall x \in D \rightarrow x$ es un *cluster* individual.
 - 2: Construir la matriz de distancias (M) con las distancias euclídeas de cada par de grupos.
 - 3: Los *clusters* más parecidos se unen.
 - 4: Actualizar M con los nuevos *clusters* y las distancias con los demás.
 - 5: Repetir los pasos 3 y 4 hasta alcanzar el número pre-definido de *clusters* (k).
-

En el caso del *clustering* jerárquico por división (Algoritmo 3) [97], el procedimiento es similar al descrito anteriormente, pero en sentido inverso. Inicialmente, un único grupo contiene todos los datos y, en cada iteración, se van dividiendo hasta alcanzar el número identificado (k).

Algoritmo 3 *Clustering* Jerárquico por División: *Bisection K-Means*

- 1: $\forall x \in D \rightarrow x$ pertenece a un único *cluster*.
 - 2: La suma de errores cuadráticos se calcula para cada *cluster*.
 - 3: El valor más alto se divide por K-Means.
 - 4: Repetir los pasos 2 y 3 hasta alcanzar el número pre-definido de *clusters* (k).
-

Este método es espacialmente potente cuanto entre el conjunto de datos existen relaciones jerárquicas, y permiten adaptarse a *clusters* no circulares. Sin embargo, son sensibles a valores atípicos y son muy exigentes computacionalmente.

Gaussian Mixture Modelling (GMM)

Este algoritmo [128] es un modelo de tipo maximización de las expectativas (EM), método híbrido para encontrar la máxima verosimilitud de un modelo optimizando sus parámetros. Parte de que todos los puntos son parte de una mezcla de distribuciones normales de parámetros desconocidos. Este algoritmo no asigna una probabilidad de pertenecer a cada *cluster*, ya que un punto puede pertenecer a varios grupos a la vez.

Los *clusters* creados con este método siguen una distribución gaussiana definida por dos parámetros, la media (μ) y la desviación estándar (σ o Σ). Para maximizarlos mediante el algoritmo EM, que consta de dos pasos (Ecuaciones 4.17, 4.18 y 4.19).

- Paso E para calcular la probabilidad de que un punto (x_i) pertenezca a cada *cluster* (c_k).

$$r_{ic} = \frac{\text{Probabilidad de } x_i \text{ perteneciente a } c}{\sum \text{Probabilidades de } x_i \text{ perteneciente a } c_1, c_2, \dots, c_k} \quad (4.17)$$

- Paso M para actualizar los valores de μ , Σ y de Π , que representa la densidad de puntos de una distribución.

$$\Pi = \frac{\text{Número de puntos asignados a cada cluster}}{\text{Número total de puntos}} \quad (4.18)$$

$$\mu = \frac{\sum r_{ic}x_i}{\text{Número de puntos asignados a un cluster}} \quad (4.19)$$

Ambos pasos se realizarán iterativamente, optimizando los parámetros y maximizando la función de verosimilitud asociada. El método GMM se resume en el Algoritmo 4.

Algoritmo 4 GMM

- 1: Selección del número de *clusters* (k).
 - 2: Asignación aleatoria de valores a los parámetros de las distintas distribuciones.
 - 3: Cálculo de la verosimilitud de las gaussianas con los datos del *dataset*.
 - 4: Maximizar la función logarítmica de verosimilitud optimizando los parámetros.
 - 5: Repetir los pasos 3 y 4 hasta que se complete el número indicado de iteraciones o se alcance la tolerancia dada.
-

Entre las ventajas de este método destaca la flexibilidad, asignando probabilidades en lugar de una etiqueta fija, y la elasticidad en el número y forma de los *clusters*. Sin embargo, es sensible a los valores iniciales, puede converger a un mínimo local como solución no óptima, o puede divergir y encontrar soluciones con infinitas probabilidades.

4.4.2 Reducción dimensional

Los algoritmos no supervisados de reducción de dimensión [72] permiten analizar la importancia de las características, seleccionando las que tienen un efecto relevante en los cálculos o permiten la visualización de los grupos sin perder información.

Principal Component Analysis (PCA)

El método estadístico de análisis de los componentes principales o PCA [72] se aplica como una primera aproximación para establecer cuanta información aporta cada característica al modelo de aprendizaje automático y así elegir el número de dimensiones que podemos reducir sin perder demasiada información. Se puede abordar desde el estudio de la acumulación de varianzas en las características del conjunto de datos, ya que cuanto mayor es la varianza, más información ofrece esa característica. Así, el objetivo final es encontrar las variables más relevantes, calculando los autovalores y autovectores de la matriz de covarianza y seleccionar los autovectores con mayor autovalor (Algoritmo 5).

Algoritmo 5 PCA

- 1: Calcular la matriz de covarianza (C).
 - 2: Calcular autovalores y autovectores de C .
 - 3: Seleccionar los m autovectores con mayor autovalor, siendo m la dimensión a la que reducir C .
 - 4: Proyectar los datos en los autovectores seleccionados
 - 5: **Resultado:** datos reducidos a m dimensiones.
-

Este método minimiza la pérdida de información reduciendo el ruido, a la vez que identifica las características más importantes, pero no funciona con datos correlacionados linealmente, y su visualización no es fácilmente interpretable.

ISOMAP

Este algoritmo de reducción dimensional [136] pertenece al denominado *manifold learning*, un espacio matemático donde se recrea localmente un espacio euclídeo, no a nivel global. Los puntos del conjunto de datos están condicionados por un hiperplano de una forma concreta, lo que evita que la distancia entre dos puntos sea necesariamente una línea recta, y por tanto no se puede considerar una medida de su similitud. Para conocerla, sería necesario recorrer el espacio dimensional y medir la distancia mediante una geodésica del mismo (distancia mínima entre dos puntos dentro de un espacio).

El algoritmo ISOMAP asume que los datos pertenecen a un *manifold* por lo que la reducción dimensional no será de forma lineal, manteniendo las geodésicas al proyectarlas en una dimensión menor. Para conseguirlo, se crea un grafo con la forma del *manifold* a partir de algoritmos de agrupamiento como *K-Nearest Neighbours* (KNN). Con la red formada, se calcula la geodésica de la distancia de los nodos en el grafo. Después, se usan los autovalores y autovectores para proyectar los autovectores con mayor autovalor y así, reducirlos dimensionalmente (Algoritmo 6).

Algoritmo 6 ISOMAP

- 1: Determinar los vecinos de cada punto.
 - 2: Construir el grafo de *manifold*, conectando cada punto con sus vecinos más cercanos.
 - 3: Calcular la distancia mínima entre dos nodos usando el algoritmo de Dijkstra, de donde se obtiene una matriz con las distancias geodésicas de los puntos del *manifold*.
 - 4: Proyección de los datos: se eleva al cuadrado la matriz de distancias, se realiza doble centrado, y se descompone en autovalores la matriz para reducir la dimensión.
-

Este algoritmo tiene una complejidad de $O(N^2)$, lo que requiere una gran capacidad computacional con números de puntos elevados. Para evitarlo, se utiliza un algoritmo que traduce las distancias entre los puntos y los mapea sobre un espacio cartesiano.

Entre las ventajas del algoritmo ISOMAP destaca la capacidad de realizar reducciones dimensionales con no linealidades, mientras que depende de la elección correcta del parámetro k , que representa el número de vecinos más cercanos.

T-Distributed Stochastic Neighbor Embedding (t-SNE)

Este algoritmo de reducción de dimensión no lineal [74] está destinado principalmente a la visualización de conjuntos de datos de alta dimensión. t-SNE atrae datos similares y repele los que no se parecen, formando grupos donde los datos de alta dimensión se transforman y agrupan con datos que se estima que son parecidos entre sí, en una dimensión menor (Algoritmo 7).

Algoritmo 7 t-SNE

-
- 1: Medir la similitud de los datos en el espacio de alta dimensión. A cada punto se le asigna una distribución gaussiana con una desviación estándar dada. Los puntos cercanos tendrán un valor alto de densidad en esas distribuciones, mientras que los puntos separados tendrán densidades bajas.
 - 2: Construir la matriz de similitud en el espacio de alta dimensión.
 - 3: Los datos se proyectan aleatoriamente a un espacio de menor dimensión.
 - 4: Se calcula la similitud de los datos en este espacio de menor dimensión.
 - 5: Construir la matriz de similitud en el espacio de menor dimensión.
 - 6: Tratar de hacer los valores de esta segunda matriz lo más parecidos posibles a los de la primera matriz aplicando la métrica de divergencia Kullback-Leibler y el descenso del gradiente, que agrupa puntos similares y los separa de los demás.
-

El resultado es una representación de los datos teniendo en cuenta las posibles distribuciones que pueden darse en el espacio de alta dimensión.

Además, este algoritmo tiene asociado un hiperparámetro (*perplexity*) [27] que determina el valor de la desviación estándar de las distribuciones utilizadas para llevar a cabo el cálculo de la similitud.

Tiene la ventaja de realizar la reducción de dimensión teniendo en cuenta la distribución de los puntos, mientras que el coste computacional es muy alto, y el hiperparámetro debe ajustarse correctamente para que el algoritmo funcione adecuadamente.

Uniform Manifold Approximation and Projection (UMAP)

Este reductor dimensional [78] se basa en generar gráficos a partir de los datos que pertenecen a un espacio de alta dimensión y los asigna siempre a otro gráfico de menor dimensión.

Una de las mejoras que plantea este algoritmo [38] es la forma de construir los grafos de alta dimensión: un radio de tamaño variable que se extiende sobre cada punto. Cuando los radios intersecan, se conectan los puntos. El tamaño de los radios se define en función de la densidad del área en que se encuentra el dato, que se determina mediante un algoritmo como KNN y un acuerdo sobre qué se entiende por alta densidad en una zona. Tras realizar las conexiones y determinar la densidad de las zonas, las relaciones entre los datos se ponderan en función del área en que se encuentren. A continuación se realiza otro grafo en las dimensiones a las que se quiere reducir el resultado. Los datos con conexiones muy ponderadas se mantienen juntos mientras que los demás tienden a separarse. Pueden ocurrir rotaciones en las proyecciones pero la estructura de los datos se mantiene.

Los hiperparámetros principales que condicionan el funcionamiento de este algoritmo son el número de vecinos considerados para crear el grafo de alta dimensión, y la distancia entre puntos en el grafo de baja dimensión. El primero controla cómo UMAP equilibra entre mantener la estructura local del *dataset* y la estructura global, ya que un número de vecinos bajo mejora la representación de estructuras locales, mientras que valores altos potencian la representación global. Por otro lado, la distancia mínima define la compresión de los grupos

de puntos: valores bajos harán *clusters* estrechos, y distancias mayores harán *clusters* más anchos, preservando la estructura amplia del *dataset*.

Como alternativa a t-SNE presenta ventajas como la eficiencia o el balance entre la representación de estructuras locales y globales, pero la representación es menos fiel, al tener que ajustar dos hiperparámetros.

4.4.3 Métricas

Un aspecto principal en el diseño y evaluación de modelos de aprendizaje es comprobar su rendimiento, funcionamiento y precisión, realizando distintas pruebas para validar el resultado real del algoritmo mediante la aplicación de una métrica que permita medir una de las propiedades o comparar modelos.

En aprendizaje no supervisado es complicado estimar si el resultado de un modelo es bueno, al no tener una verdad absoluta para compararlo, pero existen métricas para valorar cómo funciona el modelo de *clustering*.

Within Set Sum of Squared Error (WSSSE)

Para saber qué tal se agrupan los datos tras la ejecución de un algoritmo, se puede obtener una primera impresión mediante la visualización en un diagrama de dispersión, teniendo en cuenta el tipo de dimensión de los datos, ya que si existen más de tres características es necesario reducir la dimensión antes de representarlos, y por tanto se pierde información.

Para evitarlo, existe una métrica que permite conocer el error total de las distancias de los datos en relación con el centroide de su *cluster*.

Matemáticamente se representa en la Ecuación 4.20 [145], siendo k el número de grupos y S_j el conjunto de datos del *cluster* j . Así, se calcula la distancia entre cada punto del *cluster* j (x_{ij}) y el centroide del grupo j (\bar{x}_j).

$$WSSSE = \sum_{j=1}^k \sum_{i \in S_j} \|(x_{ij} - \bar{x}_j)\|^2 \quad (4.20)$$

El resultado es el error de los procesos de agrupamiento, que permite optimizar hiperparámetros como el valor de k , la métrica de distancia o la tolerancia. El objetivo será reducir esta métrica lo más posible sin sobre-ajustar el modelo, utilizando métodos como el ‘punto de codo’.

Silhouette

Es otra medida que analiza la formación de grupos [109]. Permite conocer la cohesión de un *cluster* y la separación con los demás. Para ello se debe calcular la distancia entre cualquier punto y los que pertenecen a su *cluster* (a) y la distancia entre cualquier punto y el resto de puntos de los grupos más cercanos (b).

El valor de la métrica Silhouette debe encontrarse en el rango [-1, 1], donde valores altos (cercaos al 1) indican que los datos están correctamente asignados (alta cohesión del *cluster*

y separación con los otros).

La expresión matemática para calcularlo (Ecuación 4.21) incluye las medias expresadas anteriormente (a y b) para el punto i .

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \quad (4.21)$$

El método para optimizar el número de *clusters* es similar al utilizado para WSSSE, buscando el codo de la gráfica. Ya que ambos métodos permiten optimizar este hiperparámetro, y como Silhouette proporciona información más precisa sobre la formación de los grupos, prevalece la optimización alcanzada mediante esta métrica antes que utilizando los errores.

4.5 Conclusiones

A partir de la integración de la IA en el campo de la ciberseguridad se ha llevado a cabo una revolución en tácticas tanto de ataque como de protección de los sistemas y las redes. Las nuevas tecnologías han proporcionado un enfoque más completo para los desafíos que surgen en la defensa frente a ciberataques. Al combinar IA y aprendizaje automático se obtiene un entorno robusto y flexible capaz de identificar y abordar las amenazas en tiempo real.

Para la definición de ontologías, los lenguajes de la Web Semántica cuentan con la ventaja de trabajar sobre una sintaxis y semántica formales que las máquinas sean capaces de interpretar, dotando a las ontologías de expresividad en la representación de conceptos y favoreciendo el intercambio de información y la reutilización de los modelos, ya que pueden incorporar conceptos de otras ontologías creadas anteriormente. Entre ellos, destacan RDFS y OWL Full, siendo este último el que mejor se adapta a la propuesta de esta Tesis Doctoral por su expresividad y compatibilidad con otros estándares y recomendaciones del W3C.

Por otro lado, en conjunto, SPIN presenta una flexibilidad y expresividad para definir el comportamiento esperado en el modelo de datos e inferir información que soluciona la limitación presente en OWL para llevar a cabo la inferencia, y las limitaciones de SWRL en el manejo de los datos. Esto, unido a su facilidad de uso y la integración con OWL Full, lo convierte en la mejor herramienta para establecer el comportamiento del entorno de conciencia cibersituacional que se presenta en esta Tesis Doctoral, complementándose con el razonador semántico HermiT. Esta elección se basa en la compatibilidad con el lenguaje de programación básico del entorno propuesto, Python, y por tener los mejores resultados al trabajar con ontologías complejas, como la propuesta en esta Tesis Doctoral.

En la otra rama de la IA, el aprendizaje automático, por ser un campo muy amplio y desarrollado, a lo largo de este capítulo se han descrito las tecnologías principales que se utilizan en la propuesta. Dentro de los algoritmos supervisados, destacan en este tipo de tareas los basados en árboles de decisión, remarcando la importancia de un correcto pre-procesado de los datos y la evaluación del rendimiento de modelos con métricas extraídas de la matriz de confusión o las curvas ROC. En el ámbito de los modelos no supervisados, el enfoque es distinto, presentando los algoritmos de *clustering* como los óptimos para este tipo de tareas, combinados con técnicas de reducción dimensional para poder visualizar los resultados y

evaluados mediante métricas como Silhouette, que permiten establecer la calidad de los grupos formados por el modelo a partir de los datos de entrada.

En conjunto, todas estas tecnologías proporcionan la base de una posible respuesta a los problemas de seguridad de los sistemas, conformando un entorno de conciencia cibersituacional adaptable y eficaz en la protección de los activos y la reducción de los riesgos generados a partir de los ciberataques en la actualidad. En los próximos capítulos se detallará el enfoque propuesto en esta Tesis Doctoral que utiliza las tecnologías presentadas en este capítulo para alcanzar los objetivos definidos en la investigación.

Capítulo 5

Gestión de riesgos

Como parte de los entornos de conciencia cibersituacional, la gestión de riesgos es una parte fundamental, ya que permite analizar las amenazas y riesgos asociados a los incidentes que se detectan, y además permite establecer las políticas de acción para hacerles frente. Tras una breve introducción en la Sección 5.1, se presentan en detalle las metodologías más utilizadas a nivel europeo, terminando la Sección 5.2 con una comparación entre ellas para localizar puntos en común que permitan la interoperabilidad. Finalmente, en la Sección 5.3, se presentan las conclusiones extraídas a lo largo del capítulo.

5.1 Introducción

El objetivo principal de los marcos para la gestión de riesgos es coordinar las actividades de identificación y evaluación que puedan afectar a una organización. Buscan crear y proteger el valor, por lo que deben incluir un plan para reducir y controlar los riesgos y el efecto en sus activos [131].

Los riesgos se relacionan con pérdidas mediante la explotación de vulnerabilidades hasta lograr una intrusión, por lo que la seguridad es un pilar para responder ante las amenazas y también para poder prevenirlas. Con el objetivo de llevar a cabo una gestión efectiva, los procesos deben estar integrados en las actividades de la organización, deben ser estructurados, entendibles y adaptados a las necesidades propias, pero sobre todo deben ser dinámicos, ya que los riesgos surgen, cambian o desaparecen, y la gestión de riesgos debe anticiparse, detectar y responder a estos eventos adecuadamente [131]. Por este motivo, y basándose en las directrices que proporciona la Organización Internacional de Normalización (*International Organization for Standardization*, ISO), han surgido distintos estándares y metodologías que proponen una aproximación a la gestión y evaluación de riesgos desde distintos puntos de vista.

Según la norma ISO 31000, el marco de gestión de riesgos debe desarrollar un plan para enfrentarse a esos riesgos, identificar qué decisiones deben tomarse. Se divide en distintos procesos [131]: En primer lugar, la identificación de riesgos, que consiste en identificarlos y describirlos (causas, amenazas, vulnerabilidades, activos, etc.). Para seguir, se analizan,

definiendo la probabilidad de ocurrencia y el impacto de la materialización de los riesgos (pérdida de confidencialidad, integridad y disponibilidad), su naturaleza y, en caso de que existan contramedidas, su efectividad. Después, el proceso de evaluación de riesgos de seguridad de la información establece y mantiene criterios como la aceptación de riesgos. También determina los rangos de los niveles de riesgo y los clasifica, los compara con el criterio establecido y los prioriza para aplicar el tratamiento de riesgos. Aquí, la gestión de riesgos actúa como soporte a la toma de decisiones, valorando si cada uno debe eliminarse, mitigarse, transferirse o aceptarse [132].

Los conceptos principales [37] de todas las metodologías de gestión de riesgos son: activos, amenazas, vulnerabilidades, impacto, probabilidad y riesgo.

- Activos: cualquier recurso de la organización del que dependa su actividad principal, y cuyo deterioro implique un daño sobre ella.
- Amenaza: circunstancia desfavorable que, si ocurre, tiene un impacto negativo sobre los activos (pérdida de valor).
- Vulnerabilidad: debilidad en los activos que favorece la materialización de amenazas.
- Impacto: consecuencia de la materialización de la amenaza sobre el activo, aprovechando una vulnerabilidad.
- Probabilidad: frecuencia de ocurrencia de una amenaza, basándose en datos objetivos o análisis de expertos.
- Riesgo: estimación de lo que puede ocurrir, valorado cuantitativamente como el producto del impacto y la probabilidad.

Estos términos se repetirán en las metodologías más utilizadas, al igual que en la propuesta desarrollada en esta Tesis Doctoral.

5.2 Estándares y metodologías

5.2.1 EBIOS

Expressions des Besoins et Identification des Objectifs de Sécurité (EBIOS) [40, 45, 48, 121] es una metodología francesa creada en 1995 para evaluar y abordar los riesgos de la seguridad de la información. Proporciona un marco para la gestión de riesgos que incluye la instalación de un sistema de gestión acompañado de una estrategia de seguridad y su integración en distintos proyectos. Esta metodología se divide en cinco talleres que se aplican de manera iterativa, como se observa en la Figura 5.1 [121].

1. Estudio del contexto: proporciona herramientas para definir el estudio de riesgos, identificar, delimitar y describir el sistema en estudio, su ecosistema y definir las líneas básicas de ciberseguridad.
2. Identificación del origen del riesgo: permite que los riesgos se caractericen en duplas (origen del riesgo, origen del objetivo) que se evaluarán en los módulos 3 y 4.

3. Análisis de riesgos a nivel de los activos primarios: proporciona herramientas para analizar los riesgos identificados anteriormente, evaluando escenarios estratégicos de origen de riesgo, asignando a cada uno un nivel de gravedad.
4. Análisis de riesgos a nivel de los activos de soporte: proporciona herramientas para estudiar más a fondo el riesgo, analizando escenarios operativos en términos de probabilidad porcentual.
5. Evaluación, tratamiento y aceptación de riesgos: permite la evaluación de los riesgos identificados y analizados en los módulos 2 y 3, y proporciona medios para abordarlos y decidir si aceptar o no los riesgos residuales.

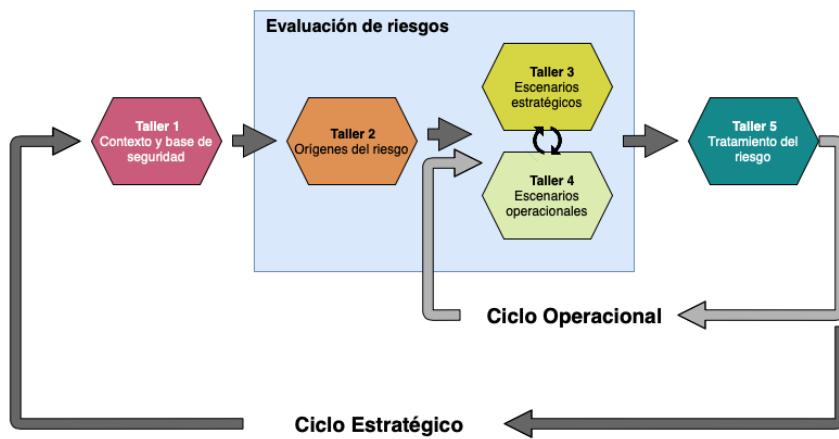


Figura 5.1: Esquema del proceso de análisis y gestión de riesgos EBIOS

Destaca por su flexibilidad y su capacidad de identificar los bloques que componen el riesgo, a diferencia de otras metodologías que solo determinan los elementos de escenarios predefinidos. Sin embargo, esta metodología es de autoevaluación y es subjetiva.

5.2.2 MAGERIT

La Metodología de Análisis y Gestión de Riesgos de los Sistemas de la Información (MAGERIT) [40, 60] es un marco de gestión de riesgos desarrollado por el Consejo Superior para la Administración Electrónica del Gobierno Español con el objetivo de reducir los riesgos de la implementación y uso de Tecnologías de la Información en el sector público. El principal objetivo es concienciar a las organizaciones y administraciones de la existencia de estos riesgos, proporcionando un método sistemático de análisis que incluya un plan de tratamiento de los riesgos.

Está compuesto por tres libros: “Método”, “Catálogo de Elementos” y “Guía de Técnicas”. El primero cubre los procedimientos de análisis de riesgos y gestión de riesgos, el segundo contiene catálogos de activos, amenazas o salvaguardas, y finalmente el último es una guía técnica sobre la base legal, la conceptualización y el propósito del análisis.

MAGERIT es una metodología cualitativa basada en activos con una herramienta para analizar riesgos de sistemas de información, PILAR (Procedimiento Informático-Lógico para

el Análisis de Riesgos), desarrollado por el Centro Nacional de Inteligencia.

Los procesos que esta metodología lleva a cabo para la gestión de riesgos son las siguientes y se representan en la Figura 5.2 [59].

1. Caracterización de Activos: identificación de activos relevantes, dependencias y su valoración según su relevancia.
2. Caracterización de Amenazas: identificación y evaluación de amenazas, caracterizándolas según una estimación de ocurrencia o probabilidad y el daño causado o degradación.
3. Caracterización de Salvaguardas: se divide en la identificación de salvaguardas relevantes y su evaluación. El objetivo de este proceso es conocer lo necesario para proteger el sistema y supervisar los riesgos para limitar el daño causado.
4. Estimación del Estado de Riesgo: el propósito es obtener un cálculo supuesto de lo que se espera que ocurra y lo que es probable que ocurra.
5. Evaluación y Tratamiento de Riesgos Residuales: dependiendo de la evaluación del riesgo residual, se inicia el tratamiento enfocado en reducirlo o ampliarlo.
6. Plan de Seguridad: proyectos en los que se materializan las decisiones adoptadas para el tratamiento de riesgos. Consta de tres tareas: identificación del proceso de seguridad, planificación e implementación, que servirán como ayuda para tratamientos posteriores o como fuente para estándares.

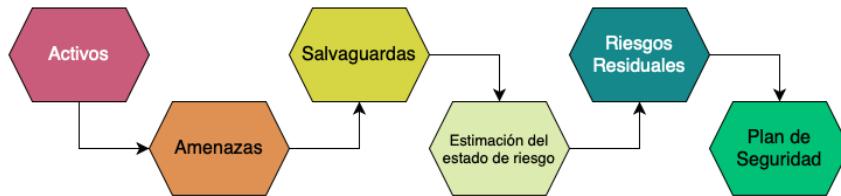


Figura 5.2: Esquema del proceso de análisis y gestión de riesgos MAGERIT

Ofrece un enfoque sistemático para procesos de evaluación, auditoría, certificación o acreditación, pero evalúa todo desde el punto de vista económico, por lo que debe ser traducido a partir de otros tipos de valoraciones.

5.2.3 MONARC

Method for an Optimized aNALysis of Risks by Cases (MONARC) [39, 40] es una herramienta de gestión de riesgos que capitaliza el riesgo en función de análisis previos. Se basa en una biblioteca de modelos de riesgo que ofrecen diferentes escenarios según los activos. Está dividido en cuatro fases, que aparecen en la Figura 5.3 [39].

1. Establecimiento del Contexto: identificación de actividades clave, procesos comerciales críticos, así como sus posibles amenazas y vulnerabilidades mediante un método de evaluación cualitativa.

2. Modelado del Contexto: los activos identificados se representan en un diagrama que muestra sus dependencias y las amenazas o vulnerabilidades que los afectan.
3. Evaluación y Tratamiento de Riesgos: se utiliza un método de cuantificación para estimar el riesgo del sistema, sobre el que se implementan las medidas necesarias y se despliega un plan para reducir el riesgo. Los distintos riesgos se organizan según un criterio, que permite compararlos de acuerdo con un umbral de aceptación definido.
4. Implementación y Monitorización: el objetivo es optimizar la seguridad y ampliar el alcance del análisis de riesgos mediante una verificación recurrente de la seguridad del sistema.

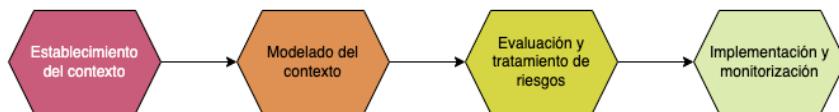


Figura 5.3: Esquema del proceso de análisis y gestión de riesgos MONARC

MONARC aprovecha análisis previos de otros sistemas en el mismo entorno empresarial, partiendo de que sufrirán ataques similares en estructuras de activos similares, y obteniendo escenarios de riesgo generalizados para cada tipo de negocio.

5.2.4 ITSRM

IT Security Risk Management Methodology (ITSRM) [40, 41] es una metodología que forma parte de un conjunto de normas para la seguridad de la información establecidas por la Dirección General de Informática de la Comisión Europea. Este marco consta de siete procesos (Figura 5.4) [41].

1. Caracterización de la Seguridad del Sistema: consiste en recopilar información sobre el sistema que se utilizará en otros procesos.
2. Activos Primarios: identificación de los activos cruciales (principalmente datos y funciones) para la organización en la consecución de sus objetivos comerciales; y de sus posibles atacantes.
3. Activos de Soporte: identificación de aquellos activos utilizados en la gestión de los activos primarios.
4. Modelado del Sistema: desarrollo de un modelo de asociación entre los activos principales y los activos de soporte, siguiendo el flujo de datos y la arquitectura del sistema.
5. Identificación de Riesgos: en este paso, se construyen escenarios de riesgo, representando los riesgos de los activos principales y sus consecuencias en sí mismos y en los activos de soporte. El objetivo es determinar los riesgos que se analizarán, evaluarán y abordarán en los procesos siguientes.
6. Análisis y Evaluación de Riesgos: este proceso calcula el nivel de riesgo residual en los escenarios definidos, según una lista de medidas de seguridad definidas para mitigar

esas amenazas.

7. Tratamiento de Riesgos: en este paso, se seleccionan las medidas más apropiadas para responder a los riesgos identificados, teniendo en cuenta las limitaciones de la organización.

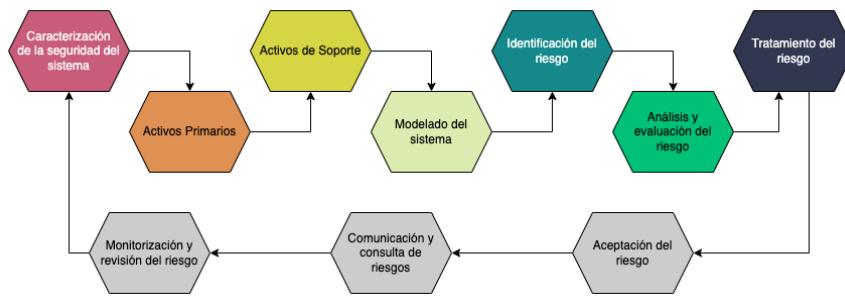


Figura 5.4: Esquema del proceso de análisis y gestión de riesgos ITSRM

Esta metodología utiliza estándares ISO altamente adaptables y proporciona un marco común para la gestión de riesgos. También ofrece catálogos de amenazas, medidas de seguridad, limitaciones, posibles adversarios y activos de soporte. Sin embargo, aún no es una solución completa y su implementación es compleja.

5.2.5 CRAMM

CCTA Risk Analysis and Management Method (CRAMM) [47] realiza un análisis cualitativo de riesgos propuesto por el gobierno del Reino Unido y cuenta con una herramienta para llevar a cabo el proceso. El objetivo principal es justificar las inversiones realizadas en seguridad al demostrar cuantitativamente la necesidad de acción. Las fases de CRAMM son las siguientes (Figura 5.5).

1. Identificación y Valoración de Activos: los tres tipos de activos evaluados en esta metodología son datos, software de aplicación y activos físicos.
2. Evaluación de Amenazas y Vulnerabilidades y Cálculo de Riesgos. Se estudian según los grupos de activos seleccionados. Además, se estima el riesgo de cada grupo de activos en función de sus amenazas y vulnerabilidades.
3. Identificación y Recomendación de Contramedidas: se seleccionan una serie de contramedidas aplicables al sistema, evaluándolas positivamente si protegen contra más de una amenaza, no hay otras alternativas, tienen el menor coste, son más efectivas o si previenen incidentes.

Este marco tiene en cuenta todas las etapas del ciclo de vida de un sistema, cuenta con una amplia base de datos de contramedidas que se actualiza con frecuencia, permite revisiones y crea conciencia sobre la necesidad de ciberseguridad, pero requiere la intervención de un profesional cualificado para su implementación.

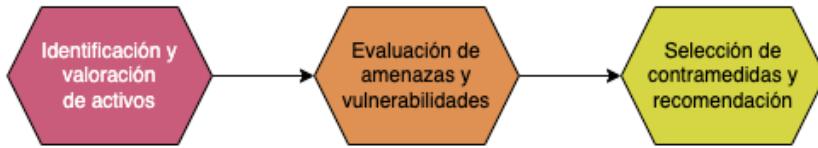


Figura 5.5: Esquema del proceso de análisis y gestión de riesgos CRAMM

5.2.6 Comparación de metodologías

Con el objetivo de encontrar puntos comunes entre los diversos marcos analizados, presentaremos un análisis de las conclusiones del estudio llevado a cabo por ENISA [40, 41], con el fin de desarrollar una metodología interoperable.

En primer lugar, se analizan los aspectos generales de los marcos considerados:

- Análisis cuantitativo o cualitativo, dependiendo de si la evaluación se basa en percepciones o en datos disponibles y verificables.
- Gestión de riesgos basada en activos o en procesos, ya sea centrada en evaluar activos y las amenazas y vulnerabilidades a las que están expuestos para estimar el riesgo, o si se centra en situaciones o procesos dentro del sistema que pueden ser explotados por atacantes.

La comparación de las metodologías teniendo en cuenta estos puntos se muestra en la Tabla 5.1. En ella, podemos observar que la mayoría de las metodologías se basan en activos en lugar de procesos, y casi todas realizan ambos tipos de análisis, cualitativo y cuantitativo.

Tabla 5.1: Aspectos generales de las metodologías - Comparación

Metodología Analizada	Análisis Cualitativo vs Análisis Cuantitativo	Basado en activos vs Basado en procesos
E BIOS	Ambos	Ambos
MAGERIT	Ambos	Activos
MONARC	Cualitativo	Activos
ITSRM	Ambos	Activos
CRAMM	Ambos	Activos

Para continuar, se analizan aspectos funcionales de los marcos de gestión de riesgos:

- Taxonomía de activos: clasificación sistemática de activos, analizando la categoría a la que pertenecen, si pueden ser modificados o si se pueden introducir nuevos activos.

- Evaluación de activos: si la metodología contiene pautas para evaluar activos utilizando alguna escala o criterio, y si esta escala puede ser modificada o si se pueden agregar nuevos criterios.
- Catálogo de amenazas: si la metodología cuenta con un catálogo interno de amenazas y si éste es modificable.
- Catálogo de vulnerabilidades: si la metodología cuenta con un catálogo interno de vulnerabilidades y si es adaptable.
- Cálculo de riesgo: establecimiento de un método para calcular el nivel de riesgo.
- Catálogo de contramedidas y cálculo de riesgo residual: existencia de un manual de contramedidas para hacer frente al riesgo y si hay un método para calcular el riesgo residual después de aplicar una de las medidas definidas.

Las características de las metodologías elegidas en este aspecto se resumen en las Tablas 5.2, 5.3, 5.4, 5.5 y 5.6.

Tabla 5.2: Taxonomía de activos - Comparación

Metodología analizada	Taxonomía de activos
EBIOS	Activos Primarios y Activos de Soporte.
MAGERIT	Catálogo en el Libro II. No admite nuevos tipos.
MONARC	Activos Principales y Activos Secundarios.
ITSRM	Activos Primarios y Activos de Soporte.
CRAMM	Datos, Aplicaciones Software y Activos Físicos.

Tabla 5.3: Valoración de activos - Comparación

Metodología analizada	Valoración de activos
EBIOS	Umbrales de severidad. Modificable.
MAGERIT	Criterios de evaluación de 0 a 10 en las dimensiones de seguridad de los activos.
MONARC	Evaluación como base de la implementación. No modificable.
ITSRM	Reutiliza evaluaciones, escala de impacto o evaluación formal de impacto. Modificable.
CRAMM	Escala subjetiva de impacto en las dimensiones de los activos de 0 a 10.

Tabla 5.4: Catálogo de amenazas- Comparación

Metodología analizada	Catálogo de amenazas
EBIOS	Proporciona un catálogo. Modificable.
MAGERIT	Lista por defecto en el Libro II. Admite nuevos catálogos.
MONARC	Lista predefinida y modificable.
ITSRM	Proporciona un catálogo. Admite nuevas listas.
CRAMM	Tablas de amenazas predefinidas según el tipo de activos.

Tabla 5.5: Catálogo de vulnerabilidades - Comparación

Metodología analizada	Catálogo de vulnerabilidades
E BIOS	No se especifica. Puede importar nuevos catálogos.
MAGERIT	No se especifica.
MONARC	Lista predefinida y modificable.
ITSRM	No define un catálogo. Son un componente funcional independiente.
CRAMM	Añadidas mediante entrevistas.

Tabla 5.6: Cálculo del riesgo - Comparación

Metodología analizada	Cálculo del riesgo
E BIOS	Cálculo basado en probabilidad e impacto.
MAGERIT	En el Libro III se proporcionan técnicas cualitativas y cuantitativas. No es modificable.
MONARC	Riesgo = Amenaza x Vulnerabilidad x Impacto.
ITSRM	Riesgo = Probabilidad x Consecuencia.
CRAMM	En la herramienta se calculan los riesgos en una escala de 1 a 7.

5.3 Conclusiones

La desarrollo tecnológico ha aumentado las amenazas y los riesgos del ciberespacio, dejando expuesta información confidencial, dañando la reputación de las organizaciones, y así causando daños económicos. Además, la frecuencia de estas amenazas provoca que la seguridad de las organizaciones sea imprescindible y compleja, priorizando los riesgos más altos o inminentes.

El entorno de cualquier organización implica analizar un alto volumen de información, que dificulta los distintos procesos involucrados en la gestión de riesgos, y requiere el desarrollo de sistemas que colaboren en el análisis y procesado de la información.

La necesidad para cualquier organización de tener una metodología de gestión de riesgos dinámica, estructurada y entendible, radica en la importancia de los activos que se protegen, principalmente la información. Existe mucha documentación alrededor de estos marcos, que detallan minuciosamente procesos para identificar los riesgos, analizarlos, evaluarlos y tratarlos.

Como se ha analizado a lo largo de este capítulo, al comparar diferentes marcos de trabajo podemos observar que cada uno lleva a cabo una evaluación de riesgos centrada en un área específica, de manera que la interoperabilidad, entendida como la capacidad de reutilizar la información proporcionada por componentes de otras metodologías, podría considerarse un aspecto de trabajo futuro en términos de las metodologías actuales. Conocer las experiencias de otras organizaciones, saber cómo se han enfrentado a las amenazas e incidentes, permite adaptar los procesos de gestión de riesgos a los retos existentes.

Capítulo 6

Arquitectura global del modelo propuesto

6.1 Diseño del modelo de arquitectura

En los capítulos anteriores se ha presentado el contexto en el que se enmarca la propuesta del modelo de caracterización de ataques para entornos de conciencia cibersituacional de esta Tesis Doctoral.

A partir de los objetivos definidos, y tras completar las tareas correspondientes al estudio del estado del arte, en los siguientes apartados se desarrollarán los distintos módulos que permiten la caracterización de ciberataques en entornos con fuentes de datos heterogéneas.

Dado el panorama actual de la ciberseguridad, en el que la evolución es continua y en el que se generan nuevos tipos de ataque con frecuencia, existe una tendencia en los mecanismos de defensa consistente en derivar a la tecnología tareas recurrentes y llevar a cabos procesos reactivos con un enfoque de *Cyber Threat Hunting* para la detección de ciberataques.

Las limitaciones de los sistemas defensivos tradicionales, que analizan matemáticamente los datos con algoritmos o reglas, generan un problema a la hora de detectar los ataques más complejos, que pasan desapercibidos. Es en este punto donde aparece la necesidad de recopilar información del tráfico existente en la red en busca de signos de estos incidentes, de forma que al conseguir una caracterización del ciberataque, el impacto alcance todos los ámbitos de la ciberseguridad: se pueden desarrollar nuevos sistemas para detectarlos, la gestión de riesgos dispone de información enriquecida sobre los mismos para estimar el impacto que causan sobre el sistema, y la respuesta automática para mitigar estas consecuencias se puede adaptar y optimizar así la reducción del riesgo.

Con este objetivo se diseña el modelo presentado en la Figura 6.1, donde la arquitectura incluye distintos módulos que abarcan los objetivos definidos en esta Tesis Doctoral. Cada uno de los sistemas identificados que lo componen se presentarán a continuación pero su explicación se detallará en los próximos capítulos y serán validados en conjunto en el Capítulo 10.

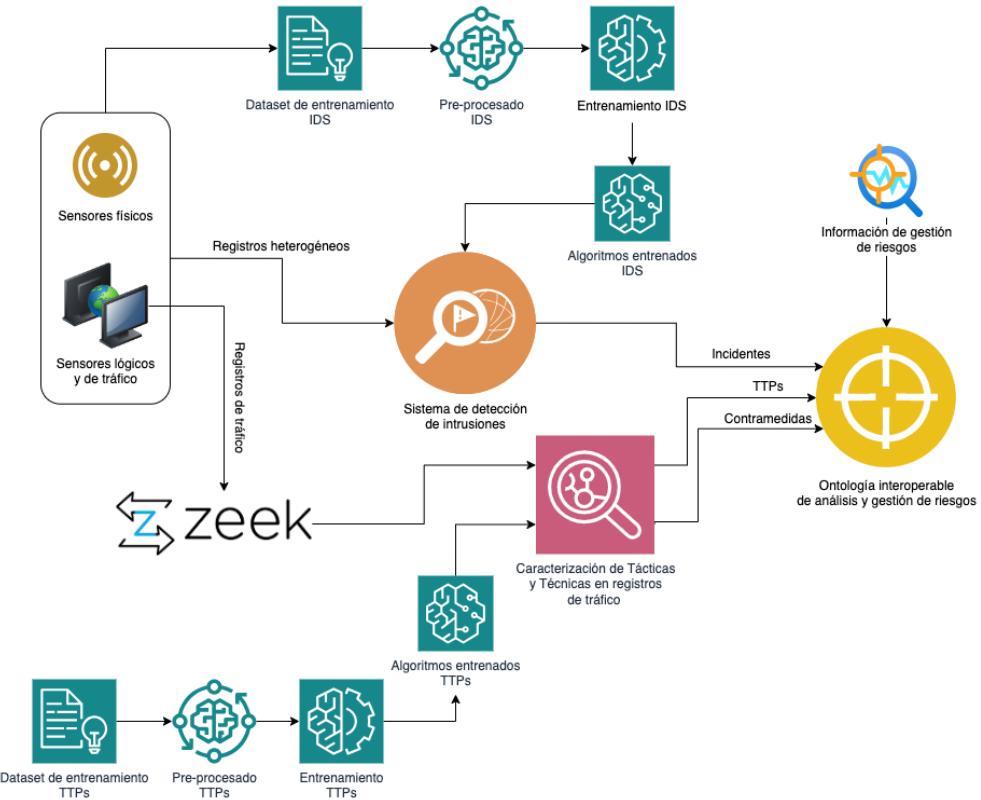


Figura 6.1: Arquitectura global del modelo propuesto

6.1.1 Sistema de detección de intrusiones

Este módulo, representado en naranja en la figura y que se describe en profundidad en el Capítulo 7, permite analizar el tráfico procedente de fuentes heterogéneas, cada vez más presentes en la actualidad. Para componerlo, por un lado se requiere un conjunto de dispositivos físicos y lógicos que generen datos para analizar (Objetivo 1 de esta Tesis Doctoral). Una parte de estos datos se utilizará como *dataset* de entrenamiento, a partir del que se definirá el pre-procesado que necesitan para que el modelo los acepte (Objetivo 2). Finalmente se eligen y entranan un conjunto de modelos de aprendizaje automático no supervisado que aprendan a discernir entre los datos aquellos que representen anomalías en relación con el comportamiento normal de la red (Objetivo 3). Estos modelos entrenados luego se utilizarán para la detección en tiempo real, siendo la salida de este sistema la identificación de incidentes o ciberataques entre el tráfico recibido.

6.1.2 Sistema de caracterización de TTPs

Por otra parte, profundizando en el análisis del tráfico de red, en la propuesta de esta Tesis Doctoral se define el módulo que lleva a cabo la caracterización de técnicas MITRE ATT&CK, en rosa en la figura. En este caso, como se detallará más adelante en el Capítulo 8, para preparar los modelos de IA se elige un conjunto de datos de entrenamiento que contenga registros etiquetados con TTPs. Estos datos se preparan correctamente y se entranan los

modelos para realizar la clasificación entre tráfico benigno y las técnicas que represente el malicioso (Objetivo 4 de la Tesis Doctoral).

Estos modelos entrenados se pueden aplicar en la caracterización en tiempo real de tráfico capturado en el entorno por un SIEM, con el requisito de que debe ser enriquecido con una herramienta como Zeek para conseguir los campos de con los que se han preparado los algoritmos supervisados. De este módulo se extrae principalmente las tácticas y técnicas utilizadas por el atacante que se asocian con el tráfico recopilado. Esta información es muy valiosa, ya que permite conocer en qué fase se encuentra un ataque, extraer información acerca de las debilidades explotadas por las tácticas y plantea una serie de contramedidas para responder ante el ciberataque, completando la caracterización (Objetivo 5).

6.1.3 Ontología interoperable para la gestión de riesgos

Finalmente, los incidentes detectados por el primer módulo y la información extraída del segundo se recogen en una ontología que lleva a cabo el proceso de gestión dinámica de riesgos basándose en una metodología interoperable, representada en color amarillo y que se define en el Capítulo 9. Además, las caracterizaciones llevadas a cabo hasta este punto se complementan con información propia de la gestión de riesgos, como los activos o los escenarios de riesgo. Esta ontología tiene la capacidad de realizar traducciones de un conjunto de metodologías ampliamente adoptadas, como MAGERIT, EBIOS o CRAMM, a una escala común, ITSRM, de forma que se puedan comparar resultados y compartir información. Esto implica que los catálogos de entrada se deben adaptar a la metodología elegida, razonando sobre todos los datos para obtener un nivel de riesgo. La salida de este sistema será un nivel de riesgo potencial para las amenazas identificadas y otro del entorno a nivel global. Además, al realizar una recomendación para la respuesta automática frente al incidente según un catálogo de contramedidas y las mitigaciones propuestas por MITRE para hacer frente a las técnicas identificadas, se calcularán los riesgos residuales correspondientes (Objetivo 6 de la Tesis Doctoral).

El funcionamiento del modelo de arquitectura global propuesto se define a partir del objetivo principal y contemplando las interconexiones entre los distintos módulos individuales (Objetivo 7).

Capítulo 7

Propuesta para la detección automática de ciberataques en registros heterogéneos

A lo largo de este capítulo se presenta el diseño y desarrollo de un sistema para la detección automática de anomalías en registros procedentes de fuentes heterogéneas mediante la aplicación de modelos no supervisados de aprendizaje automático. En la Figura 7.1 se identifica esta contribución dentro de la propuesta global de la Tesis Doctoral, que se introdujo en la Figura 6.1.

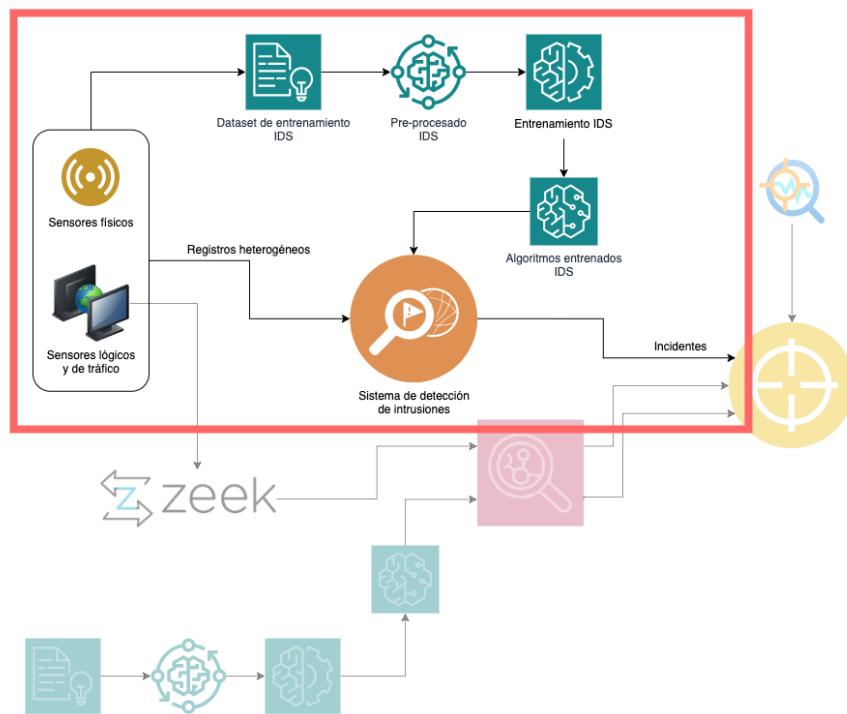


Figura 7.1: Módulo IDS en la arquitectura global propuesta

Este módulo permite el análisis de registros y la identificación de incidentes en entornos heterogéneos, que suponen riesgos para el sistema en su conjunto. Tras haber analizado el marco teórico relacionado con esta temática en la Sección 4.4, en este capítulo se hará un estudio de los trabajos previos (Sección 7.2) y se presentará el desarrollo llevado a cabo (Secciones 7.3-7.4), incluyendo los resultados obtenidos (Sección 7.5) y las conclusiones que se extraen del estudio (Sección 7.6).

7.1 Introducción

Uno de los objetivos principales de la investigación propuesta en esta Tesis Doctoral es la identificación de incidentes que se traducen en amenazas para el sistema. La clasificación de estos ataques permite además plantear distintos escenarios de riesgo, estableciendo procedimientos de reacción frente a ellos. Sin embargo, el análisis de grandes volúmenes de datos heterogéneos, como es este entorno, implica un coste computacional alto para llevar a cabo la detección de intrusiones en tiempo real. Esto ha llevado a derivar esta tarea en los modelos de aprendizaje automático como los presentados en la Sección 4.4, que permiten generar conocimiento de eventos no etiquetados, como es la procedente de sensores físicos y lógicos. Estos algoritmos deben identificar el tráfico normal y comparar la entrada para decidir si se parece y es un comportamiento habitual, o no y, por tanto, representa un incidente de ciberseguridad.

Dada la compleja naturaleza del entorno de ciberseguridad en el que se sitúa esta propuesta, con el tiempo se han desarrollado nuevos métodos, como la incorporación de IA, que permiten ayudar a comprender la evolución de las amenazas. Los incidentes de seguridad no contienen únicamente información de tráfico, sino datos temporales y geográficos o de comportamiento [107]. Analizar estos datos e identificar las intrusiones permite desarrollar medidas preventivas para las características únicas de la red, facilitando la detección de actividad inusual.

Para cumplir con el objetivo propuesto en la investigación y poder identificar ciberataques en datos heterogéneos, se diseña un entorno real basado en un sistema escalable de código abierto que permita gestionar grandes cantidades de datos y con capacidad de detectar anomalías en un conjunto de datos procedente de sensores y dispositivos de seguridad en tiempo real.

Los modelos se entrenan con el comportamiento normal de cada fuente aplicando un algoritmo de agrupamiento y definiendo un umbral a partir del cual se considera anómalo. Mediante el uso de métricas como el WSSSE y Silhouette se pueden optimizar los hiperparámetros y evaluar el comportamiento del modelo. Estos datos se almacenan en un módulo para analizar los resultados y utilizarlos en otros subsistemas que, en conjunto, forman un entorno de conciencia cibersituacional.

7.2 Trabajos relacionados

Para resolver el problema de la detección en tiempo real de anomalías, existen multitud de estudios previos en los que se presenta un mecanismo capaz de realizar la tarea de manera simple y efectiva. El enfoque más repetido son las tecnologías de Big Data, porque permiten gestionar grandes cantidades de datos que se generan en tiempo real, especialmente si es

información no etiquetada. Entrenar modelos que detecten intrusiones requieren de un *dataset* adecuado y correctamente etiquetado y, en caso de no estar disponible, elegir modelos no supervisados [75].

Por ese motivo existe literatura donde se aplican los algoritmos de aprendizaje no supervisado. Los autores de [148] utilizan un método de agrupamiento K-Means para etiquetar un conjunto de datos procedentes de redes de *Internet of Things* (IoT) y poder usarlo como punto de partida para entrenar modelos supervisados y detectar tarjetas SIM anómalas. Similar a esta propuesta, en [99] los autores aplican el algoritmo de reducción de dimensión PCA sobre un conjunto de datos para después entrenar un Mini Batch K-Means. Esto permite mejorar el tiempo de ejecución y las métricas que evalúan los *clusters* formados.

Como la propuesta que se plantea en este capítulo, se describe en [49] un método en el que los autores aplican un umbral sobre cada *cluster* formado para eliminar ruido o anomalías dentro de un marco de trabajo en el que se analizan los resultados de aplicar el algoritmo K-Means sobre grandes conjuntos de datos de IoT, pudiendo así eliminar los valores atípicos.

Por otro lado, los autores de [100] proponen un método que combina técnicas de agrupamiento y *Support Vector Machine* (SVM) para la detección de anomalías en el *dataset* NSL-KDD. Los investigadores en [87] describen un sistema capaz de detectar anomalías en las transmisiones de sensores IoT aplicando modelos estadísticos y de aprendizaje profundo.

En [14], los autores presentan un sistema de detección de anomalías no supervisado basado en *autoencoders* y redes generativas adversativas (*Generative Adversarial Networks* o GAN) que se prueba sobre cinco *datasets* públicos (SWaT, WADI, SMD, SMAP y MSL) y uno interno. Los autores en [63] proponen una implementación en tiempo real del *Isolated Forest* para la detección de anomalías en un conjunto de datos de simulación de aero-propulsión. En [103], los autores aprovechan las ventajas de un *Adversarial Auto Encoder* o AAE en la detección de anomalías dentro de tres conjuntos de datos del espectro Wi-Fi y un conjunto sintético. Además, en [81] los autores describen un enfoque para la detección de anomalías Bluetooth basándose en datos de seguimiento y el algoritmo de *Isolation Forest*.

En la Tabla 7.1 se resumen los procedimientos de estas investigaciones previas.

Un aspecto importante a destacar en varios de los estudios es la métrica mediante la que se evalúan los resultados. Al no tener etiquetas, las métricas son una evaluación indirecta del rendimiento del modelo. De esta forma, estudios como [92] recopilan las distintas métricas aplicadas en un algoritmo de agrupamiento.

Tabla 7.1: Resultados de los trabajos previos

Investigación	<i>Dataset</i>	Pre-procesado	Modelos	Tiempo Real
[148]	Propio	Sí	K-Means	Sí
[99]	KDDCUP99	No	Mini-Batch K-Means y PCA	No
[49]	Repositorio UCI Machine Learning	No	K-Means y aproximación basada en distancia	No
[100]	NSL-KDD	Sí	SCC-OCSVM	No
[87]	Yahoo Webscope	No	Modelos de aprendizaje profundo y estadístico	No
[14]	SWaT, WADI, SMD, SMAP, MSL y interno	Sí	<i>Autoencoders</i>	No
[63]	Simulación sistema aero-propulsión	Sí	<i>Isolated Forest</i>	Sí
[103]	Propio	No	AAE	No
[81]	Propio	Sí	<i>Isolated Forest</i>	Sí

Por lo general, los modelos de aprendizaje no supervisado se diseñan considerando un tipo específico de datos que se encuentran en el entorno estudiado. De este modo, las propuestas analizadas no consideran las particularidades de los entornos heterogéneos y ninguna trata de manera correcta la necesidad identificada en esta Tesis Doctoral. Por lo tanto, en este capítulo se propone un método no supervisado basado en el entrenamiento a partir de datos que corresponden al comportamiento normal de una red en la que existen dispositivos físicos y sensores lógicos que generan datos heterogéneos. A partir de ahí, se realiza la detección de anomalías en tiempo real en base a la clasificación errónea de los eventos en los grupos formados. Se evaluará el resultado mediante la visualización de los *clusters* y recopilando métricas que proporcionen información sobre el rendimiento de las agrupaciones.

7.3 Propuesta

En la sección anterior se identificaron trabajos que aplicaban el aprendizaje no supervisado a la detección de anomalías en escenarios concretos. Sin embargo, actualmente la interconexión de dispositivos está a la orden del día, generando en el entorno datos heterogéneos que deben ser analizados y tratados correctamente para evaluar amenazas procedentes de fuentes al margen de redes como Internet. En este capítulo se presenta una propuesta para la identificación de ciberataques en estos casos: en lugar de centrarse únicamente en el tráfico de red o de un dispositivo concreto, se plantea estudiar todas las comunicaciones y tecnologías que existen en el escenario.

Para este análisis se parte de fuentes de datos heterogéneas que se producen en tiempo real. Esta información puede representar tráfico normal o ataques, y por tanto debe analizarse y clasificarse al ser recibida. De esta forma, es necesario un módulo que identifique el incidente y permita al resto del sistema reaccionar frente a él.

Este entorno se caracteriza por el desequilibrio entre el volumen de tráfico normal y el que representa estos ataques, por lo que entrenar modelos con datos reales es una tarea complicada. Este es el motivo por el que, en la aproximación a la problemática, se optó por entrenar modelos no supervisados para detectar este comportamiento normal y lo que se aleje más de un determinado umbral de este comportamiento, se considerará tráfico anómalo.

En la Figura 7.2 se presenta con mayor detalle la arquitectura del módulo resaltado en la Figura 7.1 de la propuesta global de esta Tesis Doctoral. Se muestran los distintos sub-sistemas que conforman el IDS diseñado, desde la recepción de los datos originales hasta la clasificación de los eventos por los distintos modelos, como se detalla en las próximas secciones.

Este planteamiento parte de la existencia de sensores en el entorno que generan datos heterogéneos, donde se ha identificado la problemática principal en la literatura existente. No obstante, para el desarrollo se hace referencia a los sensores utilizados en el proyecto PLICA, mediante el que se ha validado. Éstos analizan las comunicaciones mediante Wi-Fi, *Bluetooth*, redes móviles, y radiofrecuencia. Además, sensores que recogen el comportamiento de usuario (*User and Entity Behavior Analytics*, UEBA) y datos de tráfico como SIEM y Firewall.

En el subsistema que entrena los modelos no supervisados elegidos se utilizan datos reales y generados sintéticamente en caso de que no existan suficientes datos reales para el entrenamiento y la validación. Este es el caso de la información de redes móviles, ya que el sensor no captura la cantidad de muestras necesaria para poder entrenar correctamente un modelo, ya que el resto de sensores sí captan suficientes datos. En total se generan un par de *datasets* normales/anómalos por cada dispositivo. Los modelos entrenados se pasan al sistema de tiempo real, que permite el procesado de los datos procedentes de los sensores en el momento en el que se reciben.

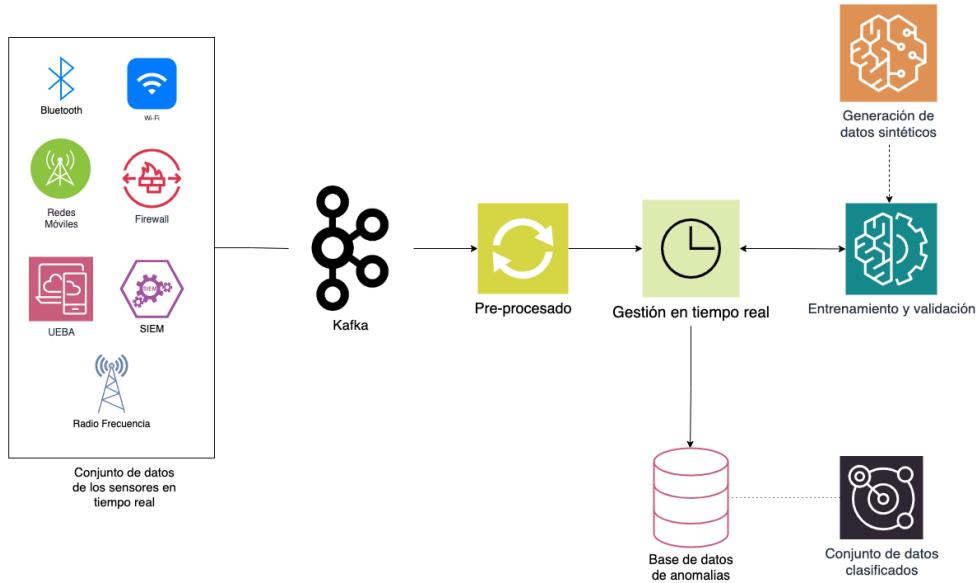


Figura 7.2: Arquitectura de la propuesta

Estos dispositivos físicos y lógicos envían la información mediante un módulo Kafka [1] y los datos se transforman (pre-procesado) mediante funciones matemáticas para introducirse en los algoritmos que se encuentran en el módulo de tiempo real y el modelo de entrenamiento. Aquí se eligen las propiedades de los datos que permiten optimizar el entrenamiento.

Cuando se reciben datos procedentes de los sensores en tiempo real, se transforman y, una vez clasificados en tráfico normal (0) o anomalía (1), se almacenan en una base de datos para que puedan utilizarse en otros sistemas del entorno.

7.4 Diseño de la propuesta

La propuesta, como se ha expuesto hasta este momento, se centra en la aplicación de algoritmos de agrupamiento no supervisados para detectar anomalías en tiempo real entre datos heterogéneos. El motivo principal de elegir los algoritmos de *clustering*, como se presentó en el Capítulo 4, Sección 4.4, y como se ha analizado en los trabajos previos presentados en la Sección 7.2 de este mismo capítulo, tienen muy buenos resultados en este tipo de entornos.

Por las características de este tipo de entornos, el desarrollo de la arquitectura presentada debe adecuarse a los sensores que actúen como fuente de datos. Sin embargo, la propuesta definida es independiente del número y tipo de dispositivos, como se refleja en la Figura 7.3: se entrena un conjunto de modelos de agrupamiento para identificar el comportamiento normal en datos heterogéneos y poder detectar anomalías cuando se reciban en tiempo real. Los registros o eventos recibidos deben pre-procesarse para traducir los campos que capture cada fuente en información numérica que pueda analizar un algoritmo, por lo que estos procesos dependen del formato de los datos. En azul se muestra el conjunto de pasos para el entrenamiento del sistema, mientras que en rosa aparece el proceso de detección de ciberataques en tiempo real.

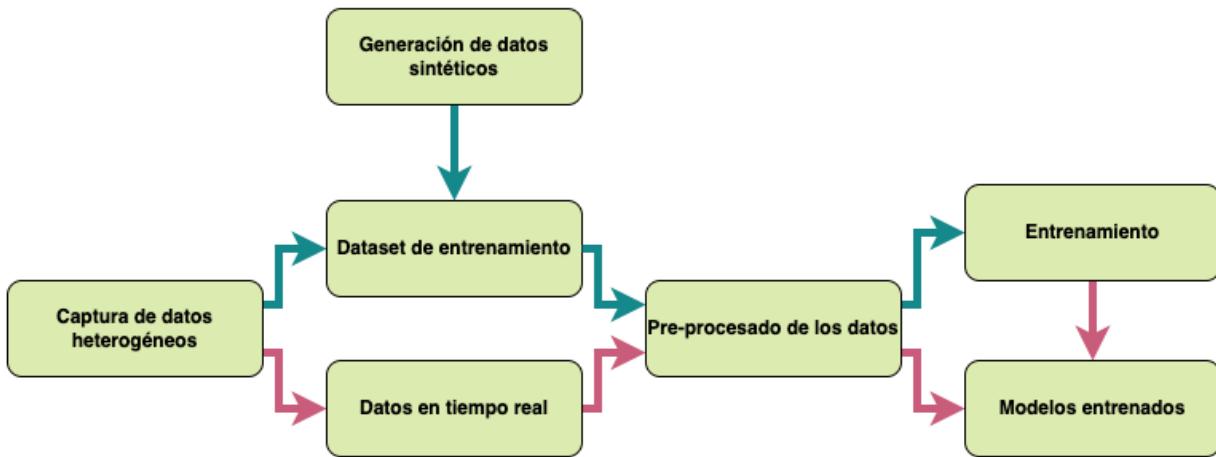


Figura 7.3: Proceso para el diseño de un IDS utilizando fuentes de datos heterogéneos

Como se mencionó anteriormente, para ilustrar el diseño llevado a cabo se utilizan como referencia los sensores utilizados en el proyecto PLICA. A continuación se van a definir con mayor detalle los módulos presentados en la Figura 7.2, adaptándose a los dispositivos mencionados. La contribución parcial presentada en este capítulo ha sido validada en [111], y en el Capítulo 10 se evaluará el resultado del caso concreto para el conjunto de sensores elegido como ejemplo.

7.4.1 Conjunto de datos de entrenamiento

La entrada del sistema corresponde con los datos que generan los distintos dispositivos fuente (los sensores Wi-Fi, *Bluetooth*, redes móviles, radiofrecuencia, registros del SIEM y del cortafuegos, y los dispositivos UEBA).

De los datos que capta cada dispositivo se pueden extraer un conjunto de parámetros y características. Aunque no todas se utilizarán en el entrenamiento de los modelos, ya que es necesario optimizar los procesos, los campos de cada uno se detallan en las Tablas 7.2 - 7.13.

Tabla 7.2: Campos de los datos de los dispositivos redes móviles

Campos	Descripción
<i>Time</i>	Tiempo de descubrimiento del dispositivo.
IMEI	ID del dispositivo detectado.
IMSI	ID internacional del abonado.
RAT	Tipo de acceso radio (2G, 3G, 4G).

Tabla 7.3: Campos de los datos de los dispositivos radiofrecuencia

Campos	Descripción
<i>Time</i>	Fecha de la medida. Formato EPOCH, resolución en segundos.
<i>Signal</i>	Nivel de potencia de la señal (dBms).
<i>Freq</i>	Frecuencia de la señal (MHz).
<i>mod</i>	Tipo de modulación de la señal (OOK, 2FSK, NONE).
<i>payload</i>	Datos asociados o extraídos de la señal.

Tabla 7.4: Campos de los datos de los dispositivos *Bluetooth*

Campos	Descripción
<i>Time</i>	Tiempo de creación.
<i>status</i>	Estado en el que se encuentra el dispositivo.
<i>classic_mode</i>	Define si el <i>Bluetooth</i> se encuentra en modo clásico.
<i>le_mode</i>	<i>Bluetooth</i> de baja energía.
<i>lmp_version</i>	Versión <i>Bluetooth</i> utilizada por el dispositivo detectado.
<i>address</i>	Dirección física del dispositivo detectado.

Tabla 7.5: Campos de los datos de los dispositivos Wi-Fi

Campos	Descripción
<i>Time</i>	Fecha de la medida en formato EPOCH, resolución en segundos.
<i>userid</i>	Identificador. Dirección MAC.
<i>footprint</i>	Identificador en formato MAC para direcciones aleatorias.
<i>tseen</i>	Tiempo (s) en el que el usuario ha sido detectado en el intervalo de medición.
<i>tacum</i>	Tiempo (s) en el que el usuario ha sido detectado desde que el sensor está operativo.
<i>visits</i>	Número de veces que el usuario ha sido detectado después de haber sido desconectado (entradas y re-entradas).
<i>pwr</i>	Potencia media del dispositivo durante el intervalo de medida.
<i>tx_packets / tx_bytes</i>	Número de paquetes/bytes transmitidos en el intervalo de medida.
<i>rx_packets / rx_bytes</i>	Número de paquetes/bytes recibidos en el intervalo de medida.
<i>apwr</i>	Potencia media del punto de acceso visto por el sensor.
<i>type</i>	Tipo de dirección MAC (MAL, LMA, CID, y <i>Unknown</i>).

Tabla 7.6: Campos de los datos de los dispositivos SIEM

Campos	Descripción
<i>Date</i>	Fecha de la medida. Formato EPOCH, resolución en segundos.
<i>Sensor</i>	Tipo de sensor SIEM.
<i>Risk</i>	Nivel de riesgo.
<i>Signature</i>	Descripción del evento creado por el SIEM.
<i>Source</i>	Dirección IP:<puerto>origen.
<i>Destination</i>	Dirección IP:<puerto>destino.

Tabla 7.7: Campos de los datos de los cortafuegos

Campos	Descripción
<i>Time</i>	Fecha de la medición.
<i>sequencenum</i>	Permite ordenar registros con el mismo sello temporal y origen.
<i>source port</i>	Puerto origen.
<i>destination port</i>	Puerto destino.
<i>Xlate (NAT) source port</i>	Puerto origen tras aplicar <i>Hide NAT</i> en la IP origen.
<i>Xlate (NAT) destination port</i>	Puerto destino tras aplicar NAT.
<i>VPN Peer Gateway</i>	Dirección IP principal de la puerta de enlace de seguridad del par VPN.
<i>Blade</i>	Nombre de producto.
<i>Action</i>	Acción de la regla coincidente en la política de acceso.
<i>type</i>	Tipo de registro.
<i>interface direction</i>	Dirección de la conexión.
<i>source zone</i>	Interna o Externa.
<i>destination zone</i>	Interna o Externa.
<i>IP protocol</i>	Protocolo IP utilizado.
<i>needs browse time</i>	Tiempo de navegación necesario para la conexión.
<i>protocol</i>	Protocolo detectado en la conexión.
<i>ICMP</i>	Mensaje ICMP añadido al registro de conexión.
<i>ICMP Type</i>	Para la conexión ICMP, se añade al registro información de tipo.
<i>ICMP Code</i>	Para la conexión ICMP, se añade al registro información de código.
<i>PPP</i>	Estado de la autenticación.
<i>Authentication method</i>	Protocolo de autenticación de contraseñas utilizado (PAP o EAP).
<i>scheme</i>	Esquema usado para el registro.
<i>methods</i>	Método HTTP.
<i>VPN Feature</i>	L2TP / IKE / Link Selection.

Tabla 7.8: Campos de los datos de los dispositivos UEBA - Monitor de actividad

Campos	Descripción
<i>Time</i>	Fecha y hora de la medida. Formato EPOCH, resolución en segundos.
<i>clicks</i>	Número de <i>clicks</i> realizados en el ratón.
<i>pulsations</i>	Número de pulsaciones que el usuario ha dado sobre el teclado.
<i>moves</i>	Número de desplazamientos del ratón por la pantalla.
<i>scrolls</i>	Número de veces que la funcionalidad <i>scroll</i> se ha usado.

Tabla 7.9: Campos de los datos de los dispositivos UEBA - Buscador

Campos	Descripción
<i>Date</i>	Fecha y hora de la medida. Formato EPOCH, resolución en segundos.
<i>URL</i>	Dirección web visitada.

Tabla 7.10: Campos de los datos de los dispositivos UEBA - Procesos

Campos	Descripción
<i>Pid</i>	PID que identifica al proceso.
<i>Name</i>	Nombre del fichero que ha ejecutado el proceso.
<i>Create time</i>	Fecha del momento en el que se inició el proceso.
<i>Cores</i>	Número de cores que el proceso tiene permiso para usar.
<i>Cpu usage</i>	Máximo porcentaje de CPU que ha llegado a usar el proceso durante su vida.
<i>Nice</i>	Prioridad del proceso.
<i>Memory usage</i>	Uso máximo de la memoria que realizó el proceso durante su vida mostrada en bytes.
<i>N threads</i>	Número de threads máximos creados por el proceso durante su ejecución.
<i>Childrens</i>	Número de sub-procesos que ha generado el proceso durante su vida.
<i>Username</i>	Nombre del usuario que inició el proceso.
<i>Finish Time</i>	Fecha en el que el proceso finalizó.

Tabla 7.11: Campos de los datos de los dispositivos UEBA - *Sockets*

Dispositivo	Campos
Fd	Descriptor del archivo <i>socket</i> .
Type	Tipo de conexión a la que está ligada el <i>socket</i> .
Laddr	Dirección IP local.
Raddr	Dirección IP remota.
Pid	PID del proceso ligado al <i>socket</i> .
<i>Detection time</i>	Fecha y hora en la que el <i>socket</i> fue detectado. Formato EPOCH, resolución en segundos.
<i>Closed time</i>	Fecha y hora en la que se cierra el <i>socket</i> . Formato EPOCH, resolución en segundos.
Laddrport	Número de puerto local.
Raddrport	Número de puerto remoto.

Tabla 7.12: Campos de los datos de los dispositivos UEBA - Documentos

Campos	Descripción
<i>Date</i>	Fecha y hora de la medida. Formato EPOCH, resolución en segundos.
<i>path</i>	Dirección completa del fichero sobre el que se ha realizado la acción.
<i>type</i>	Tipo de acción realizada sobre el archivo.

Tabla 7.13: Campos de los datos de los dispositivos UEBA - Red

Campos	Descripción
<i>Date</i>	Fecha y hora de la medida. Formato EPOCH, resolución en segundos.
<i>Name</i>	Nombre de la interfaz supervisada.
<i>Total bytes sent</i>	Nº total de bytes que se han enviado en esa interfaz.
<i>Total bytes recv</i>	Nº total de bytes que se han recibido en esa interfaz.
<i>Total packets sent</i>	Nº total de paquetes que se han enviado en esa interfaz
<i>Total packets recv</i>	Nº total de paquetes que se han recibido en esa interfaz.
<i>Total errin</i>	Nº total de errores producidos mientras esa interfaz recibe paquetes.
<i>Total errout</i>	Nº total de errores producidos mientras la interfaz envía paquetes.
<i>Total dropout</i>	Nº total de paquetes recibidos que han sido descartados.
<i>Bytes sent</i>	Nº de bytes que se han enviado en esa interfaz desde el anterior evento.
<i>Bytes recv</i>	Nº de bytes que se han recibido en esa interfaz desde el anterior evento.
<i>Packets sent</i>	Nº de paquetes que se han enviado en esa interfaz desde el último evento.
<i>Packets recv</i>	Nº de paquetes que se han recibido en esa interfaz desde el último evento.
<i>Errin</i>	Nº de errores producidos mientras recibe paquetes desde el último evento.
<i>Errout</i>	Nº de errores producidos mientras envia paquetes en esa interfaz.
<i>Dropout</i>	Nº paquetes recibidos que han sido descartados desde el último evento.

Cada uno de estos dispositivos envía los datos que genera a través del subsistema de gestión de flujos Kafka, utilizando *topics* diferenciados, uno por dispositivo. Cada modelo de aprendizaje se suscribe al *topic* correspondiente y, por tanto, se desarrolla, entrena y valida un modelo de aprendizaje no supervisado por fuente de datos (tras seleccionar uno de los posibles modelos). Algunos de estos dispositivos ya han sido probados en otros proyectos [10, 138].

7.4.2 Generación de datos sintéticos

Este módulo tiene el objetivo de crear conjuntos de datos sintéticos para los modelos donde los datos de entrada no cumplen las condiciones mínimas, por ejemplo, que el dispositivo no recoja información suficiente por pertenecer a un entorno controlado. En este caso, se utiliza para generar información similar a la que envía el sensor de redes móviles.

Los módulos de este sub-sistema se relacionan según aparece en la Figura 7.4.

El sistema de generación de datos sintéticos utiliza datos procedentes de las fuentes y establece las relaciones necesarias entre los atributos para evitar entradas erróneas. Define perfiles para generar un número determinado de eventos por paso de reloj.

La configuración del reloj se define previamente y especifica el tiempo de simulación y el tiempo entre pasos. Estará condicionada por un perfil temporal donde las características de

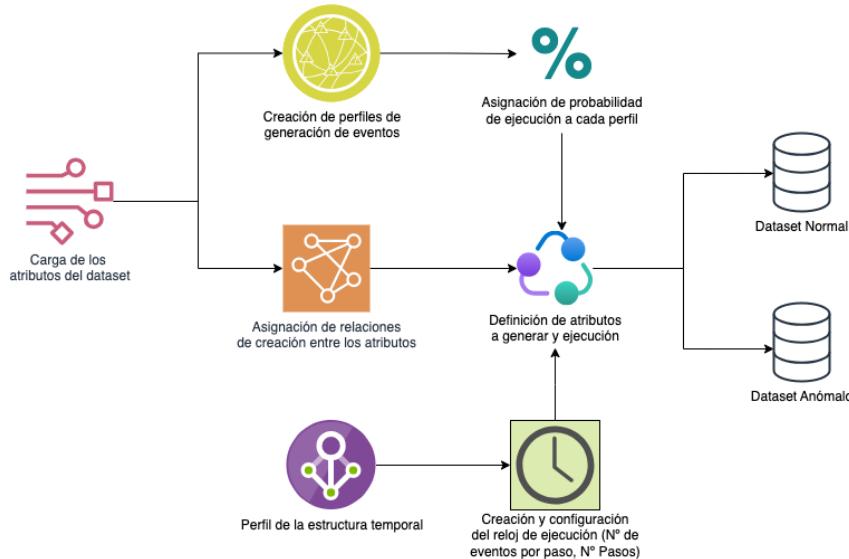


Figura 7.4: Módulo de generación de datos

la generación de eventos se especifiquen en función del tiempo.

Finalmente, los atributos generados se definen con las configuraciones, relaciones y perfiles. Cabe destacar la existencia de dos configuraciones distintas, para el tráfico normal y para el tráfico anómalo. Las características de los *datasets* generados son idénticas a los que provienen de los dispositivos que se están sobre-muestreando.

Estas configuraciones se refieren a la librería para generación de datos Trumania [4], que contiene las herramientas necesarias para generar los datos, definir las estructuras internas y permitir la generación de eventos que se ajustan a necesidades como los tipos adecuados para todos los valores del *dataset*, la creación de eventos no uniforme o la estructura temporal similar a un escenario real.

Los datos de salida corresponden a un par de registros generados con características normales y otro con características anómalas. El tamaño de estos registros es de aproximadamente 100.000 entradas que pueden variar en el orden.

7.4.3 Pre-procesado

Este sub-sistema se encarga de normalizar, transformar y estandarizar el conjunto de datos antes de entrenar los algoritmos de aprendizaje automático y el procesamiento en tiempo real. Los datos de entrada serán la información procedente de los sensores. Los *dataset* primero se estructurarán según el tipo de dato, definiendo un esquema con los atributos de cada dispositivo y el tipo de valor que contienen [69].

A continuación, se aplican los módulos Spark [2] para el pre-procesado y las funciones definidas para cada tipo de eventos, llevando a cabo transformaciones y ajustando los datos como resultados. Las funciones más utilizadas tienen los siguientes objetivos:

- *MinMaxScaler*: normalizar los datos numéricos a un rango por defecto.

- *String Indexer*: codificar las columnas de texto para convertirlas en índices.
- *One Hot Encoder*: crea distintas columnas con el conjunto de posibles valores de esa característica concreta, y asigna valor positivo al valor presente en cada fila, y cero al resto.
- *Regex Tokenizer, Count Vectorizer, Term Frequency-Inverse Document Frequency (TF-IDF)*: separar las cadenas de texto en tokens, crear una matriz con los términos más frecuentes y ajustar los pesos de los términos representando la importancia de cada uno en el contexto.
- *Word2Vec*: convertir las palabras en vectores numéricos.
- PCA: reducir la dimensión y complejidad de los datos.
- *Vector Assembler*: unir todos los datos en un vector.

Las distintas estrategias de transformación se representan de manera esquemática en las Figuras 7.5 - 7.16.

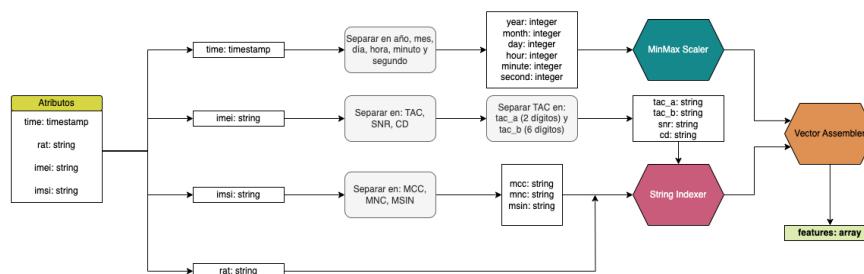


Figura 7.5: Pre-procesado de datos de sensor de redes móviles

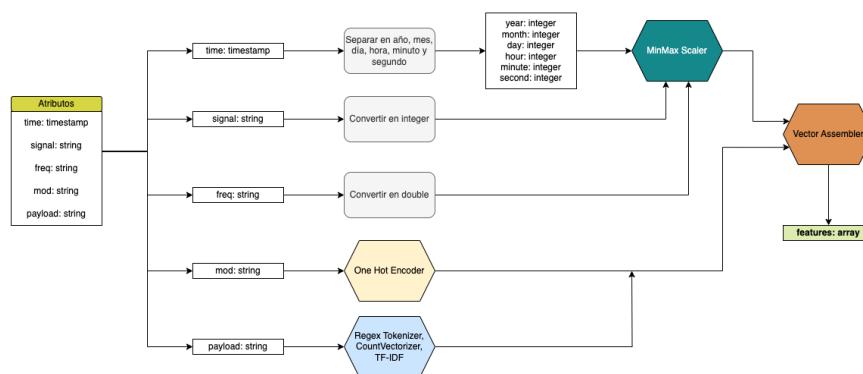


Figura 7.6: Pre-procesado de datos de sensor de radiofrecuencia

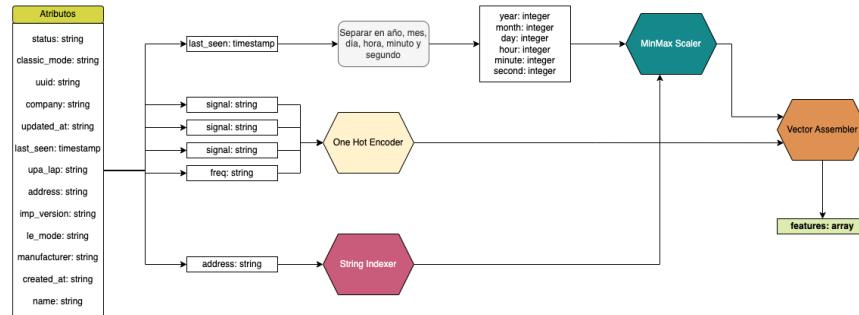


Figura 7.7: Pre-procesado de datos de sensor *Bluetooth*

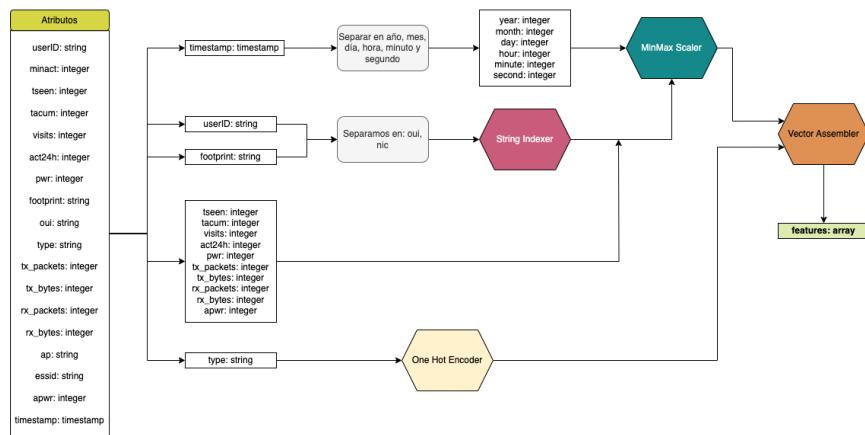


Figura 7.8: Pre-procesado de datos de sensor *Wi-Fi*

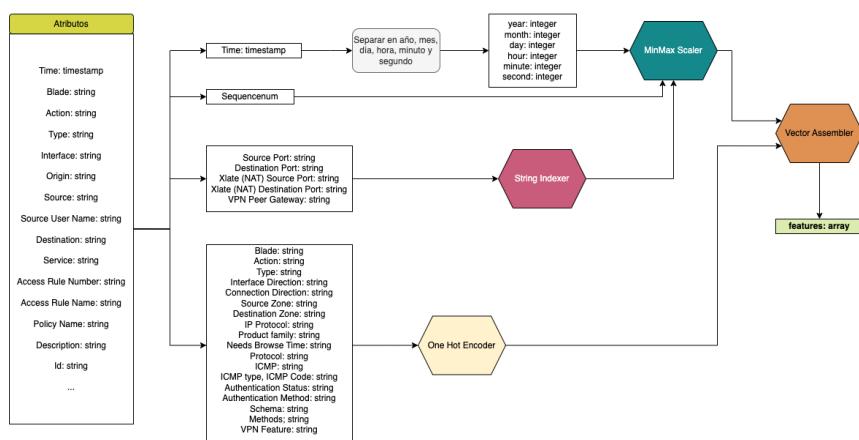


Figura 7.9: Pre-procesado de datos de cortafuegos

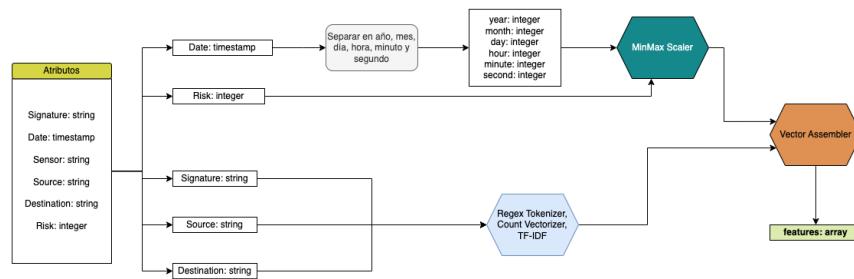


Figura 7.10: Pre-procesado de datos de SIEM

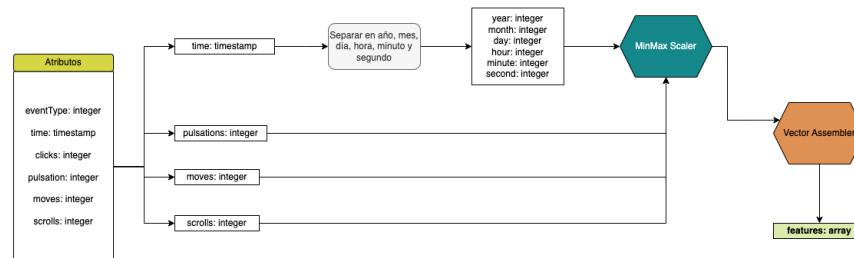


Figura 7.11: Pre-procesado de datos de actividad (UEBA)

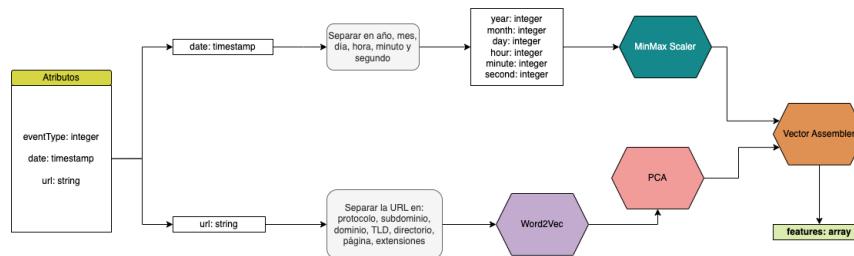


Figura 7.12: Pre-procesado de datos de buscador (UEBA)

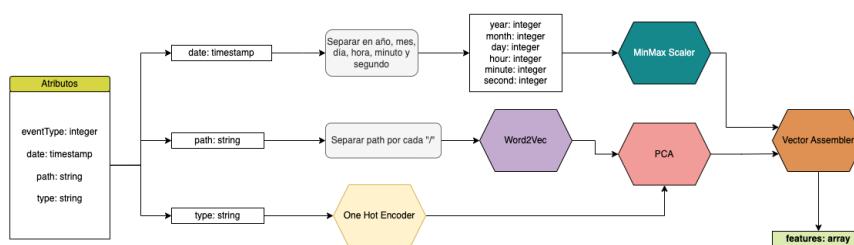


Figura 7.13: Pre-procesado de datos de documentos (UEBA)

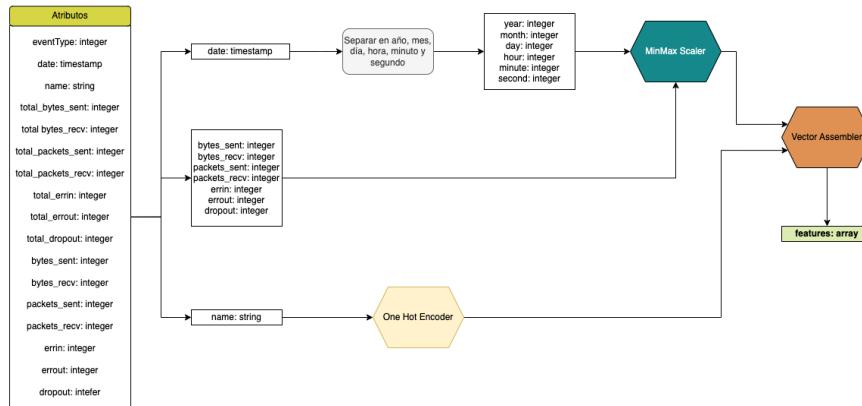


Figura 7.14: Pre-procesado de datos de red (UEBA)

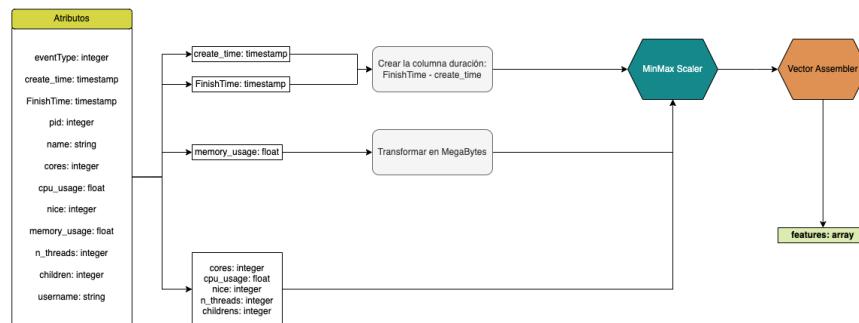
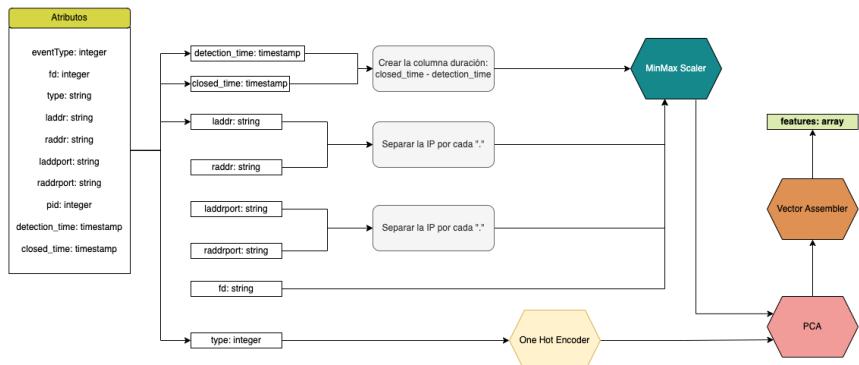


Figura 7.15: Pre-procesado de datos de procesos (UEBA)


 Figura 7.16: Pre-procesado de datos de *sockets* (UEBA)

7.4.4 Entrenamiento y validación

En este sub-sistema se generan los modelos de aprendizaje automático que permiten la detección de posibles anomalías basadas en los datos de los dispositivos descritos anteriormente. Está compuesto por los siguientes componentes: datos de entrenamiento y validación, módulo de selección de hiperparámetros, módulo de selección del algoritmo, módulo de métricas y el sistema final. La arquitectura del sub-sistema se muestra en la Figura 7.17.

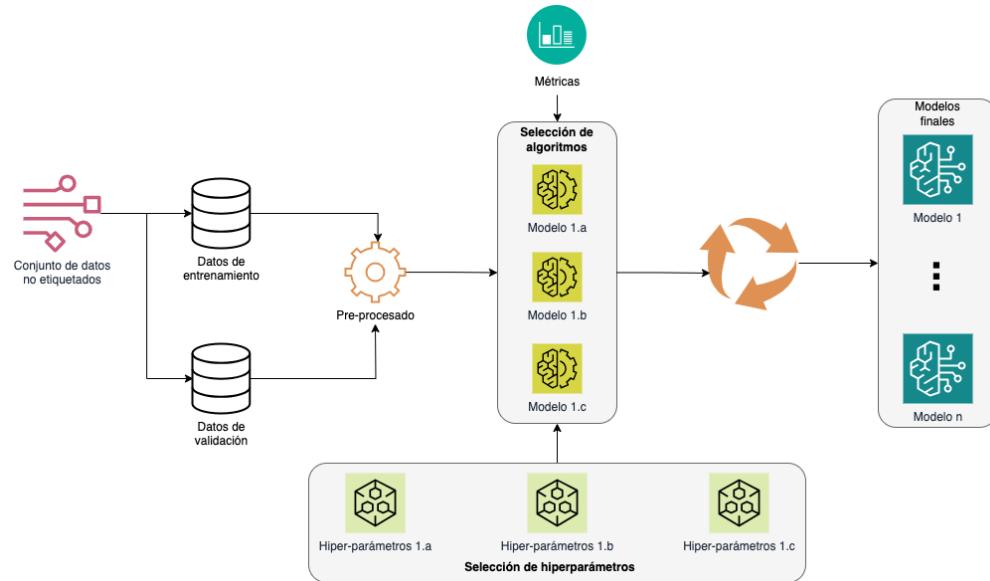


Figura 7.17: Módulo de entrenamiento y validación

Estos módulos serán los encargados de preparar y monitorizar el correcto funcionamiento de los modelos, validando que los datos de entrada sean consistentes. Se entrenarán tres tipos de modelos para cada tipo de datos, K-Means, Bisecting K-Means y GMM, eligiendo el que dé mejores resultados.

Los datos pre-procesados entrenarán los modelos, y se identificará el mejor para cada tipo de dispositivo utilizando los módulos de selección de hiperparámetros, las métricas y la validación. El primero determinará qué conjunto de parámetros para un modelo dado se adapta mejor a la detección de anomalías. Los parámetros se determinan según los resultados generados por el módulo de métricas y el *dataset* de validación. Las funciones matemáticas seleccionadas para elegir los hiperparámetros son WSSSE y Silhouette con el criterio del punto de codo en el gráfico (punto donde la gráfica cambia de pendiente). Una vez se ha obtenido el modelo más preciso, se elige el óptimo para cada dispositivo, obteniendo un modelo final entrenado para cada fuente.

Algoritmo de aprendizaje automático. Umbral

Los algoritmos utilizados para identificar anomalías, K-Means, Bisecting K-Means y GMM, agrupan los datos de manera no supervisada en *clusters* de características similares.

La elección entre los algoritmos se realiza tras la comparación de las métricas WSSSE y Silhouette, como se presentará en la Sección 7.5, seleccionando el modelo con mejor rendimiento. Por ello, es importante configurar los modelos para que distingan correctamente el tráfico normal de los eventos anómalos, utilizando un umbral en los *clusters* formados, que marca qué datos son suficientemente distintos del resto del cluster para considerarse anomalía.

Los modelos se entrenarán con el comportamiento normal de los sensores, y cuando se reciban nuevos eventos, se compararán con ellos para decidir si son suficientemente parecidos para

etiquetarlos como tráfico normal o no. Esta implementación permite tener en cuenta cambios temporales en los datos, ya que se incluye el sello temporal como una de las características de entrada. Aunque el umbral es variable y puede ser restrictivo en mayor o menor medida, según sea necesario, por defecto para este escenario se define como límite el punto más lejano del centroide del *cluster*.

Selección de hiperparámetros

Este paso es vital en el diseño de los modelos de aprendizaje automático. Es un proceso iterativo en el que se evalúa un rango de valores para cada hiperparámetro del algoritmo y seleccionando el que obtenga mejor rendimiento según las métricas WSSSE y Silhouette, en este caso, con el menor número de *clusters* y tiempo de entrenamiento.

En caso de igualdad, como se indicó en el Capítulo 4.4.3 se priorizará el resultado de la métrica Silhouette, ya que analiza tanto la compacidad del *cluster* y la separación con los demás grupos, mientras que WSSSE solo analiza la dispersión entre los puntos del *cluster*. Si varias pruebas dan resultados similares en esta métrica, se tendrá en cuenta WSSSE y, después, el tiempo de entrenamiento.

Para cada uno de los algoritmos elegidos se seleccionan los siguientes hiper-parámetros (Tabla 7.14):

Tabla 7.14: Hiperparámetros de los modelos no supervisados

Modelo	Hiper-parámetro	Definición
K-Means	Número de <i>clusters</i> (k)	Número de grupos que se van a formar
	Medida de distancia	Métrica para calcular la distancia entre puntos
	Máximas iteraciones	Número máximo de iteraciones que se realizan
	Tolerancia	Valor de convergencia
<i>Bisection</i> K-Means	Número de <i>clusters</i> (k)	Número de grupos que se van a formar
	Medida de distancia	Métrica para calcular la distancia entre puntos
	Máximas iteraciones	Número máximo de iteraciones que se realizan
GMM	Número de <i>clusteres</i> (k)	Número de grupos que se forman
	Máximas iteraciones	Número máximo de iteraciones que se realizan
	Tolerancia	Valor de convergencia

Se entrena los tres tipos de modelos para todos los datos que se tratan, obteniendo los valores que se recogen en las Tablas 7.15 - 7.26. En las Figuras 7.18 - 7.39 se muestran las gráficas utilizadas para la selección de los hiper-parámetros del algoritmo K-Means, en aquellos parámetros que no mantienen el valor por defecto.

Tabla 7.15: Valores de los hiper-parámetros seleccionados para los datos de redes móviles

Modelo	Hiper-parámetro	Valor
K-Means	Número de <i>clusters</i> (k)	k = 2
	Medida de distancia	Euclídea
	Máximas iteraciones	Por defecto: 100
	Tolerancia	Por defecto: 10^{-4}
<i>Bisection K-Means</i>	Número de <i>clusters</i> (k)	No se puede optimizar
	Medida de distancia	No se puede optimizar
	Máximas iteraciones	No se puede optimizar
GMM	Número de <i>clusteres</i> (k)	k = 3
	Máximas iteraciones	Por defecto: 100
	Tolerancia	Por defecto: 10^{-4}

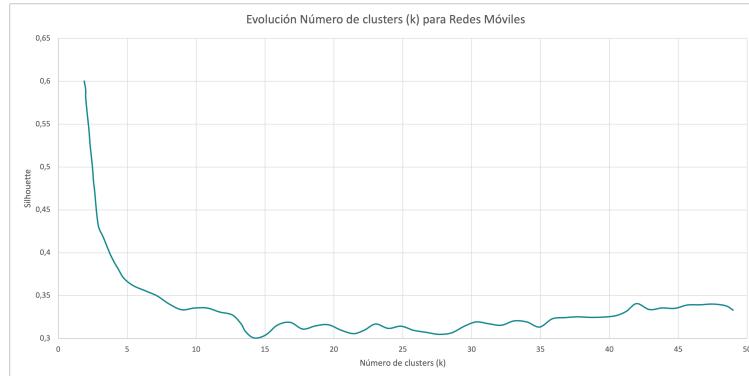
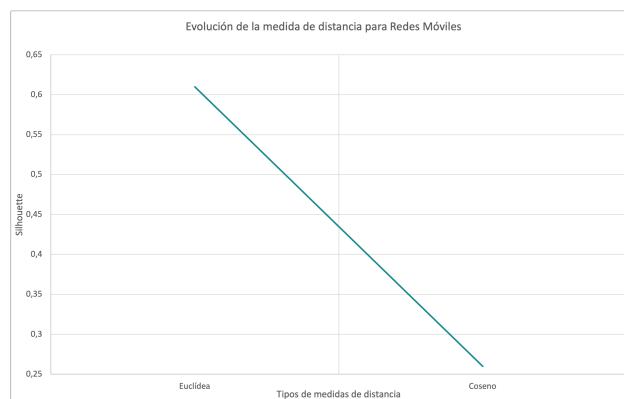
**Figura 7.18:** Métrica Silhouette - K-Means - Número de *clusters* - Redes móviles**Figura 7.19:** Métrica Silhouette - K-Means - Medida de la distancia - Redes móviles

Tabla 7.16: Valores de los hiper-parámetros seleccionados para los datos de radiofrecuencia

Modelo	Hiper-parámetro	Valor
K-Means	Número de <i>clusters</i> (k)	k = 2
	Medida de distancia	Coseno
	Máximas iteraciones	Por defecto: 100
	Tolerancia	Por defecto: 10^{-4}
<i>Bisecting K-Means</i>	Número de <i>clusters</i> (k)	k = 2
	Medida de distancia	Euclídea
	Máximas iteraciones	Por defecto: 100
GMM	Número de <i>clusteres</i> (k)	k = 3
	Máximas iteraciones	Por defecto: 100
	Tolerancia	Por defecto: 10^{-4}

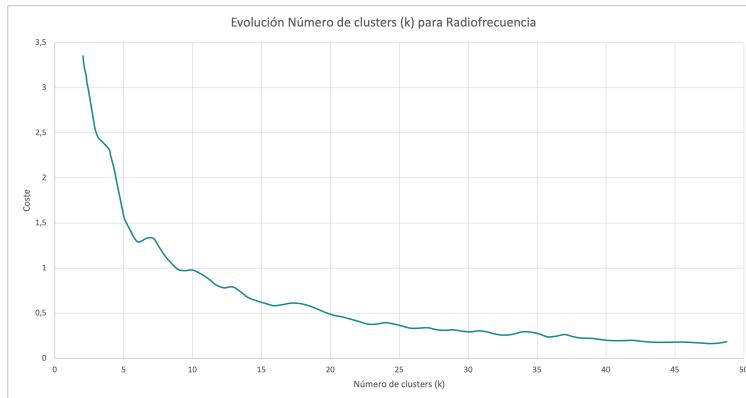


Figura 7.20: Coste - K-Means - Número de *clusters* - Radiofrecuencia

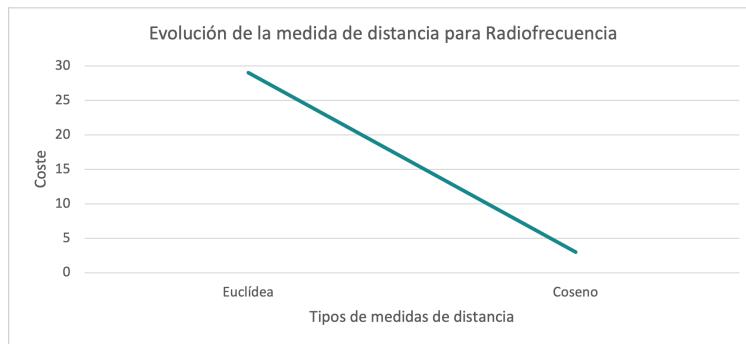


Figura 7.21: Coste - K-Means - Medida de la distancia - Radiofrecuencia

Tabla 7.17: Valores de los hiper-parámetros seleccionados para los datos *Bluetooth*

Modelo	Hiper-parámetro	Valor
K-Means	Número de <i>clusters</i> (k)	k = 2
	Medida de distancia	Coseno
	Máximas iteraciones	Por defecto: 100
	Tolerancia	Por defecto: 10^{-4}
<i>Bisecting K-Means</i>	Número de <i>clusters</i> (k)	k = 2
	Medida de distancia	Coseno
	Máximas iteraciones	Por defecto: 100
GMM	Número de <i>clusteres</i> (k)	k = 2
	Máximas iteraciones	Por defecto: 100
	Tolerancia	Por defecto: 10^{-4}

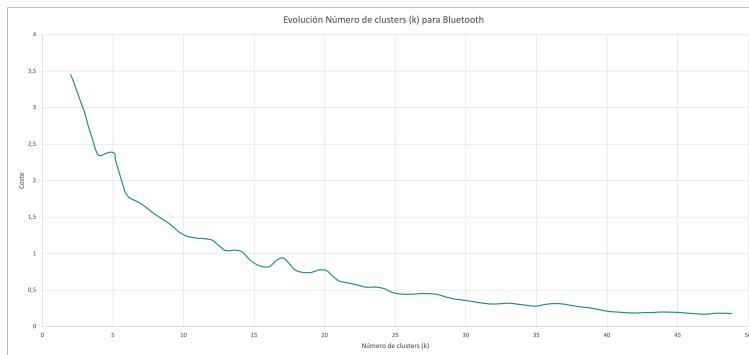
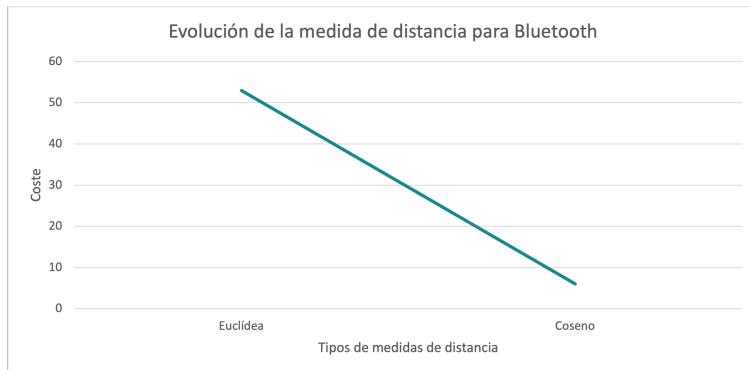
**Figura 7.22:** Coste - K-Means - Número de *clusters* - *Bluetooth***Figura 7.23:** Coste - K-Means - Medida de la distancia - *Bluetooth*

Tabla 7.18: Valores de los hiper-parámetros seleccionados para los datos Wi-Fi

Modelo	Hiper-parámetro	Valor
K-Means	Número de <i>clusters</i> (k)	k = 3
	Medida de distancia	Euclídea
	Máximas iteraciones	Por defecto: 100
	Tolerancia	Por defecto: 10^{-4}
<i>Bisection K-Means</i>	Número de <i>clusters</i> (k)	No se puede optimizar
	Medida de distancia	No se puede optimizar
	Máximas iteraciones	No se puede optimizar
GMM	Número de <i>clusteres</i> (k)	k = 2
	Máximas iteraciones	Por defecto: 100
	Tolerancia	Por defecto: 10^{-4}

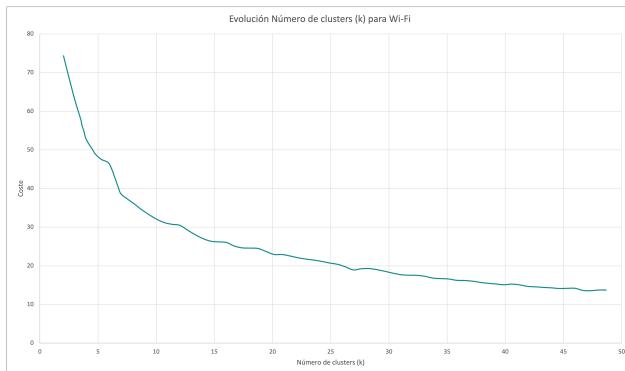


Figura 7.24: Coste - K-Means - Número de *clusters* - Wi-Fi

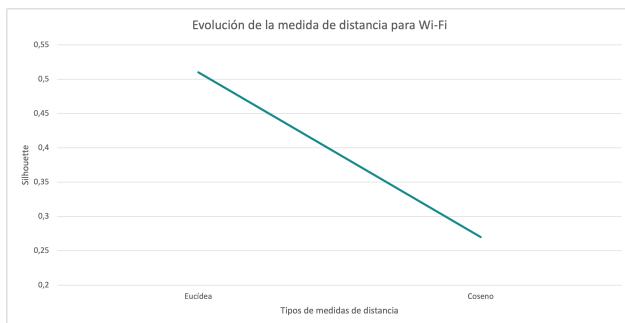


Figura 7.25: Métrica Silhouette - K-Means - Medida de la distancia - Wi-Fi

Tabla 7.19: Valores de los hiper-parámetros seleccionados para los datos del cortafuegos

Modelo	Hiper-parámetro	Valor
K-Means	Número de <i>clusters</i> (k)	k = 13
	Medida de distancia	Coseno
	Máximas iteraciones	Por defecto: 100
	Tolerancia	Por defecto: 10^{-4}
<i>Bisecting K-Means</i>	Número de <i>clusters</i> (k)	k = 9
	Medida de distancia	Coseno
	Máximas iteraciones	Por defecto: 100
GMM	Número de <i>clusteres</i> (k)	k = 4
	Máximas iteraciones	Por defecto: 100
	Tolerancia	Por defecto: 10^{-4}

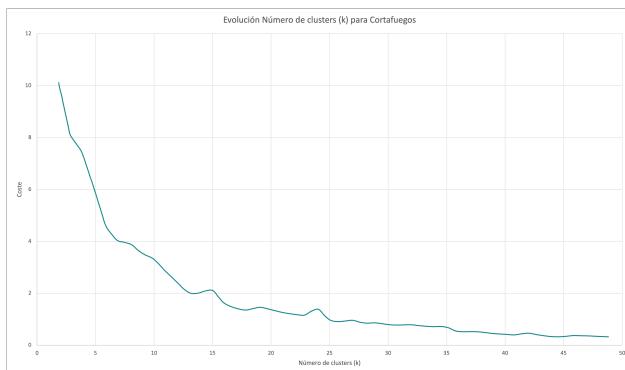


Figura 7.26: Coste - K-Means - Número de *clusters* - Cortafuegos

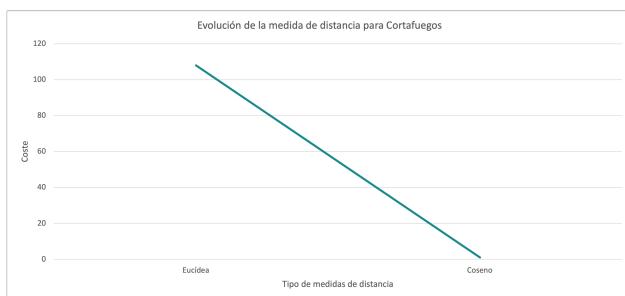


Figura 7.27: Coste - K-Means - Medida de la distancia - Cortafuegos

Tabla 7.20: Valores de los hiper-parámetros seleccionados para los datos del SIEM

Modelo	Hiper-parámetro	Valor
K-Means	Número de <i>clusters</i> (k)	k = 17
	Medida de distancia	Coseno
	Máximas iteraciones	Por defecto: 100
	Tolerancia	Por defecto: 10^{-4}
<i>Bisection K-Means</i>	Número de <i>clusters</i> (k)	No se puede optimizar
	Medida de distancia	No se puede optimizar
	Máximas iteraciones	No se puede optimizar
GMM	Número de <i>clusteres</i> (k)	No se puede optimizar
	Máximas iteraciones	No se puede optimizar
	Tolerancia	No se puede optimizar



Figura 7.28: Coste - K-Means - Número de *clusters* - SIEM

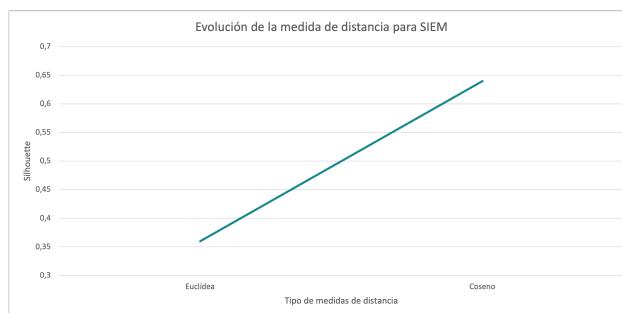
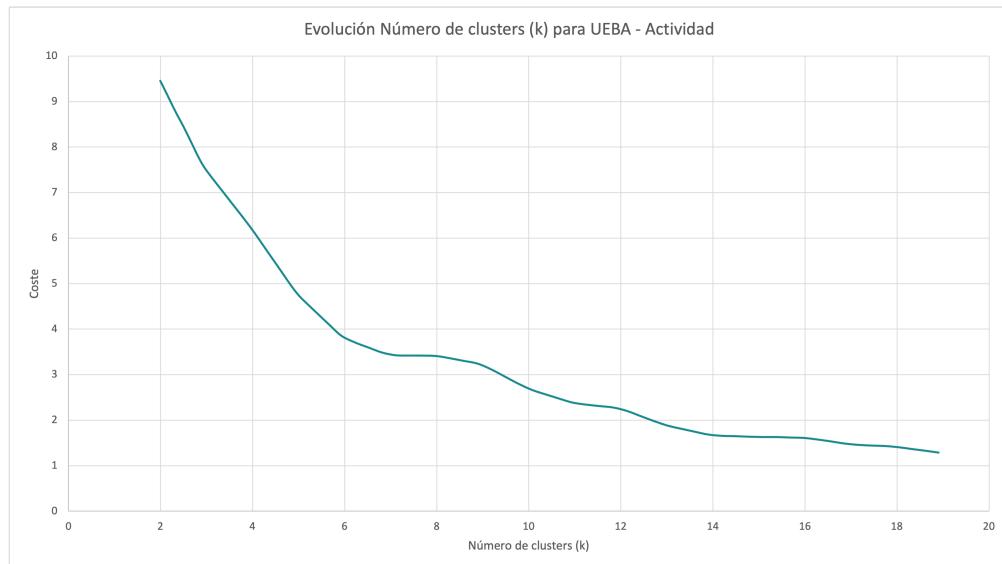


Figura 7.29: Métrica Silhouette - K-Means - Medida de la distancia - SIEM

Tabla 7.21: Valores de los hiper-parámetros seleccionados para los datos de actividad (UEBA)

Modelo	Hiper-parámetro	Valor
K-Means	Número de <i>clusters</i> (k)	$k = 2$
	Medida de distancia	Euclídea
	Máximas iteraciones	Por defecto: 100
	Tolerancia	Por defecto: 10^{-4}
<i>Bisecting K-Means</i>	Número de <i>clusters</i> (k)	$k = 8$
	Medida de distancia	Euclídea
	Máximas iteraciones	Por defecto: 100
GMM	Número de <i>clusteres</i> (k)	$k = 2$
	Máximas iteraciones	Por defecto: 100
	Tolerancia	Por defecto: 10^{-4}

**Figura 7.30:** Coste - K-Means - Número de *clusters* - Actividad (UEBA)

Debido a que algunos valores tienen módulo 0, únicamente es posible aplicar la distancia euclídea.

Tabla 7.22: Valores de los hiper-parámetros seleccionados para los datos de buscador (UEBA)

Modelo	Hiper-parámetro	Valor
K-Means	Número de <i>clusters</i> (k)	$k = 7$
	Medida de distancia	Euclídea
	Máximas iteraciones	Por defecto: 100
	Tolerancia	Por defecto: 10^{-4}
<i>Bisecting K-Means</i>	Número de <i>clusters</i> (k)	$k = 7$
	Medida de distancia	Euclídea
	Máximas iteraciones	Por defecto: 100
GMM	Número de <i>clusteres</i> (k)	$k = 9$
	Máximas iteraciones	Por defecto: 100
	Tolerancia	Por defecto: 10^{-4}

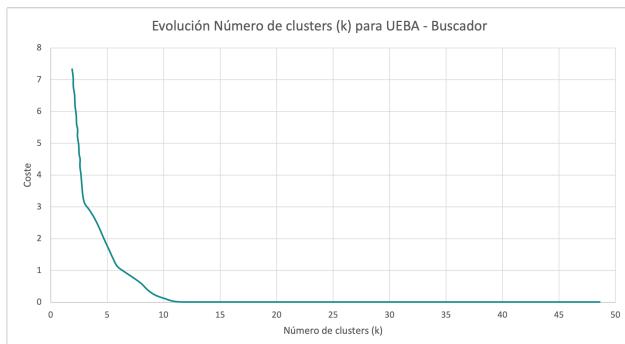


Figura 7.31: Coste - K-Means - Número de *clusters* - Buscador (UEBA)

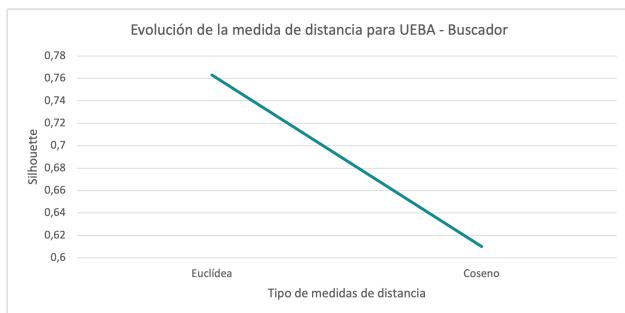


Figura 7.32: Métrica Silhouette - K-Means - Medida de la distancia - Buscador (UEBA)

Tabla 7.23: Valores de los hiper-parámetros seleccionados para los datos de documentos (UEBA)

Modelo	Hiper-parámetro	Valor
K-Means	Número de <i>clusters</i> (k)	k = 5
	Medida de distancia	Coseno
	Máximas iteraciones	Por defecto: 100
	Tolerancia	Por defecto: 10^{-4}
Bisecting K-Means	Número de <i>clusters</i> (k)	k = 3
	Medida de distancia	Coseno
	Máximas iteraciones	Por defecto: 100
GMM	Número de <i>clusteres</i> (k)	k = 6
	Máximas iteraciones	Por defecto: 100
	Tolerancia	700

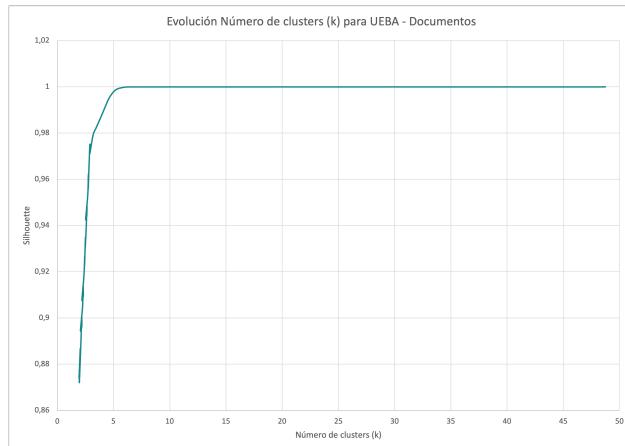
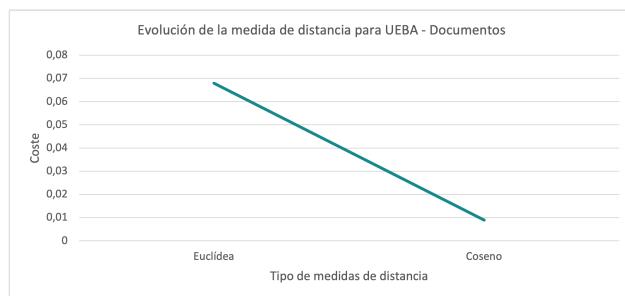
**Figura 7.33:** Métrica Silhouette - K-Means - Número de *clusters* - Documentos (UEBA)**Figura 7.34:** Coste - K-Means - Medida de la distancia - Documentos (UEBA)

Tabla 7.24: Valores de los hiper-parámetros seleccionados para los datos de red (UEBA)

Modelo	Hiper-parámetro	Valor
K-Means	Número de <i>clusters</i> (k)	k = 4
	Medida de distancia	Coseno
	Máximas iteraciones	Por defecto: 100
	Tolerancia	Por defecto: 10^{-4}
Bisecting K-Means	Número de <i>clusters</i> (k)	k = 4
	Medida de distancia	Coseno
	Máximas iteraciones	Por defecto: 100
GMM	Número de <i>clusteres</i> (k)	k = 8
	Máximas iteraciones	Por defecto: 100
	Tolerancia	Por defecto: 10^{-4}

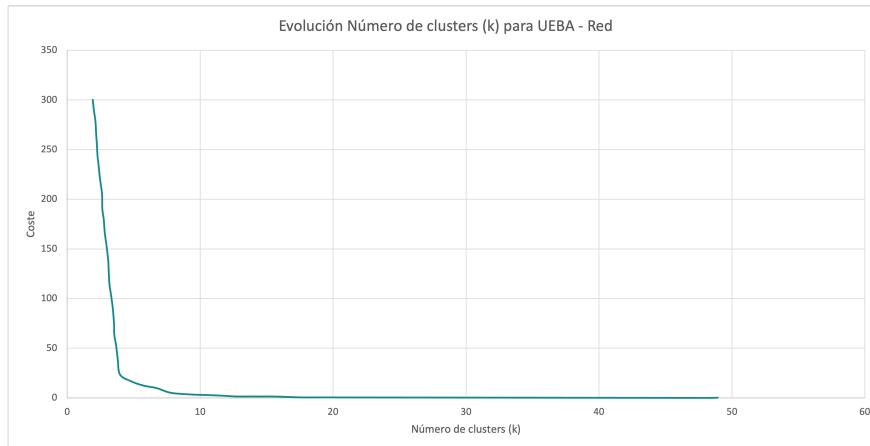
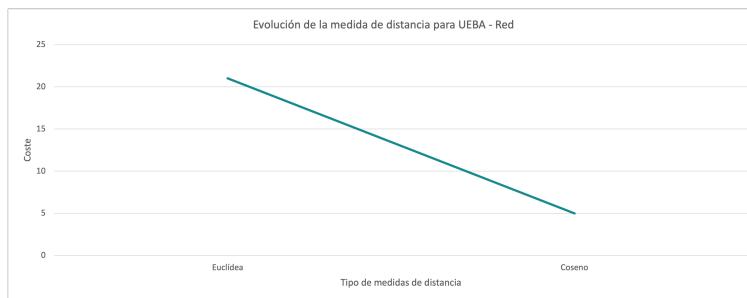
**Figura 7.35:** Coste - K-Means - Número de *clusters* - Red (UEBA)**Figura 7.36:** Coste - K-Means - Medida de la distancia - Red (UEBA)

Tabla 7.25: Valores de los hiper-parámetros seleccionados para los datos de procesos (UEBA)

Modelo	Hiper-parámetro	Valor
K-Means	Número de <i>clusters</i> (k)	k = 5
	Medida de distancia	Coseno
	Máximas iteraciones	Por defecto: 100
	Tolerancia	Por defecto: 10^{-4}
Bisecting K-Means	Número de <i>clusters</i> (k)	k = 4
	Medida de distancia	Coseno
	Máximas iteraciones	Por defecto: 100
GMM	Número de <i>clusteres</i> (k)	k = 7
	Máximas iteraciones	Por defecto: 100
	Tolerancia	Por defecto: 10^{-4}

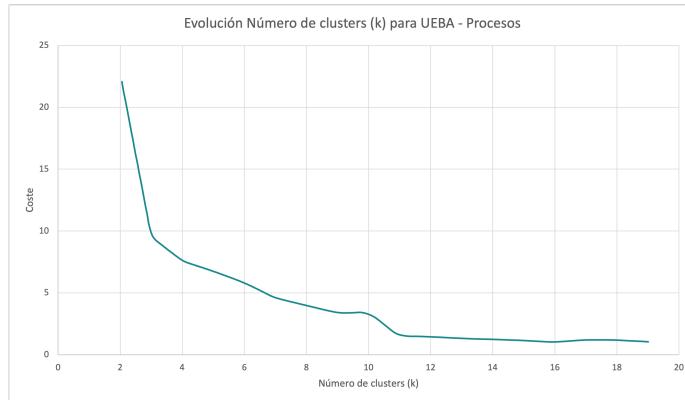
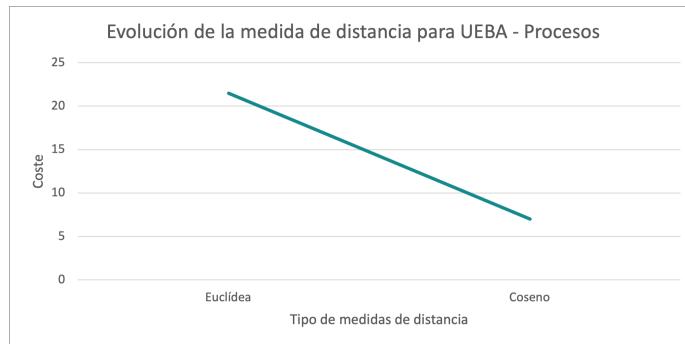
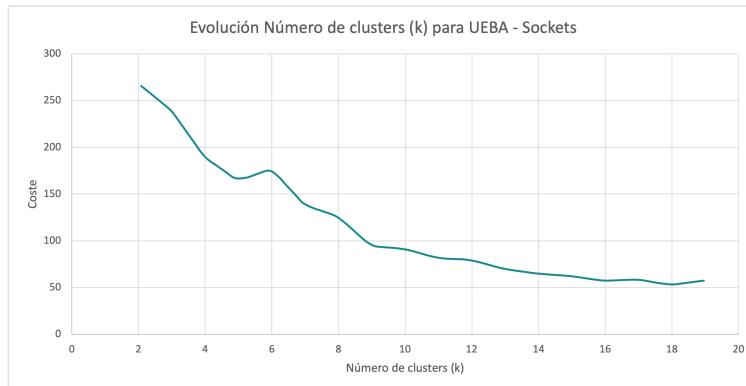
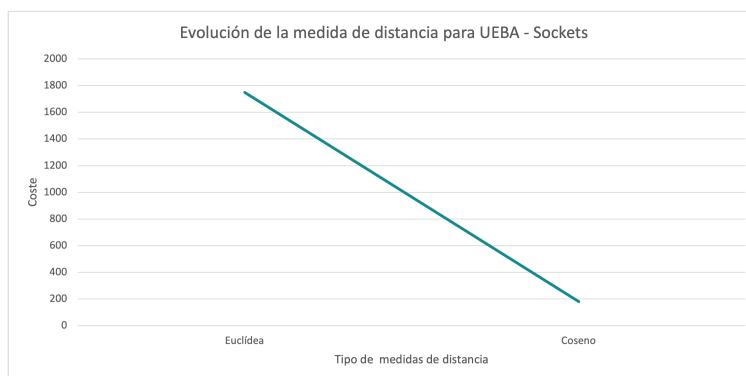
**Figura 7.37:** Coste - K-Means - Número de *clusters* - Procesos (UEBA)**Figura 7.38:** Coste - K-Means - Medida de la distancia - Procesos (UEBA)

Tabla 7.26: Valores de los hiper-parámetros seleccionados para los datos de *sockets* (UEBA)

Modelo	Hiper-parámetro	Valor
K-Means	Número de <i>clusters</i> (k)	k = 7
	Medida de distancia	Coseno
	Máximas iteraciones	Por defecto: 100
	Tolerancia	Por defecto: 10^{-4}
<i>Bisection</i> K-Means	Número de <i>clusters</i> (k)	No se puede optimizar
	Medida de distancia	No se puede optimizar
	Máximas iteraciones	No se puede optimizar
GMM	Número de <i>clusteres</i> (k)	k = 8
	Máximas iteraciones	Por defecto: 100
	Tolerancia	Por defecto: 10^{-4}

**Figura 7.39:** Coste - K-Means - Número de *clusters* - *Sockets* (UEBA)**Figura 7.40:** Coste - K-Means - Medida de la distancia - *Sockets* (UEBA)

Para la selección de hiper-parámetros se ha buscado minimizar el coste y maximizar el valor de Silhouette en la elección del tipo de distancia. A la hora de elegir el número de *clusters*, se ha tenido en cuenta el punto de codo de las gráficas mostradas. El elevado coste computacional del entrenamiento del modelo Bisecting K-Means automáticamente lo descarta como solución viable para la propuesta.

7.4.5 Gestión en tiempo real

El sub-sistema de procesamiento en tiempo real se encarga de identificar las posibles anomalías en los datos que los dispositivos captan en tiempo real, utilizando los modelos previamente entrenados y validados. En la Figura 7.41 se presenta la arquitectura definida para este sub-sistema.

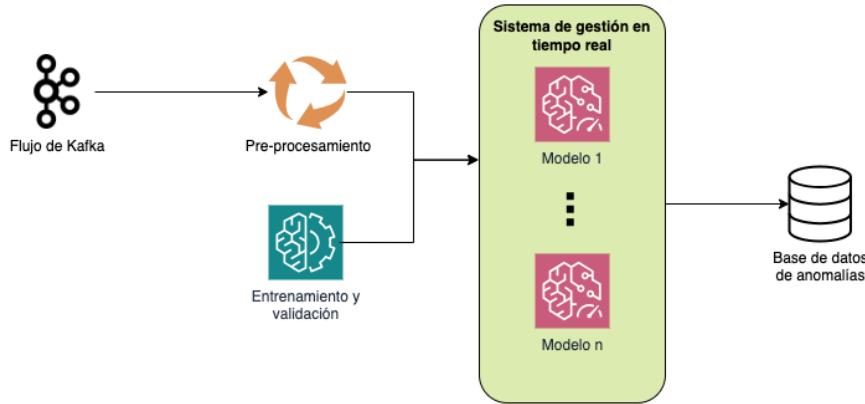


Figura 7.41: Módulo de gestión en tiempo real

Los datos de entrada se reciben mediante los *topics* de Kafka. Después se envían al módulo de pre-procesamiento, que los convierte en datos estructurados para poder utilizarlos en los modelos entrenados y elegidos para cada tipo de datos. La salida del sistema es la identificación de los eventos como normales o anómalos.

La traza de entrada y la etiqueta asignada se almacenan en una base de datos con un identificador único del evento.

7.5 Resultados

7.5.1 Comparación de modelos

Tras la selección de hiper-parámetros para los tres modelos, se entrena y comparan para elegir el que se adapta mejor a cada uno de los dispositivos de entrada. Para ello se comparan las métricas WSSSE, Silhouette y el tiempo de entrenamiento. Los tres modelos asociados a cada fuente se entrena con el mismo conjunto de datos y así poder extraer de los resultados el rendimiento de cada modelo.

Esta evaluación se presenta en la Tabla 7.27, donde destaca que el modelo K-Means obtiene métricas similares o incluso mejores que el *Bisecting* K-Means, reduciendo el tiempo de entrenamiento significativamente. Además, el modelo GMM obtiene los resultados más bajos. Esto se debe a que este algoritmo tiende a quedar atrapado en mínimos locales. Aun así, el tiempo de entrenamiento es similar al obtenido con K-Means.

Tabla 7.27: Comparación de los modelos entrenados para cada fuente

Fuente	Modelo	WSSSE	Silhouette	T. Entrenamiento
Redes móviles	K-Means	87393.66	0.61	5.39 s
	Bisecting K-Means	-	-	Más de 60s
	GMM	Ninguno	0.14	22.5 s
Radiofrecuencia	K-Means	3.38	0.815	1.01 s
	Bisecting K-Means	3.38	0.815	4.15 s
	GMM	Ninguno	0.12	1.86 s
<i>Bluetooth</i>	K-Means	3.47	0.62	0.91 s
	Bisecting K-Means	3.47	0.62	7.01 s
	GMM	Ninguno	0.57	0.82 s
Wi-Fi	K-Means	8571.55	0.50	1.13 s
	Bisecting K-Means	-	-	Más de 60 s
	GMM	Ninguno	0.20	2.81 s
Cortafuegos	K-Means	1.99	0.71	3.35 s
	Bisecting K-Means	3.89	0.61	25.9 s
	GMM	Ninguno	0.14	3.53 s
SIEM	K-Means	23.19	0.64	15.20 s
	Bisecting K-Means	-	-	-
	GMM	-	-	-
UEBA - Actividad	K-Means	9.49	0.66	3.66 s
	Bisecting K-Means	4.58	0.35	19.10 s
	GMM	Ninguno	0.61	1.07 s
UEBA - Buscador	K-Means	0.87	0.76	0.89 s
	Bisecting K-Means	1.01	0.66	18.7 s
	GMM	Ninguno	0.62	1.53 s
UEBA - Documentos	K-Means	0.01	0.99	0.86 s
	Bisecting K-Means	0.36	0.97	13.20 s
	GMM	Ninguno	0.99	1.60 s
UEBA - Red	K-Means	5.03	0.96	0.75 s
	Bisecting K-Means	5.07	0.96	11.6 s
	GMM	Ninguno	0.94	1.63 s
UEBA - Procesos	K-Means	6.72	0.88	0.78 s
	Bisecting K-Means	8.01	0.86	11.90 s
	GMM	Ninguno	0.69	1.64 s
UEBA - Sockets	K-Means	137.82	0.49	11.30 s
	Bisecting K-Means	-	-	Más de 60 s
	GMM	Ninguno	0.48	10.9 s

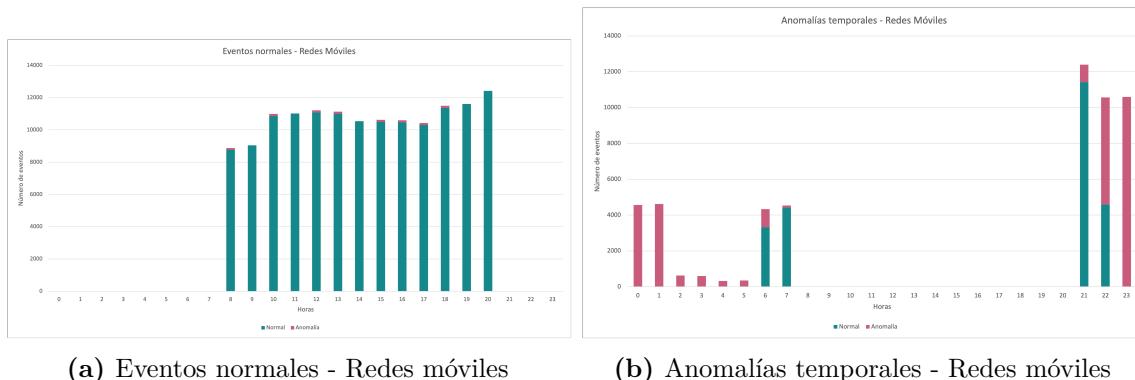
Teniendo en cuenta estos resultados, en el escenario presentado K-Means es el modelo que mejor rinde, y por tanto será el elegido para la detección de anomalías. No presenta problemas de convergencia o relacionados con el tiempo de entrenamiento que eviten que se pueda aplicar a alguno de los conjuntos de datos, lo que sí ocurre con los otros dos algoritmos.

7.5.2 Ratio de detección

Para evaluar el funcionamiento del sistema y cómo se comporta, se plantean tres tipos de pruebas para cada fuente: identificar como datos normales los datos que se consideren no anómalos, identificar como posibles anomalías los datos que no se han recibido anteriormente y que se diferencien lo suficiente del conjunto de entrenamiento, e identificar como anomalías los datos con los que el modelo ha entrenado pero desplazados temporalmente a una hora no común, como las noches o las madrugadas.

Se forman dos conjuntos de datos por cada sensor para llevar a cabo estas pruebas, uno con datos normales y el otro con anomalías. Para el tercer test se utiliza el primer *dataset* desplazado en el tiempo.

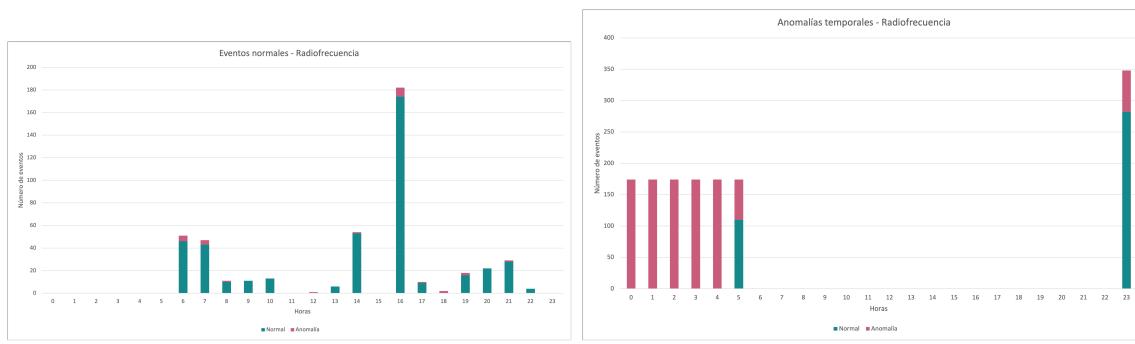
En las Figuras 7.42-7.47, se presentan los desplazamientos temporales realizados para los datos. El color verde representa que el evento se ha clasificado como normal, mientras que el color rosa, representa los clasificados como anomalías.



(a) Eventos normales - Redes móviles

(b) Anomalías temporales - Redes móviles

Figura 7.42: Clasificación de eventos - Redes móviles



(a) Eventos normales - Radiofrecuencia

(b) Anomalías temporales - Radiofrecuencia

Figura 7.43: Clasificación de eventos - Radiofrecuencia

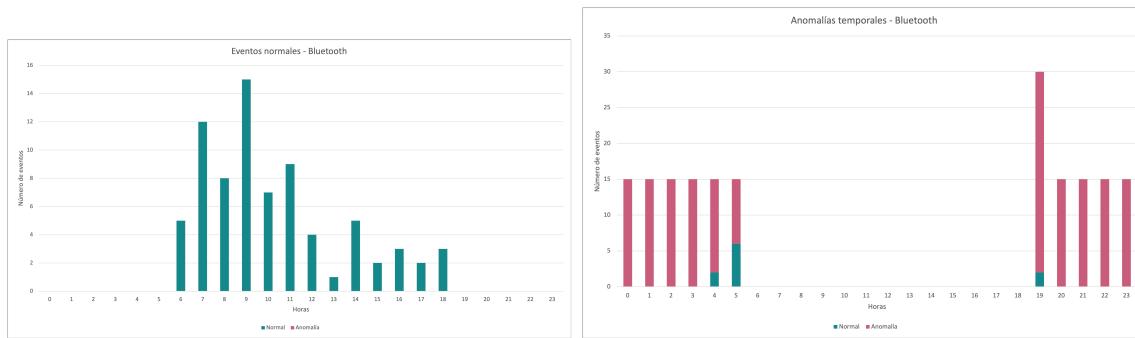


Figura 7.44: Clasificación de eventos - *Bluetooth*

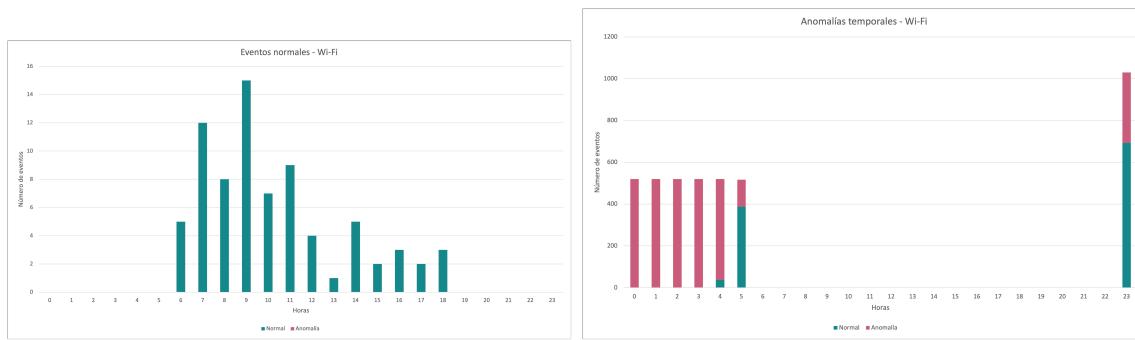


Figura 7.45: Clasificación de eventos - *Wi-Fi*

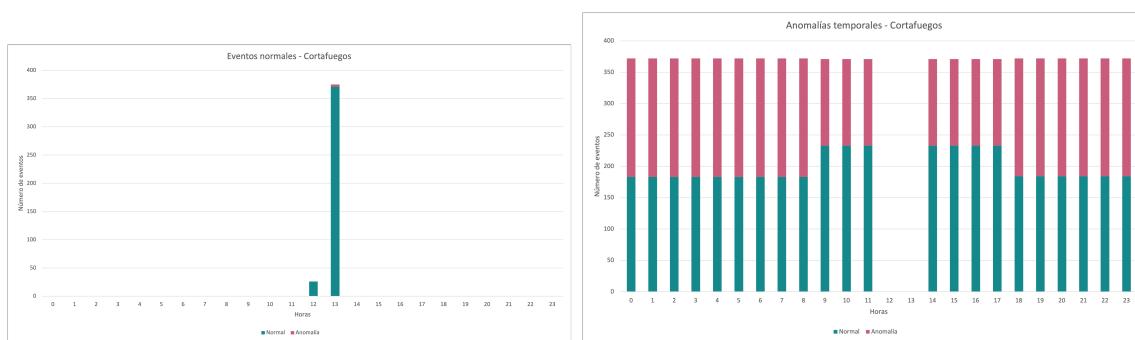
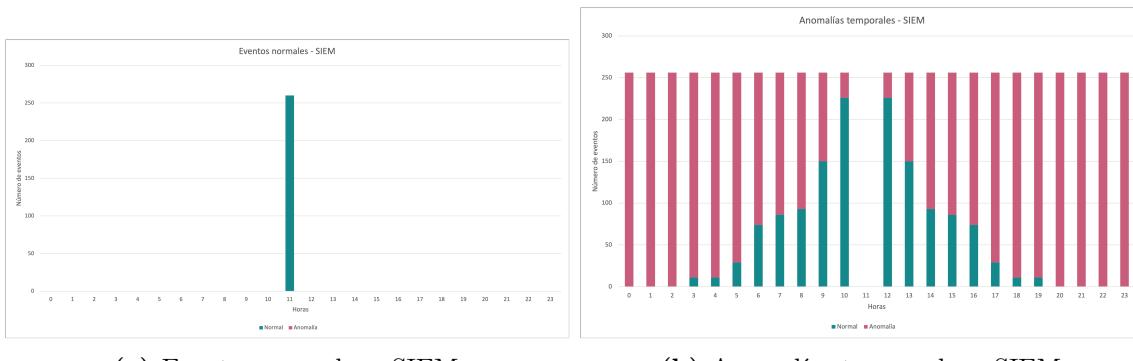


Figura 7.46: Clasificación de eventos - *Cortafuegos*



(a) Eventos normales - SIEM

(b) Anomalías temporales - SIEM

Figura 7.47: Clasificación de eventos - SIEM

En la Tabla 7.28 se muestra el resultado en términos de exactitud.

Tabla 7.28: Exactitud en la detección de anomalías

Fuente	Tráfico Normal	Posibles Anomalías	Anomalías Temporales
Redes móviles	98.64 %	96.00 %	56.65 %
Radiofrecuencia	94.56 %	98.00 %	71.84 %
<i>Bluetooth</i>	99.32 %	99.26 %	94.44 %
Wi-Fi	98.90 %	99.30 %	72.89 %
Cortafuegos	97.98 %	96.00 %	76.97 %
SIEM	99.00 %	98.75 %	46.51 %
UEBA - Actividad	99.00 %	99.30 %	-
UEBA - Buscador	98.00 %	86.00 %	-
UEBA - Documentos	99.00 %	97.63 %	-
UEBA - Red	96.00 %	95.00 %	-
UEBA - Procesos	97.66 %	99.60 %	-
UEBA - <i>Sockets</i>	98.90 %	95.00 %	-

Los resultados muestran que los datos normales y las posibles anomalías se clasifican correc-

tamente. Sin embargo, las anomalías temporales tienen resultados considerablemente más bajos, detectándose principalmente en el rango entre la 1 y las 5 de la madrugada, donde el número de eventos es menor. El error en los eventos desplazados no es uniforme, aumenta cerca de las horas límite definidas como normales. Para las fuentes de datos UEBA no se tuvo en cuenta la detección de este tipo de anomalías, debido a los resultados obtenidos en la detección de anomalías temporales con el resto de sensores y los resultados obtenidos en la detección de posibles anomalías con los modelos entrenados para UEBA, esperando resultados poco fiables. En el futuro, por tanto, será necesario profundizar en el entrenamiento de este tipo de anomalías temporales.

7.5.3 Visualización de *clusters*

Otra forma de evaluar la creación de *clusters* es mediante la visualización de los grupos creados por los distintos modelos, comprobando si la forma es adecuada para aplicar el umbral de la detección de anomalías.

Para poder visualizar los *clusters* se requiere la reducción de dimensión y poder representar los datos en un plano. Se aplican cuatro tipos de algoritmos (PCA, ISOMAP, t-SNE, UMAP), manteniendo el que permite la mejor visualización, donde se pueda verificar que las anomalías se detectan correctamente. Estos resultados se muestran en la Figura 7.48.

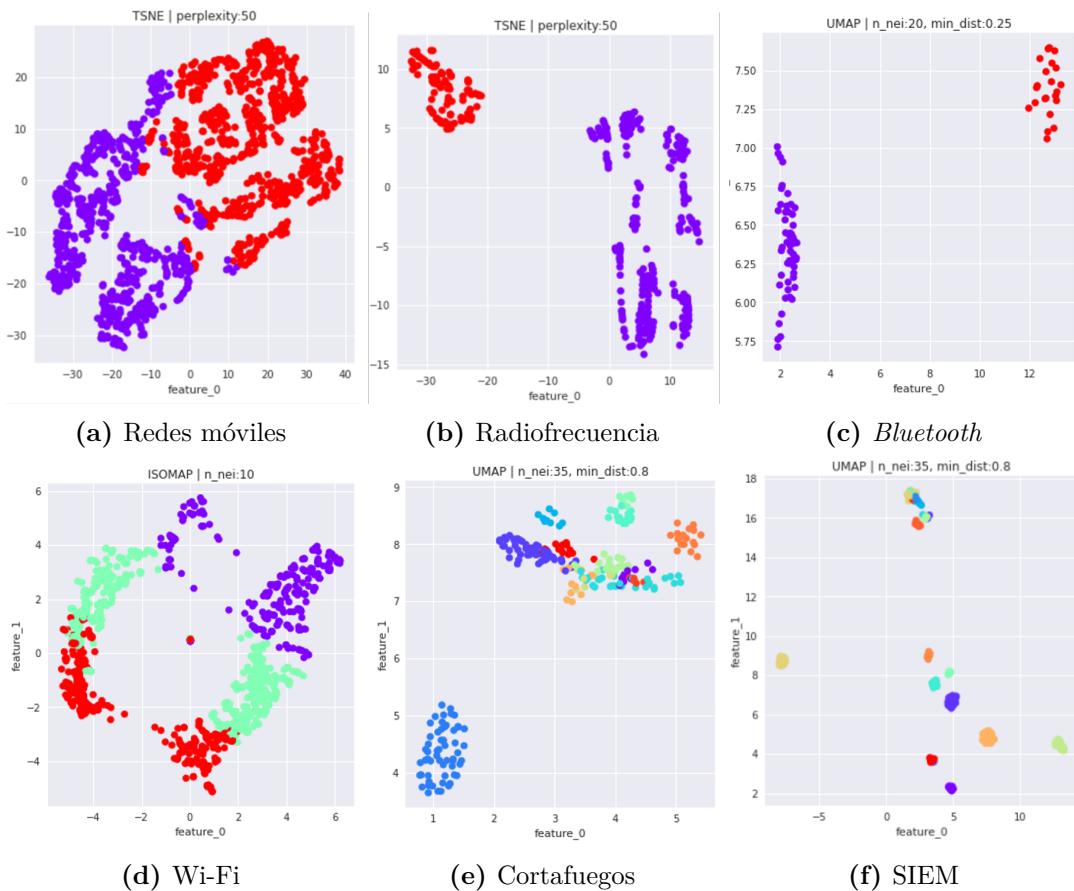


Figura 7.48: Representación de los *clusters* de cada fuente

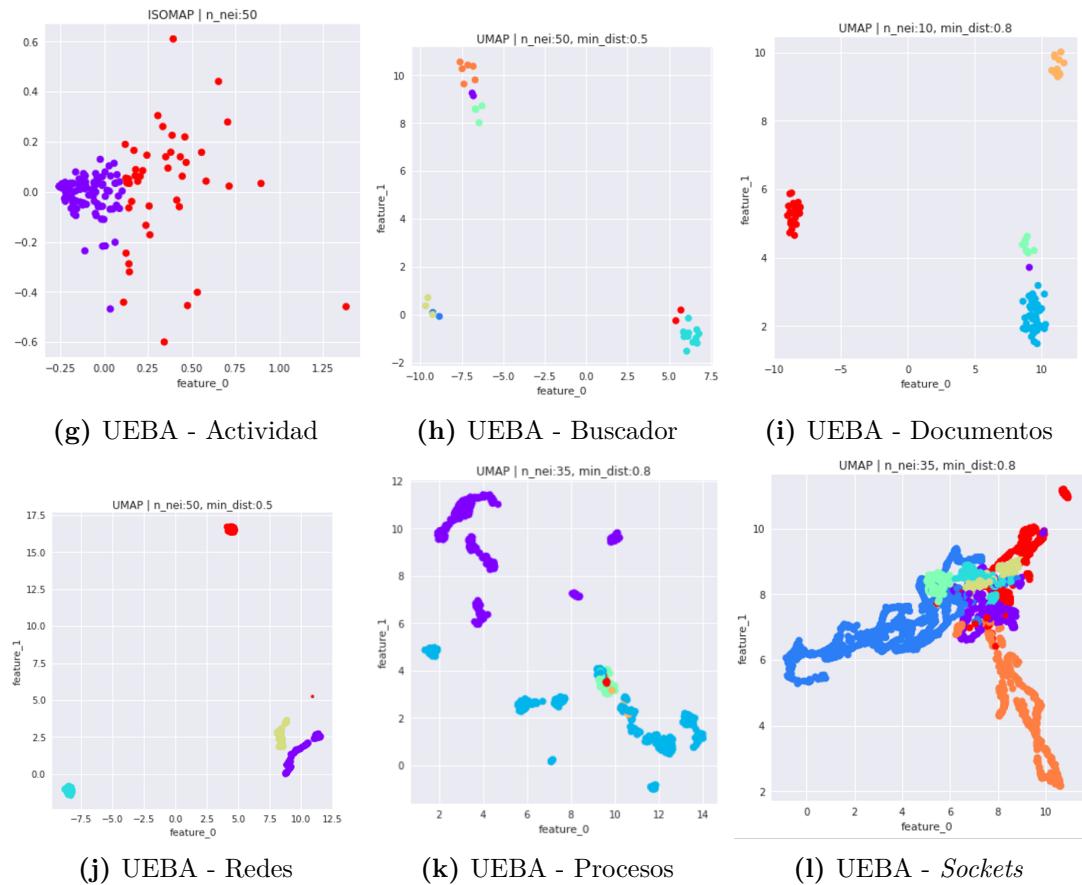


Figura 7.48: Representación de los *clusters* de cada fuente

7.5.4 Comparación con trabajos anteriores

Los trabajos presentados en la Sección 7.2 se recogieron en una comparación que se completa en la Tabla 7.29, incluyendo los resultados obtenidos en esta propuesta. En ella se resumen las metodologías aplicadas, buscando identificar las ventajas de este trabajo en comparación a los resultados anteriores.

Destaca principalmente que este sistema ha obtenido resultados de calidad incluyendo, como puntos clave del desarrollo, distintas técnicas como el procesado en tiempo real y el estudio y pre-procesado de los datos recogidos por los dispositivos.

Tabla 7.29: Comparación de los trabajos previos con esta propuesta

Investigación	<i>Dataset</i>	Pre-procesado	Modelos	Tiempo Real
[148]	Propio	Sí	K-Means	Sí
[99]	KDDCUP99	No	Mini-Batch K-Means y PCA	No
[49]	Repositorio UCI Machine Learning	No	K-Means y aproximación basada en distancia	No
[100]	NSL-KDD	Sí	SCC-OCSVM	No
[87]	Yahoo Webscope	No	Modelos de aprendizaje profundo y estadístico	No
[14]	SWaT, WADI, SMD, SMAP, MSL y interno	Sí	<i>Autoencoders</i>	No
[63]	Simulación sistema aero-propulsión	Sí	<i>Isolated Forest</i>	Sí
[103]	Propio	No	AAE	No
[81]	Propio	Sí	<i>Isolated Forest</i>	Sí
Esta propuesta	Propio	Sí	K-Means	Sí

7.6 Conclusiones

Como resultado del desarrollo presentado en este capítulo, tras exponer y validar los módulos que componen el sistema de detección de anomalías en tiempo real, las conclusiones que se extraen se centran en cuatro puntos principales.

Por una parte, en el análisis de la literatura se ha podido contextualizar el problema que trata este desarrollo, la detección de anomalías en tiempo real entrenando modelos con datos no etiquetados procedentes de sensores heterogéneos. A lo largo del capítulo se ha presentado una solución adaptada a los datos de entrada, que consistían en un conjunto de sensores físicos y lógicos: redes móviles, radiofrecuencia, *Bluetooth*, Wi-Fi, cortafuegos, SIEM y un conjunto de dispositivos que analizan el comportamiento de usuario (UEBA). Estos datos se caracterizan

y analizan para convertirlos en entradas adecuadas a los distintos modelos.

Además, los modelos elegidos (K-Means, *Bisecting* K-Means y GMM), se configuran con los hiper-parámetros óptimos para los datos de entrada en base a las métricas WSSSE, Silhouette y el tiempo de ejecución y el coste computacional, lo que es básico para un buen desempeño de los algoritmos. Una vez configurados, el modelo que proporciona mejores resultados en el problema de detección mediante la definición de umbral es K-Means, claramente superior en tiempo de ejecución a *Bisecting* K-Means, y con mejores resultados en las métricas que GMM.

El modelo óptimo se entrena con datos considerados normales, conformando distintos *clusters* y definiendo un umbral para cada una de las fuentes de información. Por defecto, el umbral se establece en el punto más lejano al centroide de su grupo, pero este valor puede ajustarse como sea necesario.

Los resultados obtenidos permiten concluir que la detección de eventos normales y posibles anomalías han tenido un buen rendimiento, obteniendo exactitudes (*accuracy*) superiores al 97 % en muchos casos. Sin embargo, las anomalías temporales son un punto de mejora, ya que el rendimiento es bastante inferior en comparación con el resto de detecciones, especialmente en horarios cercanos al límite con el comportamiento normal.

Este sistema formará parte de un entorno de conciencia cibersituacional, compuesto por los distintos elementos que se presentan en los próximos capítulos, pero tras el desarrollo presentado en este apartado queda validada la Hipótesis 1 de la Tesis Doctoral: los modelos no supervisados presentan buenos resultados en términos de exactitud en la detección de ciberataques con datos de fuentes heterogéneas (que no se asemejan a los de ningún *dataset* conocido) a la hora de entrenar un IDS para detectar el comportamiento normal de una red.

Capítulo 8

Propuesta para la caracterización de técnicas MITRE ATT&CK en ciberataques

En este capítulo se desarrolla un sistema que permite identificar y caracterizar un conjunto de técnicas MITRE ATT&CK en ataques de ciberseguridad. Como parte del modelo global (Figura 6.1), este sistema realiza la función principal de caracterización de ciberataques, enriqueciendo el proceso de recomendación de contramedidas y gestión de riesgos al asociar los TTPs a mitigaciones especializadas o la explotación de una debilidad concreta relacionada con una vulnerabilidad. Esta contribución se resalta sobre la arquitectura global en la Figura 8.1.

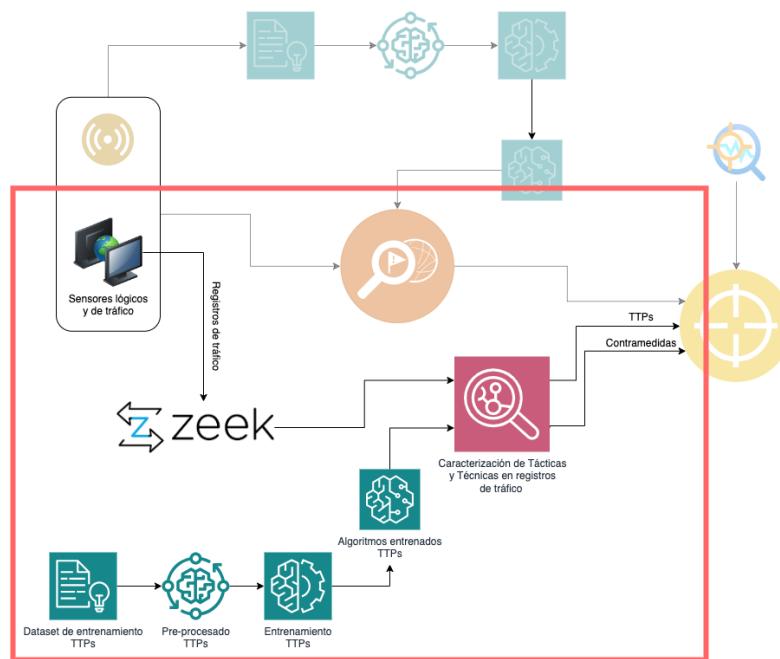


Figura 8.1: Módulo de caracterización de TTPs en la arquitectura global propuesta

Tras haber analizado el estado del arte en la Sección 3.4, en este capítulo se hará una revisión de la literatura publicada (Sección 8.2) y se presentará el desarrollo llevado a cabo (Secciones 8.3-8.4) incluyendo los resultados obtenidos (Sección 8.5) y las conclusiones que se extraen del estudio (Sección 8.6).

8.1 Introducción

Para conseguir la caracterización de los ciberataques, es imprescindible la identificación de las técnicas que se asocian con ellos. Esta información tiene múltiples aplicaciones, por lo que es un objetivo frecuente en las investigaciones. De este conocimiento se pueden extraer relaciones con otras bases de datos como CWE, CVE o CAPEC y principalmente, como se presentó en la Sección 3.4 mediante la matriz ATT&CK, con un conjunto de mitigaciones para salvaguardar el sistema y preparar el entorno para reaccionar ante las posibles amenazas [15].

Los ataques complejos están generando una evolución en las herramientas de ciberseguridad para mejorar las protecciones disponibles, desarrollando un papel fundamental en el diseño de sistemas avanzados que sean capaces de identificar estos incidentes y APTs. En este contexto, el marco de referencia MITRE ATT&CK [85] permite entender la sofisticación de los ataques y su implementación para la clasificación de los TTPs empleados por los atacantes.

Enmarcado dentro del ámbito de CTH, en este capítulo se aborda la necesidad de un sistema de aprendizaje automático que, analizando registros de tráfico, permita detectar técnicas MITRE ATT&CK, almacenando la información en una ontología para la recomendación óptima de contramedidas en tiempo real. Este enfoque combina la eficiencia del aprendizaje automático en la detección de amenazas con la capacidad de almacenar y estructurar la información de una ontología, proporcionando una base sólida para la toma de decisiones informadas, mejorando la capacidad de respuesta y fortaleciendo la ciberseguridad en los sistemas protegidos.

La novedad de este sistema radica en su capacidad, no solo de detectar ciberataques sino de identificar las técnicas que representan los registros de tráfico, y poder así proponer contramedidas de acuerdo a la amenaza recibida. A diferencia de las propuestas tradicionales, mediante firmas y reglas, la aplicación de modelos de aprendizaje automático en este tipo de tareas permite extraer relaciones entre los datos que no se perciben a simple vista y, por tanto, identificar técnicas en registros de tráfico que aparentemente no encajan con ninguna de las reglas establecidas, o en casos donde la definición de estas reglas no es posible.

Los modelos entrenados con un *dataset* se validan con las métricas comunes para este tipo de problemas, como la exactitud o *F1-Score*; y el módulo en conjunto mediante un caso de uso donde se ponga a prueba la recomendación de contramedidas.

8.2 Trabajos relacionados

En el análisis de la literatura revisada se abordan diversas facetas del marco MITRE ATT&CK. Siguiendo el ámbito de esta propuesta, existen múltiples trabajos con métodos para detectar

TTPs, resumidos en esta sección. En [6], los autores presentan un conjunto de reglas de asociación para llevar a cabo procesos de atribución en inteligencia de amenazas. Uno de los problemas principales a los que se enfrenta este tipo de proyectos son las grandes cantidades de datos a analizar, que pueden impedir encontrar información relevante. Por ello, introducen un proceso de minado de datos que pueda encontrar relaciones entre *datasets* y llevar a cabo el proceso de atribución de inteligencia de amenazas dentro del ciberataque, como por ejemplo identificar las tácticas y técnicas empleadas.

Otra de las aproximaciones ampliamente utilizada es el uso de grafos de conocimiento. En [66] se presenta un modelo de datos basado en el repositorio de MITRE (CAR) [104] que, combinado con reglas de inferencia, es capaz de detectar técnicas de ataque y mediante modelos de aprendizaje automático entrenados sobre este grafo puede predecir nuevas amenazas. Los autores en [61], a partir de un grafo, de conocimiento buscan mejorar la eficiencia en la detección temprana de riesgos de los sistemas ciber-físicos, basándose en un método de evaluación de riesgos utilizando MITRE ATT&CK. Del grafo obtiene la probabilidad de que un indicador conlleve una amenaza y calculan el valor de riesgo causado por las distintas tácticas y técnicas.

Por otro lado, los autores de RADAR [126][127] presentan un sistema basado en la ontología de TTPs de MITRE ATT&CK para identificar y clasificar comportamiento malicioso. El objetivo es entrenar modelos de aprendizaje automático para detectar y clasificar TTPs con sus correspondientes explicaciones. Identifica cada muestra con una técnica y trata de predecir si es maliciosa o no. Está entrenado para detectar tácticas y técnicas de *Reconnaissance* (T1590), *Credential Access* (T1557.001), *Discovery* (T1124 y T1135), *Lateral Movement* (T1021.001/4, T1550.003, T1563.001/2, T1570), y *Command and Control* (T1071, T1090, T1105, T1571, T1053), con la posibilidad de extenderse. Además, aprovechando el sistema desarrollado, realizan una comprobación para verificar que, al añadir información de TTPs en los datos de entrenamiento, se mejora la detección de amenazas.

Profundizando en la detección con modelos entrenados, aparecen múltiples propuestas con técnicas de aprendizaje no supervisado. En [122] los autores proponen agrupar técnicas mediante un *clustering* jerárquico con un 95 % de acierto al inferir las tácticas. Así, al detectar una técnica, se pueden deducir aquellas que están relacionadas.

Una de las principales aplicaciones de la caracterización de tácticas es la capacidad de asociarlas con las vulnerabilidades que pueden explotar. Mediante modelos de aprendizaje automático [67] se pueden identificar múltiples etiquetas de las tácticas utilizadas para explotar vulnerabilidades, lo que permite priorizar las estrategias de defensa frente a estos posibles ataques. También, los autores en [7] buscan predecir las técnicas más probables para explotar una vulnerabilidad empleando procesamiento del lenguaje natural para mapearlas, con el objetivo de priorizar riesgos y correlacionar los eventos e incidentes que ocurren con la vulnerabilidad.

El procesamiento de lenguaje natural es una de las técnicas más empleadas [79], especialmente entrenando modelos a partir del conjunto de datos de ENISA que permite asociar vulnerabilidades y técnicas. Sin embargo, a pesar de encontrar múltiples trabajos que aprovechan las ventajas de la IA, el entorno en el que se desarrolla esta propuesta es, por naturaleza,

desequilibrado, ya que no todos los ataques se dan en el mismo porcentaje y, por lo tanto, los conjuntos de datos disponibles para entrenamiento tienen un tamaño de las clases muy descompensado [36]. La solución principal a este problema es el pre-procesado de los datos, como en [65], donde el método de clasificación de TTPs en modelos de procesamiento del lenguaje natural con un tratamiento correcto de los datos, permite pasar de un 61,26 % de precisión en el *dataset* TRAM a un 98.76 % en el mismo conjunto.

De la amplitud de trabajos e investigaciones en las que se aplica el marco MITRE ATT&CK, por el contexto de esta propuesta, las referencias principales se encuentran en aquellos que permitan identificar tácticas y técnicas a partir de registros de tráfico, donde la mayoría utilizan algoritmos de aprendizaje automático [110]. En concreto, existe un *dataset* de nuevo desarrollo, que no tiene mucha literatura relacionada, UWF-ZeekDataFall22 [18] con el objetivo principal de identificar las tácticas de *Resource Development* (TA0042), *Reconnaissance* (TA0043) y *Discovery* (TA0007), aunque también incluye muestras de otras tácticas, pero en menor medida. El objetivo principal es clasificar las tácticas, no los ataques, obteniendo un 100 % de precisión en clasificación binaria entrenando modelos individuales para detectar cada táctica y un 99.99 % en la clasificación multi-clase. Existe un antecedente a este dataset, UWF-ZeekData22 [19] que incluye menor volumen de tráfico y, por tanto, se centra en las tácticas principales (*Reconnaissance* y *Discovery*). Permite un 99.4 % de precisión en la detección de *Reconnaissance* y 99.95 % en *Discovery*, entrenando también modelos binarios para identificar si el registro se asocia o no con esas tácticas [16]. Además, mediante un grafo representa la táctica de reconocimiento en busca de patrones que permitan identificarla y etiquetarla [20].

Algunos de los trabajos relacionados con el conjunto de datos UWF-ZeekData22 se centran en el tratamiento de *datasets* no balanceados en el campo de la detección de ataques [17], y resaltan la necesidad de tratar estos datos con sub-muestreo y sobre-muestreo de las clases para obtener resultados coherentes. Los autores de [21] presentan un modelo que utiliza el conjunto de datos anterior para entrenar la clasificación de *malware*. Dentro del pre-procesado se debe balancear el conjunto para poder realizar clasificaciones binarias e identificar con muy buenos resultados las tácticas de *Credential Access*, *Discovery*, *Lateral Movement*, *Reconnaissance*, *Resource development* y el tráfico benigno. A pesar de que las tácticas de *Exfiltration* y *Privilege Escalation* se confunden entre ellas, el resultado total es positivo.

La base del desarrollo presentado en este capítulo será por tanto, entrenar modelos de aprendizaje automático a partir del conjunto de datos UWF-ZeekDataFall22 para identificar técnicas con un algoritmo multi-clase, sin perder precisión a la hora de identificar estas tácticas, un aspecto muy poco abordado hasta el momento.

De todos los trabajos analizados, únicamente se ha podido extraer información relacionada con el número de tácticas y técnicas en los que se recogen en la Tabla 8.1.

Como se puede observar, no se han podido localizar trabajos hasta la fecha que aborden de manera efectiva la caracterización de técnicas específicas de la matriz MITRE ATT&CK mediante el análisis de registros de tráfico con modelos de aprendizaje automático. Esto supone una debilidad a la hora de defender un sistema frente a los ciberataques, que están en constante evolución. En este capítulo se plantea una posible solución a este problema,

Tabla 8.1: Resumen de los resultados de trabajos previos

Publicación	Método	Número de Tácticas	Número de Técnicas
[61]	Grafo	5	5
[126, 127]	Reglas	6	13
[18]	Algoritmos supervisados	3	-
[16]	Algoritmos supervisados	2	-
[21]	Algoritmos supervisados	8	-

mejorando la postura de seguridad en el ámbito digital.

8.3 Propuesta

Para afrontar la problemática identificada, la necesidad de detectar en registros de tráfico las distintas técnicas MITRE, este módulo trata de caracterizar las tácticas y técnicas en datos que representan ciberataques o tráfico normal según sus propiedades. Así, teniendo en cuenta la información recopilada por ATT&CK y presentada en la Sección 3.4, se pueden ofrecer las contramedidas recomendadas por este marco frente a los incidentes.

8.3.1 Alternativa: Decisión mediante reglas

La alternativa principal a la propuesta que se presenta en este capítulo es la detección mediante reglas. Éstas se definen tras analizar minuciosamente las técnicas y las características de los registros de tráfico estándar, estableciendo qué anomalías en cada atributo podrían representar la utilización de alguna de las técnicas ATT&CK. Por ejemplo, la técnica T1571 se podría identificar comprobando los protocolos utilizados y el puerto que tienen por defecto. Si en el registro se refleja un puerto distinto, podría ser un indicador de ciberataque mediante esta técnica. Sin embargo, un ejemplo donde la regla sería más compleja sería la T1071, en la que es necesario conocer el comportamiento normal de la red, para establecer un umbral en la diferencia entre el tamaño de los datos que entran y salen del sistema y así asociar esta técnica al registro. En cambio, este trabajo es trivial para los algoritmos de aprendizaje automático, que podrían realizar esta asociación de forma mucho más eficiente.

La definición de las reglas se podría complementar con las analíticas definidas por MITRE en el repositorio *Cyber Analytics Repository* (CAR) [104], que almacena análisis basados en ATT&CK. Proporciona un conjunto de análisis validados para poder detectar técnicas y tácticas, ejemplos en pseudocódigo y pruebas unitarias para poder implementar la identificación. En primer lugar, se localizan los comportamientos anómalos según ATT&CK, a continuación se identifican los datos necesarios para comprobarlos, se obtiene un sensor para recoger esos

datos y se lleva a cabo la analítica.

Una solución desarrollada y probada actualmente es el *Decider* de CISA [105], una herramienta gratuita para asociar comportamiento de atacantes con el marco ATT&CK a través de un conjunto de preguntas, un motor de búsqueda y un filtro. Esta metodología lo descarta como alternativa, ya que el objetivo es la caracterización de los ciberataques en tiempo real a partir de registros de tráfico de red.

Sin lugar a dudas, el enfoque mediante reglas podría ser más preciso y escalable, puesto que no depende de la eficiencia de un modelo y de los datos de entrada, pero genera situaciones que se deben analizar minuciosamente, porque igualmente podría haber registros que encajasen con varias reglas, o que se detecten como maliciosos sin que realmente conlleven un ciberataque. Por otro lado, al disponer de un conjunto de datos etiquetado para el entrenamiento, los modelos de aprendizaje automático realizan el análisis de los datos; y al basar su decisión en probabilidades, se identifica una técnica y se podrían conocer qué otras posibilidades existen, y con qué porcentaje de seguridad el modelo realiza las clasificaciones. Esta simplicidad sin perder potencia es el motivo principal de la aplicación de los modelos de IA en lugar de la definición de reglas.

8.3.2 Diseño de la metodología propuesta

Una vez identificada la ventaja de los sistemas de aprendizaje automático sobre la detección con reglas en este tipo de entornos heterogéneos, y la existencia de un conjunto de datos que permita el entrenamiento de estos modelos, se define el diseño de una metodología que permita la caracterización de ciberataques a partir de la identificación de técnicas en sus registros.

En la Figura 8.2 se presenta la metodología propuesta para esta caracterización, profundizando en el área resaltada en la arquitectura global de la Tesis Doctoral (Figura 8.1).

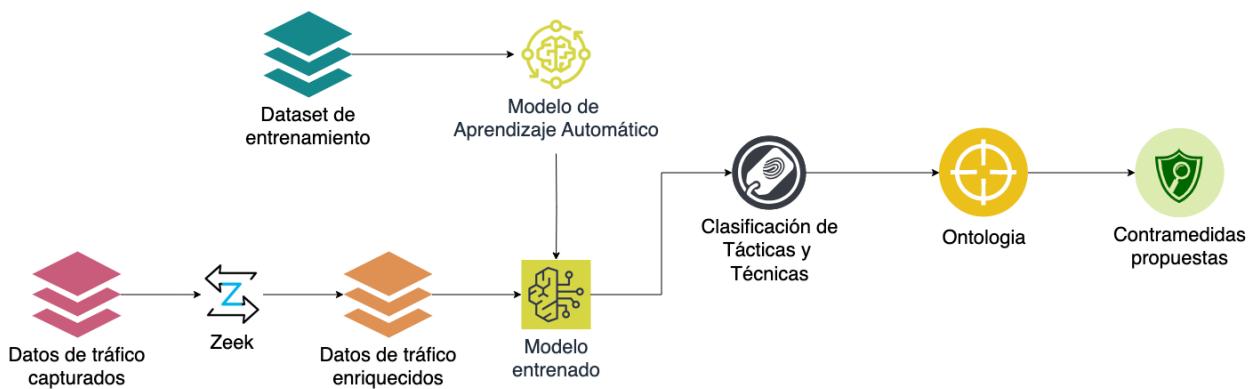


Figura 8.2: Metodología propuesta para la caracterización de TTPs

El sistema captura tráfico de red en tiempo real, y complementa las propiedades identificadas aplicando un módulo Zeek [146]. El modelo de aprendizaje automático entrenado para caracterizar técnicas MITRE se integra con una ontología que lleve a cabo una gestión de

riesgos como la que forma parte del entorno de conciencia cibersituacional de esta Tesis Doctoral, aprovechando la información para enriquecer y complementar procedimientos de análisis y gestión de riesgos con las contramedidas asociadas a estas técnicas. Para la validación del proceso de recomendación se propondrá un caso de uso sobre una ontología más sencilla, que permitirá seguir los procesos y razonamientos para evaluar su funcionamiento.

El método de selección y entrenamiento de los modelos de aprendizaje supervisado comienza con el pre-procesado del conjunto de datos con el fin de obtener las características idóneas para el entrenamiento y la validación del modelo. A continuación se eligen los algoritmos, se optimizan sus hiper-parámetros y se entrena para poder seleccionar el que mejor resultado obtenga a partir de los datos del *dataset* UWF-ZeekDataFall22.

Por otro lado, los datos capturados en tiempo real se enriquecen hasta obtener las mismas características que el *dataset*, y se analizan con los modelos entrenados, obteniendo las técnicas identificadas. A partir de un mapeo, se extrae otra información relacionada con los TTPs, como las mitigaciones o los CAPECs, y estos datos se almacenan en la ontología. Cuando se realiza la inferencia, el sistema propone una mitigación para el riesgo producido por el incidente.

8.4 Desarrollo de la solución

Para conseguir el objetivo propuesto de caracterizar tácticas y técnicas MITRE ATT&CK en registros de tráfico y extraer información útil para la gestión de riesgos -impactos, contramedidas o debilidades-, el desarrollo realizado se divide en dos módulos: análisis y pre-procesado de los datos para adaptarlos al modelo; y evaluación de distintos algoritmos y entrenamiento del que mejor se ajuste al entorno de trabajo.

8.4.1 Pre-procesado del conjunto de datos UWF-ZeekDataFall22

En cuanto al conjunto de datos de entrenamiento, UWF-ZeekDataFall22 [18], disponible en [43], que ya se mencionó brevemente en la Sección 8.2 entre los trabajos relacionados, es un conjunto que destaca entre la literatura analizada ya que permite evaluar esta investigación frente a estudios anteriores centrados en la caracterización de TTPs sin necesidad de aplicar el mismo método. Además, establece una base para el entrenamiento de modelos supervisados aplicados a esta tarea y que puede validarse con distintos *datasets*.

Consta de información de tráfico enriquecida mediante un módulo Zeek y etiquetada con 10 tácticas distintas y con las 22 técnicas presentadas en la Sección 3.4. Las columnas incluidas son: ‘community_id’, ‘conn_state’, ‘duration’, ‘history’, ‘src_ip_zeek’, ‘src_port_zeek’, ‘dest_ip_zeek’, ‘dest_port_zeek’, ‘local_orig’, ‘local_resp’, ‘missed_bytes’, ‘orig_bytes’, ‘orig_ip_bytes’, ‘orig_pkts’, ‘proto’, ‘resp_bytes’, ‘resp_ip_bytes’, ‘resp_pkts’, ‘service’, ‘ts’, ‘uid’, ‘datetime’, ‘label_tactic’, ‘label_technique’ y ‘label_binary’.

El reparto de tráfico inicial del conjunto de datos será 346.933 entradas marcadas como tráfico malicioso ('label_binary' igual a *True*), 350.339 registros normales ('label_binary' igual a *False*) y 3.068 filas marcadas como duplicadas. En relación con las tácticas, aparecen representadas *Reconnaissance* (TA0043), *Initial Access* (TA0001), *Discovery* (TA0007), *Command and Control* (TA0011), *Lateral Movement* (TA0008), *Collection* (TA0009), *Persistence* (TA0003), *Execution* (TA0002), *Defense Evasion* (TA0005) y *Resource Development* (TA0042). Por otra parte, las técnicas que contiene son: T1046, T1059, T1071, T1112, T1133, T1136, T1190, T1203, T1204, T1210, T1505, T1546, T1547, T1548, T1557, T1566, T1571, T1587, T1589, T1590, T1592, T1595.

Sin embargo, por las características del entorno, no todas las clases están igualmente representadas en el conjunto, y por lo tanto es necesario tratar los datos hasta que sean aptos para entrenar un modelo de aprendizaje automático.

En primer lugar, los archivos del conjunto se descargan de la fuente por separado, por lo que el primer paso consiste en unir todos los archivos para obtener un *dataset* con los registros de todos los períodos de tiempo capturados. A continuación, se eliminan los registros marcados como duplicados. Así, se obtienen finalmente 697.272 filas de tráfico benigno e incidentes. Despues se eliminan los registros con valores *Not a Number (NaN)* o *Null*, que no permiten entrenar los modelos correctamente. Tras este paso, en el conjunto de datos se mantienen las técnicas T1046, T1548, T1210, T1587, T1557, T1566, T1590, T1190, T1592, T1595, T1589 y T1071, que corresponden a las tácticas *Lateral Movement*, *Discovery*, *Initial Access*, *Collection*, *Command and Control*, *Resource Development*, *Reconnaissance* y *Defense Evasion*.

Examinando los valores de las columnas, existen dos que representan identificadores ('*uid*' y '*community_id*'), que toman valores distintos para cada muestra, por lo tanto no aportarán información suficiente al modelo y se eliminan. El siguiente paso en la limpieza del *dataset* es identificar las columnas con valores no numéricos, cuyos datos no puede procesar el algoritmo. La solución óptima es aplicar codificadores de etiquetas a las columnas '*conn_state*', '*history*', '*local_orig*', '*local_resp*', '*proto*' y '*service*'. Los datos '*src_ip_zeek*', '*dest_ip_zeek*' y '*datetime*', que se transforman aplicando las librerías de Python correspondientes (*datetime* e *ipaddress*) y devuelven valores numéricos asociados con las fechas y direcciones IP originales.

Por último, las tres columnas de etiquetas (binarias, tácticas y técnicas) se asocian a un valor para poder realizar predicciones. Estas transformaciones se recogen en la Tabla 8.2.

Tabla 8.2: Codificación de etiquetas

Columna	Etiqueta	Valor asociado
Binaria	Tráfico Benigno	0
	Tráfico Malicioso	1

Tabla 8.2: Codificación de etiquetas

Columna	Etiqueta	Valor asociado
Tácticas	Tráfico Benigno	0
	<i>Command & Control</i>	1
	<i>Defense Evasion</i>	2
	<i>Discovery</i>	3
	<i>Initial Access</i>	4
	<i>Lateral Movement</i>	5
	<i>Reconnaissance</i>	6
	<i>Resource Development</i>	7
	<i>Collection</i>	8
Técnicas	Tráfico Benigno	0
	T1046	1
	T1071	2
	T1190	3
	T1210	4
	T1548	5
	T1566	6
	T1587	7
	T1589	8
	T1590	9
	T1592	10
	T1595	11
	T1557	12

El desequilibrio en la representación de cada clase restante en el *dataset*, se resuelve mediante el balanceo de datos. En primer lugar, se sub-muestrean las técnicas con más datos, igualando las categorías 0 (Tráfico Benigno), 9 (T1590) y 10 (T1592) con el número de muestras de la categoría 5 (T1548), 3061 filas por cada una. Para seguir, las técnicas menos frecuentes replican aleatoriamente las muestras para obtener más datos entre los que posteriormente aplicar SMOTE. Además, se elimina la clase T1557 (en rojo), que solo contiene un registro y no permite la creación de nuevas muestras. El resultado de estos procesos y el número de registros final por cada técnica se presenta en la Tabla 8.3.

Tabla 8.3: Registros por cada técnica tras el balanceo de datos

Categoría	Original	Sub-Muestreo	Sobre-muestreo aleatorio	Sobre-muestreo SMOTE
0 - Benigno	328026	3061	3061	3061
1 - T1046	10	10	100	200
2 - T1071	6	6	60	140
3 - T1190	3	3	30	126
4 - T1210	9	9	90	180
5 - T1548	3061	3061	3061	3061
6 - T1566	9	9	90	180
7 - T1587	113	113	113	226
8 - T1589	292	292	292	584
9 - T1590	21175	3061	3061	3061
10 - T1592	21400	3061	3061	3061
11 - T1595	37	37	75	150
12 - T1557	1	-	-	-

Para terminar de depurar el conjunto de datos, se reduce el número de columnas, calculando primero la información que aporta cada una en la clasificación multi-clase. Según los resultados mostrados en la Figura 8.3, se mantienen las que aportan más valor: ‘ts’, ‘datetime’, ‘orig_ip_bytes’, ‘resp_ip_bytes’, ‘resp_bytes’, ‘duration’, ‘orig_bytes’, ‘orig_pkts’, ‘history’, ‘resp_pkts’ y ‘conn_state’.

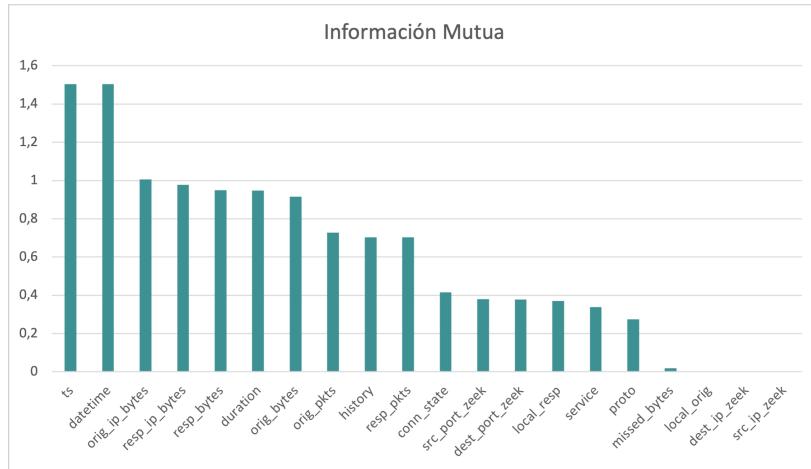


Figura 8.3: Información mutua de las columnas del *dataset*

Finalmente se calcula la matriz de correlación del *dataset*, mostrada en la Figura 8.4, y se eliminan las columnas ‘*datetime*’, ‘*resp_ip_bytes*’ y ‘*resp_pkts*’ por tener un valor mayor a 0.9.

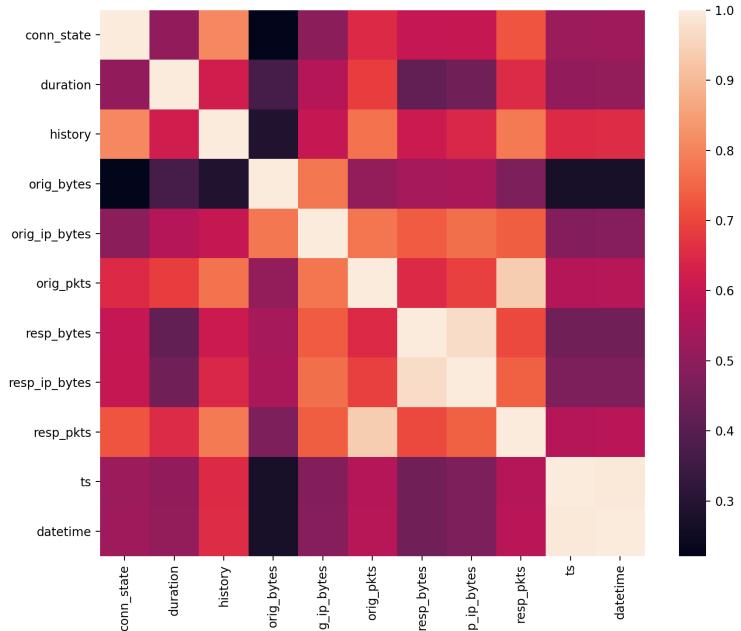


Figura 8.4: Matriz de correlación del conjunto UWF-ZeekDataFall22

El conjunto de datos resultante con el número de muestras que se indica en la última columna de la Tabla 8.3 y las propiedades restantes se divide en un conjunto para entrenamiento y otro para validación, que se guardan en formato *Comma-Separated Values* (CSV), para utilizarlos en el entrenamiento y evaluación de los modelos junto con las columnas de etiquetas binaria, tácticas y técnicas.

8.4.2 Elección de los algoritmos y entrenamiento de los modelos

En las investigaciones analizadas en la Sección 8.2 los modelos supervisados han demostrado tener un buen rendimiento al desempeñar este tipo de tareas, y además se pudo identificar el conjunto de datos etiquetado presentado que permitía su aplicación. Inicialmente se decidió probar el entrenamiento de modelos como *K-Nearest Neighbors* (KNN), que buscan similitudes con las muestras más cercanas para realizar la predicción. No obstante, al haber tenido que balancear el *dataset* con la técnica SMOTE, este tipo de algoritmos perdían mucha exactitud, por lo que el desarrollo se centró en modelos de tipo árbol de decisión. Los elegidos fueron el árbol de decisión, *Random Forest* y XGBoost, entrenados a partir de los mismos datos de entrada obtenidos en el pre-procesamiento y la columna de etiquetas correspondiente a las técnicas ('label_technique').

Los modelos se entrena por separado, almacenando los resultados para poder compararlos y elegir el más adecuado. Las librerías utilizadas serán las correspondientes a cada modelo en la clase Python de Scikit-learn: *DecisionTreeClassifier* [116], *RandomForestClassifier* [120] y *GradientBoostingClassifier* [117], respectivamente.

Los hiperparámetros de cada uno de los algoritmos, que se definieron en la Sección 4.3, han sido optimizados mediante las técnicas *Grid Search* y *Cross-validation* aplicando la función de Scikit-learn [118], que consiste en optimizar los valores de un problema de aprendizaje automático, evitando el error producido tanto por el sobre-ajuste (*overfitting*), que implica que el modelo se adapte demasiado a los datos de entrenamiento; como por el sub-ajuste (*underfitting*), donde el entrenamiento es demasiado genérico, sin tener en cuenta las características de los datos. El procedimiento de validación cruzada se lleva a cabo dividiendo los datos de entrenamiento en grupos aleatorios (normalmente k), de forma que en cada iteración, el modelo utiliza uno de los sub-conjuntos para validar los resultados y el resto para entrenar el algoritmo, probando todos los valores de los hiperparámetros cuyo rango se establece manualmente. Este procedimiento devuelve como configuración del modelo óptima la media de todas las pruebas.

Siguiendo este método, los modelos fueron entrenados con las siguientes configuraciones.

Árbol de decisión

Se optimizan los hiperparámetros de la profundidad del árbol ('*max_depth*'), la aleatoriedad del estimador ('*random_state*') y la función para medir la calidad de una división ('*criterion*'). Los valores obtenidos de la optimización fueron los siguientes, dejando el resto por defecto:

- '*max_depth*' = 10.
- '*random_state*' = 88.
- '*criterion*' = '*entropy*'.

La imagen del árbol obtenido se refleja parcialmente en la Figura 8.5. En cada nodo se muestra la variable sobre la que se está tomando la decisión y la incertidumbre u homogeneidad (entropía) del conjunto de datos. También se indica el número de muestras en total y de cada categoría y, finalmente, la clase más probable que predice ese nodo. Tiene dos salidas, en función de la respuesta a la condición sobre la variable evaluada (*True* o *False*).

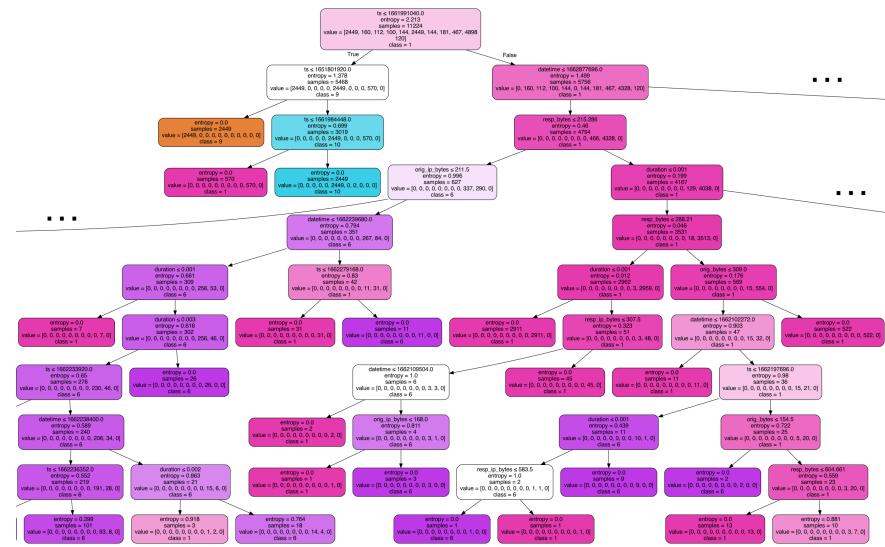


Figura 8.5: Estructura parcial de árbol del modelo árbol de decisión

Random Forest

Para este algoritmo se configuran los hiperparámetros ‘criterion’ y ‘max_depth’ y además el número de árboles en el bosque (‘n_estimators’). En este caso se obtienen los siguientes valores optimizados (el resto se mantienen por defecto):

- ‘max_depth’ = 9.
- ‘n_estimators’ = 114.
- ‘criterion’ = ‘entropy’.

Como este modelo entrena 114 árboles, no es posible mostrarlos todos, por lo que únicamente se incluyen los tres primeros a modo de ejemplo (Figura 8.6 a Figura 8.8). El primero se muestra completo, y los otros dos se muestran parcialmente, con las mismas condiciones que el árbol de decisión posterior.

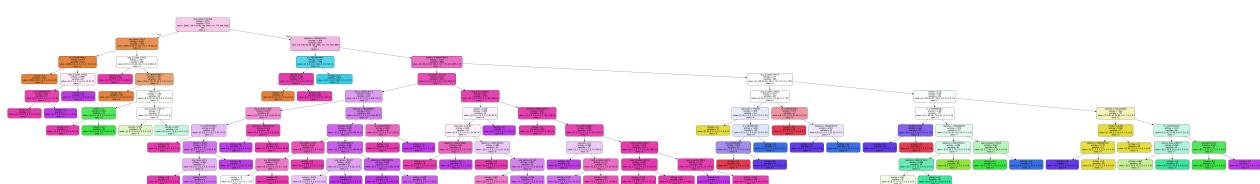


Figura 8.6: Estructura completa de árbol del primer estimador del modelo Random Forest

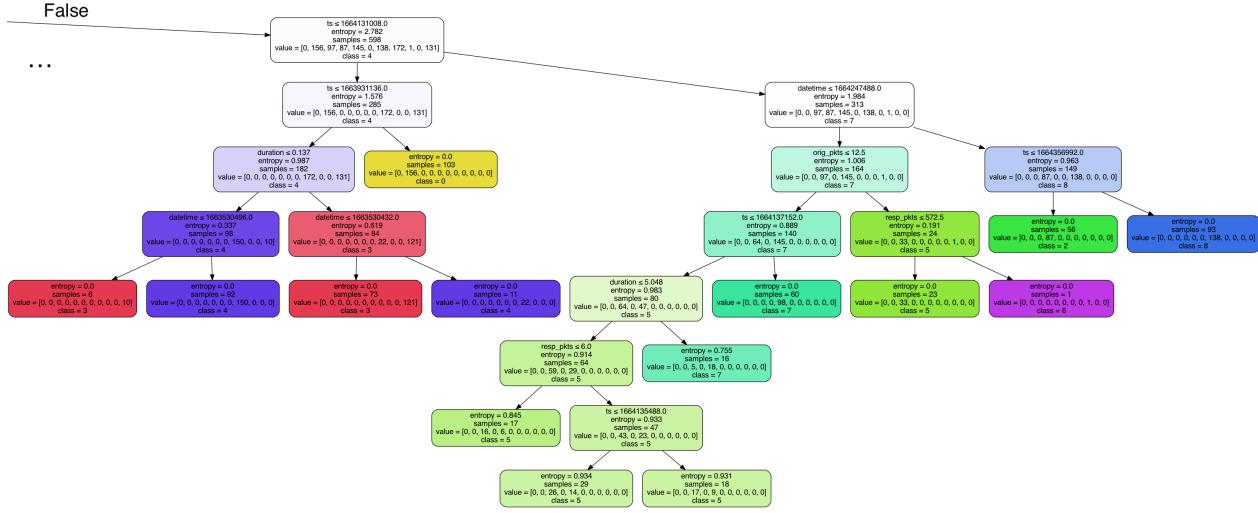


Figura 8.7: Estructura parcial de árbol del segundo estimador del modelo *Random Forest*

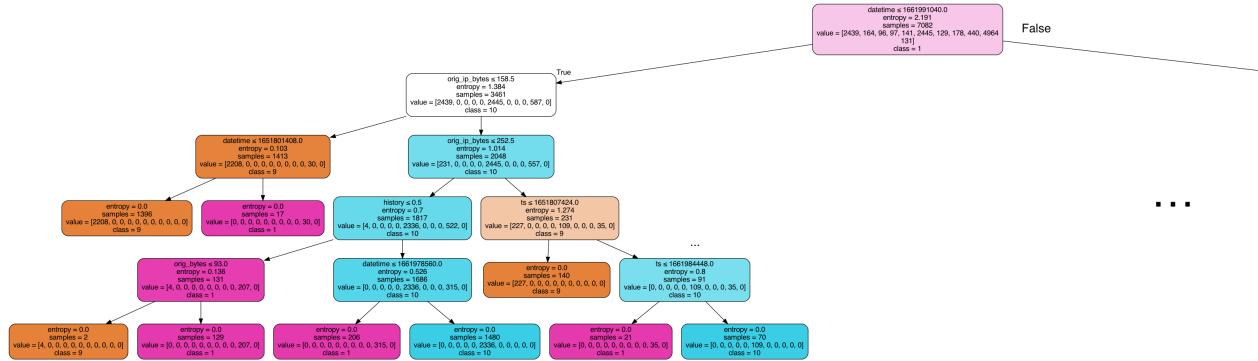


Figura 8.8: Estructura parcial de árbol del tercer estimador del modelo *Random Forest*

XGBoost

Este algoritmo se entrena con hiperparámetros como el número de etapas de refuerzo a realizar ‘*n_estimators*’, y ‘*criterion*’, que ya se mencionó en modelos anteriores, dejando el resto por defecto. Los valores optimizados son los que se indican a continuación:

- ‘*n_estimators*’ = 500
- ‘*criterion*’ = ‘*friedman_mse*’

De nuevo, en este modelo se trabaja con 500 estimadores, por lo que se incluyen dos estructuras como ejemplo (Figura 8.9 y Figura 8.10). En este caso, a pesar de que la estructura es similar a los árboles de decisión y *Random Forest*, los parámetros que aparecen en cada nodo representan

en primer lugar la característica sobre la que se toma la decisión, el error cuadrático medio de Friedman (definido como criterio en el algoritmo) que permite adaptar la optimización de la función de pérdida, el porcentaje de muestras que se evalúan en ese nodo y el valor, que se refiere a la predicción de salida para todas las muestras del nodo.

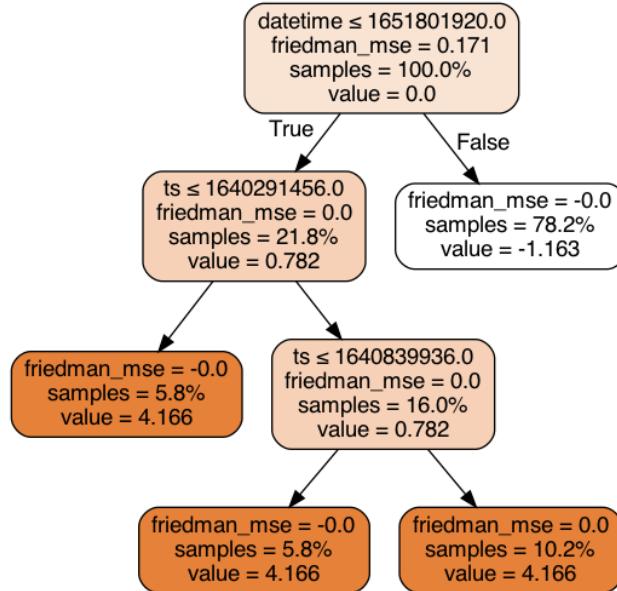


Figura 8.9: Estructura de árbol del primer estimador del modelo XGBoost

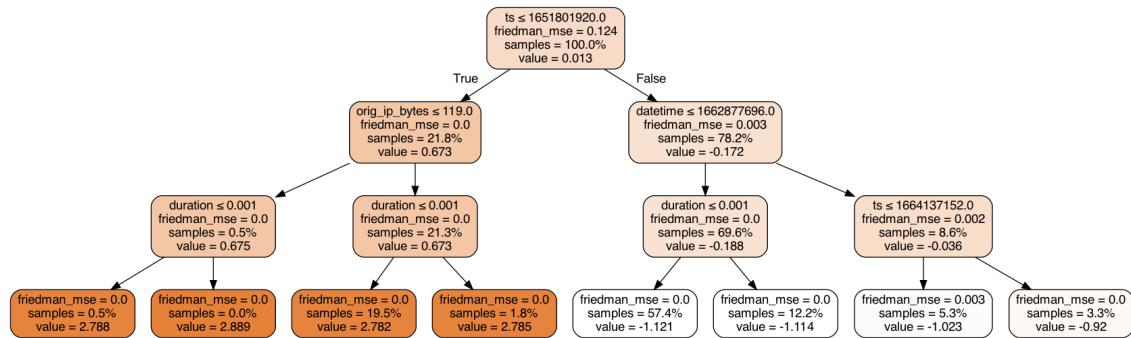


Figura 8.10: Estructura de árbol del segundo estimador del modelo XGBoost

8.4.3 Ontología para la gestión de información relacionada

El último módulo diseñado y desarrollado complementa la caracterización de ciberataques, relacionando las técnicas identificadas en los registros de tráfico con las mitigaciones propuestas por ATT&CK para hacerles frente, y con otra información a partir de los CAPECs en los que aparecen esas técnicas como las debilidades, el impacto o la probabilidad del patrón de ataque u otro tipo de contramedidas menos formales.

Para ello, se diseña una ontología sencilla de gestión de riesgos, en la que estén registradas las mitigaciones y los CAPECs que se han presentado en la Sección 3.4. Así, cuando se reciba un

incidente y se identifique la técnica, automáticamente se relacionarán estos nuevos datos para enriquecer el proceso de gestión de riesgos.

La ontología contiene las siguientes clases (Figura 8.11):

- Incidente: modelados según las propiedades del conjunto de datos. Se relaciona con una técnica, genera amenazas y afecta activos.
- Técnica: almacena las técnicas que puede identificar el modelo entrenado. Como propiedades registra la táctica a la que pertenece y se relaciona con el patrón de ataque en el que se utiliza.
- Patrón de ataque: representa los CAPECs que pueden llevarse a cabo a partir de la técnica identificada, incluyendo la información proporcionada por MITRE (severidad, probabilidad, debilidades, otras mitigaciones). Estos generan una amenaza, que hereda su severidad y probabilidad, y explotan una debilidad asociada a una vulnerabilidad.
- Amenaza: representa el efecto del incidente o ciberataque sobre el sistema, en términos de probabilidad e impacto.
- Contramedida: representa las mitigaciones definidas en ATT&CK, en relación con la técnica frente a la que se recomiendan y, por inferencia, con el incidente del que protegen.
- Vulnerabilidad: modela los CVEs que afectan a uno de los activos de la organización. Incluye valores como CWEs o *Common Vulnerability Scoring System* (CVSS).
- Activo: recurso del que depende la actividad de una organización.

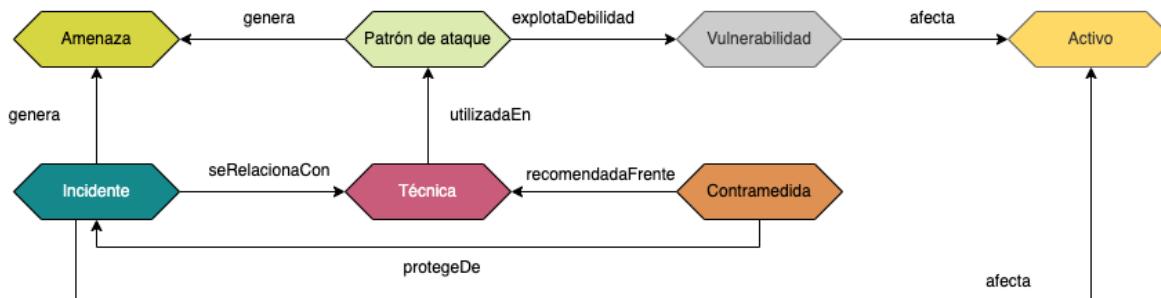


Figura 8.11: Ontología propuesta

La ontología se conecta con la salida del modelo entrenado, recogiendo los registros de tráfico analizados y la etiqueta que se ha asignado según el algoritmo. A partir de ahí, se crea un individuo nuevo que identifica ese incidente y automáticamente la ontología recomienda las contramedidas óptimas para hacerle frente dada la técnica identificada. Además, permite caracterizar el ciberataque asociándolo a patrones de ataque y vulnerabilidades que afecten a los activos del sistema. En la Tabla 8.4 se resumen las mitigaciones asociadas a cada técnica:

Tabla 8.4: Mitigaciones recomendadas por ATT&CK para cada técnica

Técnica	Mitigación
1 - T1046	M1042, M1031, M1030
2 - T1071	M1031
3 - T1190	M1048, M1050, M1030, M1026, M1051, M1016
4 - T1210	M1048, M1042, M1050, M1030, M1026, M1019, M1051, M1016
5 - T1548	M1047, M1038, M1028, M1026, M1022, M1052, M1018
6 - T1566	M1049, M1031, M1021, M1054, M1017
7 - T1587	M1056
8 - T1589	M1056
9 - T1590	M1056
10 - T1592	M1056
11 - T1595	M1056

Por otro lado, en la Tabla 8.5 se recogen los patrones de ataque modelados en CAPEC que tienen relación con alguna de estas técnicas.

Tabla 8.5: CAPECs asociados con cada técnica

Técnica	CAPECs
1 - T1046	CAPEC-300
2 - T1071	-
3 - T1190	-
4 - T1210	-
5 - T1548	CAPEC-114, CAPEC-115, CAPEC-122, CAPEC-233, CAPEC-654
6 - T1566	CAPEC-98, CAPEC-163
7 - T1587	CAPEC-542
8 - T1589	CAPEC-407
9 - T1590	CAPEC-309
10 - T1592	CAPEC-169, CAPEC-541
11 - T1595	CAPEC-169

La gestión de la información extraída a partir de la identificación de técnicas se validará mediante un caso de uso práctico.

8.5 Resultados

Los resultados del sistema se presentan en dos secciones. En primer lugar, los resultados del entrenamiento de los modelos de identificación de técnicas, eligiendo el que proporcione mejor resultado. Finalmente, se presenta un caso de uso para validar la gestión de información extraída.

8.5.1 Modelos de aprendizaje automático

Para comparar los tres modelos entrenados y elegir el que mejor se adapta a las condiciones de este entorno, se evaluarán mediante la exactitud de los modelos, las matrices de confusión

y otras métricas como *F1-Score* o el tiempo de ejecución. Las tácticas T1590 y T1592 se han agrupado debido a su similitud, ya que consisten en recoger información de la red y del dispositivo víctima.

A continuación se presentan métricas individuales obtenidas del entrenamiento y validación de cada uno de los modelos.

Árbol de decisión

En primer lugar, las métricas de exactitud completas del algoritmo de árbol de decisión se muestran en la Tabla 8.6, incluyendo las de entrenamiento y validación para la identificación de técnicas, y la de validación para la detección binaria y la identificación de tácticas.

Tabla 8.6: Exactitud del modelo árbol de decisión y Tiempo de ejecución

Exactitud entrenamiento	Exactitud validación	Exactitud Binaria	Exactitud Tácticas	Tiempo de ejecución
0.9913	0.9868	1.0	0.9939	0.7292 s

En la Tabla 8.7 se presenta el informe de clasificación de este algoritmo en la identificación de técnicas. Como se puede observar, los peores resultados se obtienen en la identificación de T1071 y T1210, con un valor de F1 entre el 70 % y el 80 %.

Tabla 8.7: Informe de clasificación del modelo de árbol de decisión

Etiqueta	Precisión	Recall	F1
0 - Benigno	1.00	1.00	1.00
1 - T1046	0.98	1.00	0.99
2 - T1071	0.65	0.93	0.76
3 - T1190	1.00	1.00	1.00
4 - T1210	0.92	0.61	0.73
5 - T1548	1.00	1.00	1.00
6 - T1566	1.00	1.00	1.00
7 - T1587	1.00	1.00	1.00
8 - T1589	0.86	0.98	0.92
9 - T1590/T1592	1.00	0.98	0.99
10 - T1595	1.00	1.00	1.00
<i>accuracy</i>			0.99
<i>macro avg</i>	0.95	0.96	0.95
<i>weighted avg</i>	0.99	0.99	0.99

En las Figuras 8.12, 8.13 y 8.14 se muestran las matrices de confusión de este modelo. Se puede apreciar que, a pesar del balanceo de datos, el conjunto sigue siendo muy desequilibrado.

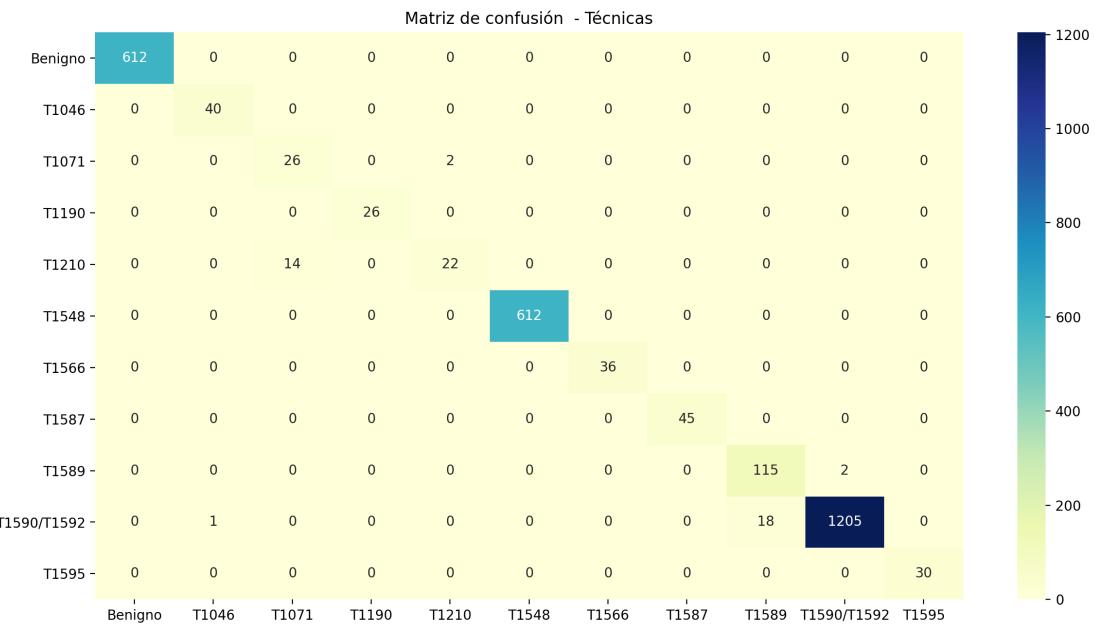


Figura 8.12: Matriz de confusión de la identificación de técnicas para el árbol de decisión

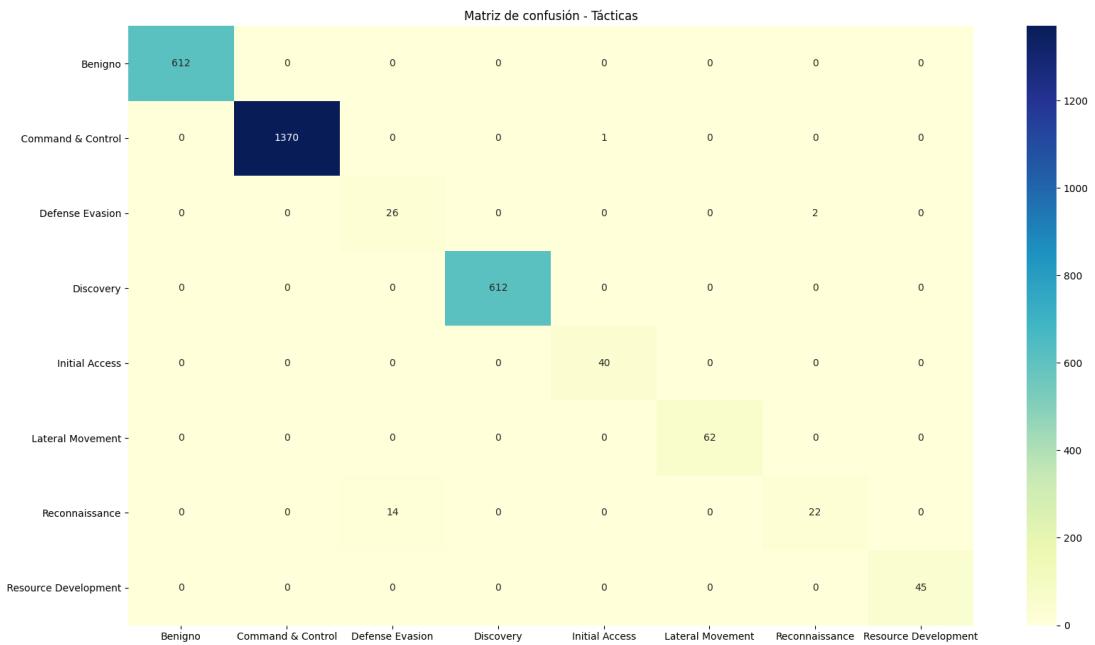


Figura 8.13: Matriz de confusión de la identificación de tácticas para el árbol de decisión

En las matrices se observa que el algoritmo permite la clasificación binaria sin ningún tipo de error. En la identificación de tácticas existe confusión entre las clases de *Defense Evasion* (2) y *Reconnaissance* (6), y entre la clase *Initial Access* (4) y *Command and Control* (1).

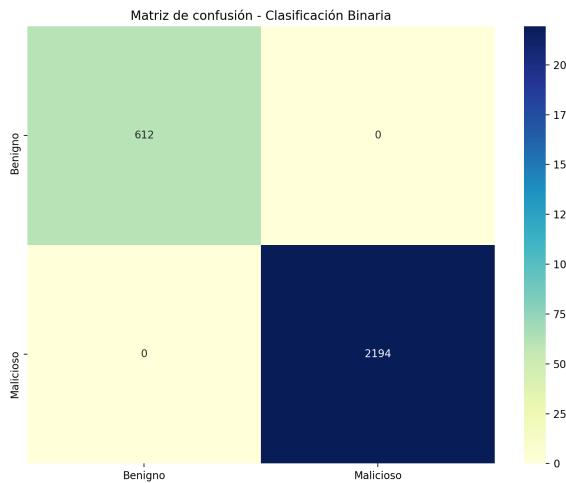


Figura 8.14: Matriz de confusión de la clasificación binaria para el árbol de decisión

Por otro lado, en cuanto a la identificación de técnicas, el modelo equivoca en algunas ocasiones las tácticas T1071 (Etiqueta 2 - *Application Layer Protocol*) y T1210 (Etiqueta 4 - *Exploitation of Remote Services*), y entre T1589 (Etiqueta 8 - *Gather Victim Identity Information*) y T1590/T1592 (Etiqueta 9 - *Gather Victim Network Information/Gather Victim Host Information*). En todos los casos, como se puede deducir del resto de métricas, la confusión existente en estos casos es menor.

Finalmente, la curva ROC de este algoritmo (Figura 8.15) muestra que el entrenamiento no sufre sobre-ajuste ni sub-ajuste a los datos.

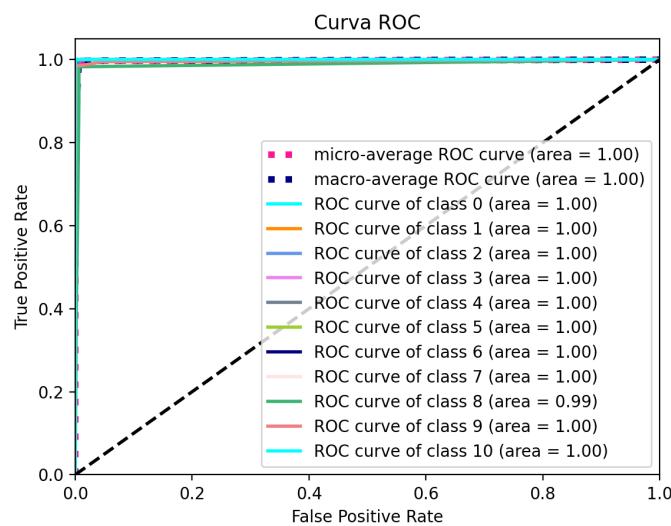


Figura 8.15: Curva ROC del modelo árbol de decisión

Random Forest

De nuevo, las métricas de exactitud del modelo *Random Forest* se muestran en la Tabla 8.8.

Tabla 8.8: Exactitud del modelo *Random Forest* y Tiempo de ejecución

Exactitud entrenamiento	Exactitud validación	Exactitud Binaria	Exactitud Tácticas	Tiempo de ejecución
0.9904	0.9878	1.0	0.9939	1.7820 s

En la Tabla 8.9 se presenta el informe de clasificación de este algoritmo en la identificación de técnicas. Como se puede observar, los resultados son muy similares a los del modelo anterior.

Tabla 8.9: Informe de clasificación del modelo *Random Forest*

Etiqueta	Precisión	Recall	F1
0 - Benigno	1.00	1.00	1.00
1 - T1046	0.98	1.00	0.99
2 - T1071	0.65	0.93	0.76
3 - T1190	1.00	1.00	1.00
4 - T1210	0.92	0.61	0.73
5 - T1548	1.00	1.00	1.00
6 - T1566	1.00	1.00	1.00
7 - T1587	1.00	1.00	1.00
8 - T1589	0.87	0.99	0.93
9 - T1590/T1592	1.00	0.99	0.99
10 - T1595	1.00	1.00	1.00
<i>accuracy</i>			0.99
<i>macro avg</i>	0.95	0.96	0.95
<i>weighted avg</i>	0.99	0.99	0.99

En las Figuras 8.16, 8.17 y 8.18 se muestran las matrices de confusión del algoritmo.

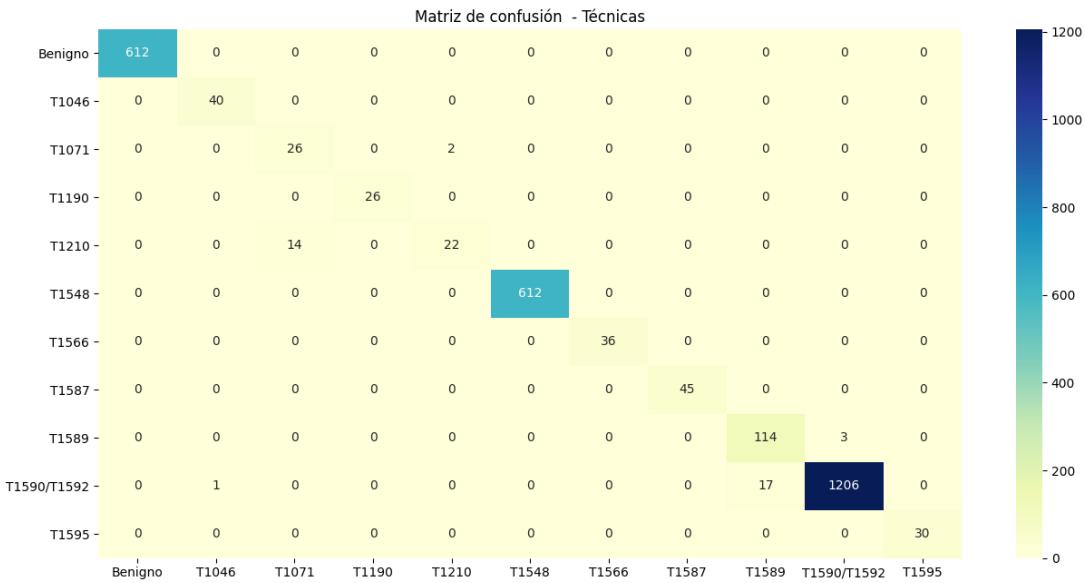


Figura 8.16: Matriz de confusión de la identificación de técnicas para *Random Forest*

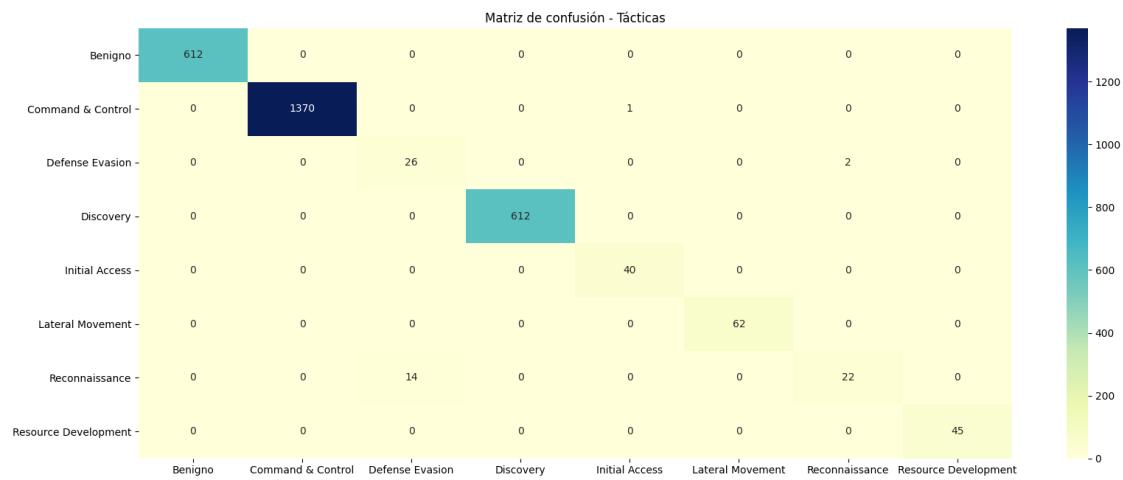


Figura 8.17: Matriz de confusión de la identificación de tácticas para *Random Forest*

El modelo *Random Forest* también permite la clasificación binaria sin ningún tipo de error. En cuanto a la identificación de técnicas, existe la misma confusión que en el algoritmo anterior entre T1071 (Etiqueta 2 - *Application Layer Protocol*) y T1210 (Etiqueta 4 - *Exploitation of Remote Services*), y entre T1589 (Etiqueta 8 - *Gather Victim Identity Information*) y T1590/T1592 (Etiqueta 9 - *Gather Victim Network Information/Gather Victim Host Information*). El modelo también confunde las tácticas de las clases *Defense Evasion* (2) y *Reconnaissance* (6), y entre la clase *Initial Access* (4) y *Command and Control* (1). De igual manera, los valores de muestras mal clasificadas en las matrices indican que las equivocaciones son menores y el modelo funciona correctamente.

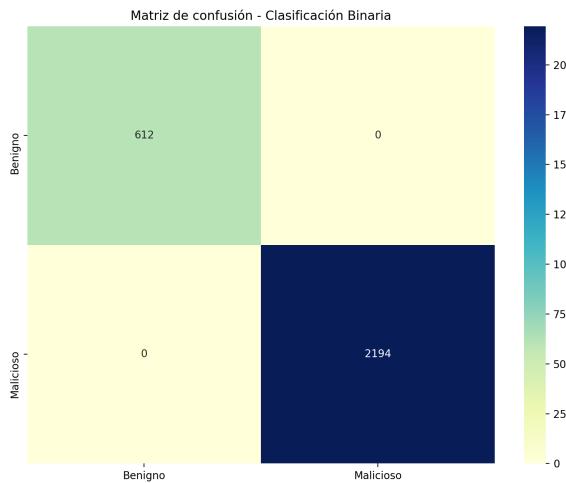


Figura 8.18: Matriz de confusión de la clasificación binaria para *Random Forest*

Finalmente, la curva ROC de este algoritmo (Figura 8.19) muestra que el entrenamiento es correcto.

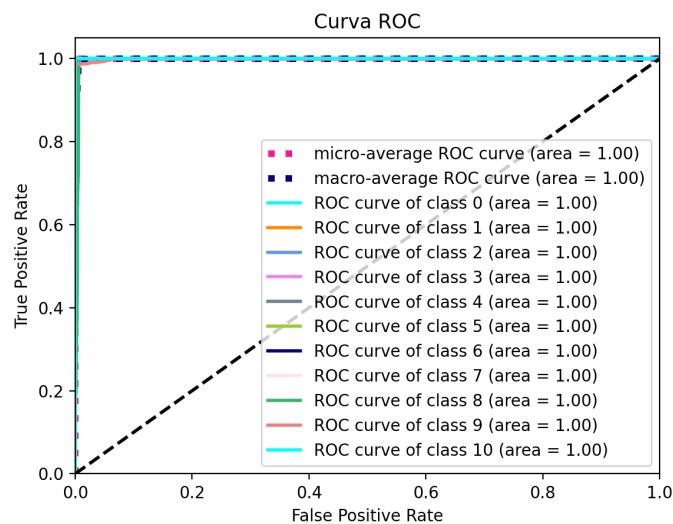


Figura 8.19: Curva ROC del modelo *Random Forest*

XGBoost

Las métricas de exactitud del modelo XGBoost se muestran en la Tabla 8.10.

En la Tabla 8.11 se presenta el informe de clasificación del modelo en la identificación de técnicas. En este caso los resultados también son muy similares a los anteriores.

En las Figuras 8.20, 8.21 y 8.22 se muestran las matrices de confusión del algoritmo XGBoost.

Tabla 8.10: Exactitud del modelo XGBoost y Tiempo de ejecución

Exactitud entrenamiento	Exactitud validación	Exactitud Binaria	Exactitud Tácticas	Tiempo de ejecución
0.9926	0.9839	1.0	0.9939	51.1392 s

Tabla 8.11: Informe de clasificación del modelo XGBoost

	Etiqueta	Precisión	Recall	F1
0 - Benigno	1.00	1.00	1.00	1.00
1 - T1046	0.98	1.00	0.99	
2 - T1071	0.65	0.93	0.76	
3 - T1190	1.00	1.00	1.00	
4 - T1210	0.92	0.61	0.73	
5 - T1548	1.00	1.00	1.00	
6 - T1566	1.00	1.00	1.00	
7 - T1587	1.00	1.00	1.00	
8 - T1589	0.86	0.91	0.88	
9 - T1590/T1592	0.99	0.99	0.99	
10 - T1595	1.00	1.00	1.00	
<i>accuracy</i>				0.98
<i>macro avg</i>				0.95
<i>weighted avg</i>				0.99
				0.98

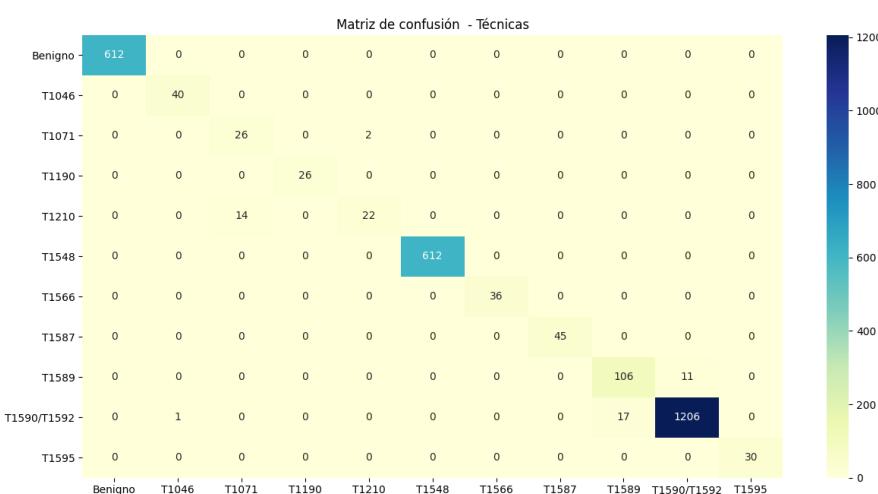


Figura 8.20: Matriz de confusión de la identificación de técnicas para XGBoost

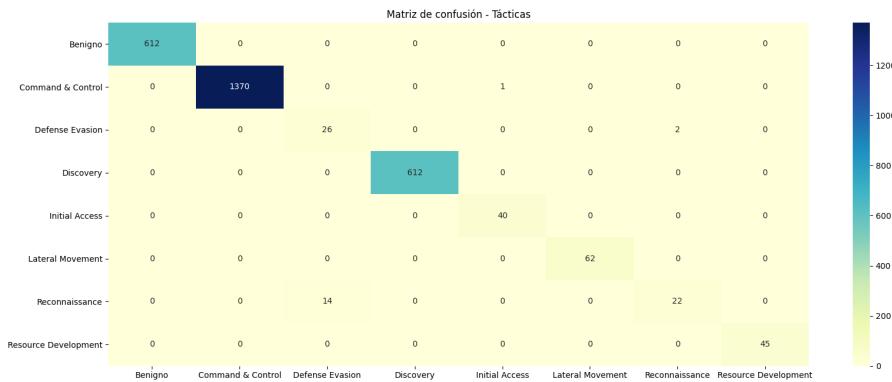


Figura 8.21: Matriz de confusión de la identificación de tácticas para XGBoost

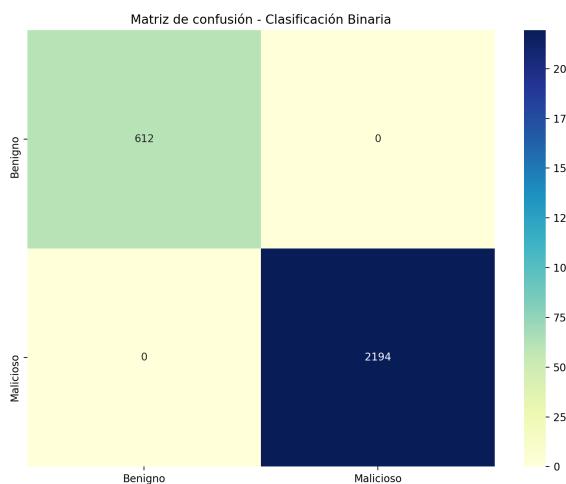
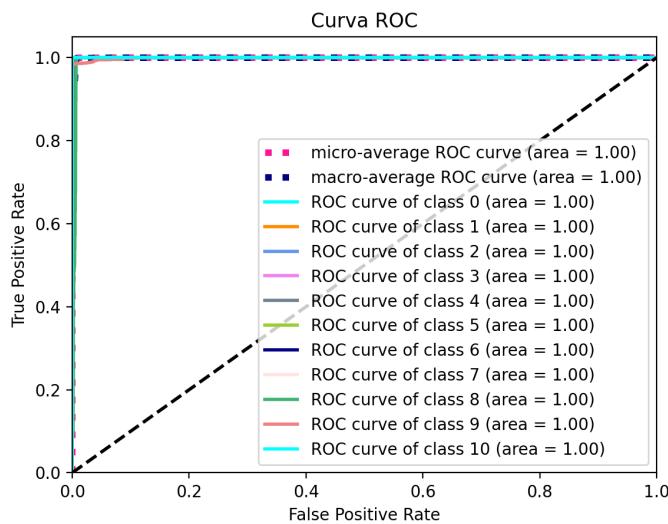


Figura 8.22: Matriz de confusión de la clasificación binaria para XGBoost

XGBoost también permite la clasificación binaria sin ningún tipo de error. En la identificación de tácticas existe confusión entre las clases de *Defense Evasion* (2) y *Reconnaissance* (6), y entre la clase *Initial Access* (4) y *Command and Control* (1). Por otro lado, en cuanto a la identificación de técnicas, existe confusión entre T1071 (Etiqueta 2 - *Application Layer Protocol*) y T1210 (Etiqueta 4 - *Exploitation of Remote Services*), y entre T1589 (Etiqueta 8 - *Gather Victim Identity Information*) y T1590/T1592 (Etiqueta 9 - *Gather Victim Network Information/Gather Victim Host Information*). En todos los casos, como se puede deducir del resto de métricas, la confusión existente en estos casos es menor.

Finalmente, la curva ROC de este algoritmo (Figura 8.23) muestra que el entrenamiento no sufre sobre-ajuste ni sub-ajuste a los datos.

**Figura 8.23:** Curva ROC del modelo XGBoost

8.5.2 Caso de uso

Para validar el sistema en conjunto, específicamente la gestión de la información relacionada con la técnica identificada en el registro de tráfico, se plantea un caso de uso donde se transmiten varios incidentes. Los modelos identifican las técnicas asociadas y se almacenan los datos en la ontología, para validar tanto la recomendación de las contramedidas planteadas por MITRE como la relación con determinadas vulnerabilidades y la creación de amenazas.

Se utilizará un conjunto de datos que representa la técnica T1046, otro registro para la técnica T1210 y finalmente otro para T1595.

Tras ejecutar el gestor del sistema de propuesta de contramedidas, se esperan los siguientes resultados (Tablas 8.12 - 8.14) de acuerdo con lo que se recoge en los catálogos ATT&CK y CAPEC.

Tabla 8.12: Asociación de información en el caso de uso para T1046

Incidente	Mitigaciones	CAPEC
Incidente 1	M1030 - <i>Network Segmentation</i> M1042 - <i>Disable or Remove Feature or Program</i> M1031 - <i>Network Intrusion Prevention</i>	CAPEC-300 : <i>Port Scanning</i> (Severidad: Baja, CWE-200)

Tabla 8.13: Asociación de información en el caso de uso para T1210

Incidente	Mitigaciones	CAPEC
Incidente 2	M1030 - <i>Network Segmentation</i>	
	M1042 - <i>Disable or Remove Feature or Program</i>	
	M1048 - <i>Application Isolation and Sandboxing</i>	Ninguna
	M1026 - <i>Privileged Account Management</i>	
	M1051 - <i>Update Software</i>	
	M1050 - <i>Exploit Protection</i>	
	M1016 - <i>Vulnerability Scanning</i>	
	M1019 - <i>Threat Intelligence Program</i>	

Tabla 8.14: Asociación de información en el caso de uso para T1595

Incidente	Mitigaciones	CAPEC
Incidente 3	M1056 - <i>Pre-compromise</i>	CAPEC-169 : <i>Footprint</i> (Severidad: Muy Baja, Probabilidad: Alta, CWE-200, contramedidas: actualización periódica del sistema y contraseñas)

Los individuos de la ontología resultantes, por tanto, deberían quedar asociados como se muestra en la Figura 8.24, teniendo en cuenta que el Incidente 2 no tiene relacionado ningún patrón de ataque y, por tanto, no se puede inferir ninguna amenaza de la técnica identificada.

En la herramienta *Protégé* se valida este proceso, mostrado en las siguientes capturas. En la primera (Figura 8.25), se muestra el estado inicial de la ontología, que únicamente contiene los individuos de las mitigaciones, los CAPECs relacionados con el caso de uso y un ejemplo de vulnerabilidad que aprovecha la debilidad CWE-200 y otra que no. Además, existe un activo afectado por la primera vulnerabilidad.

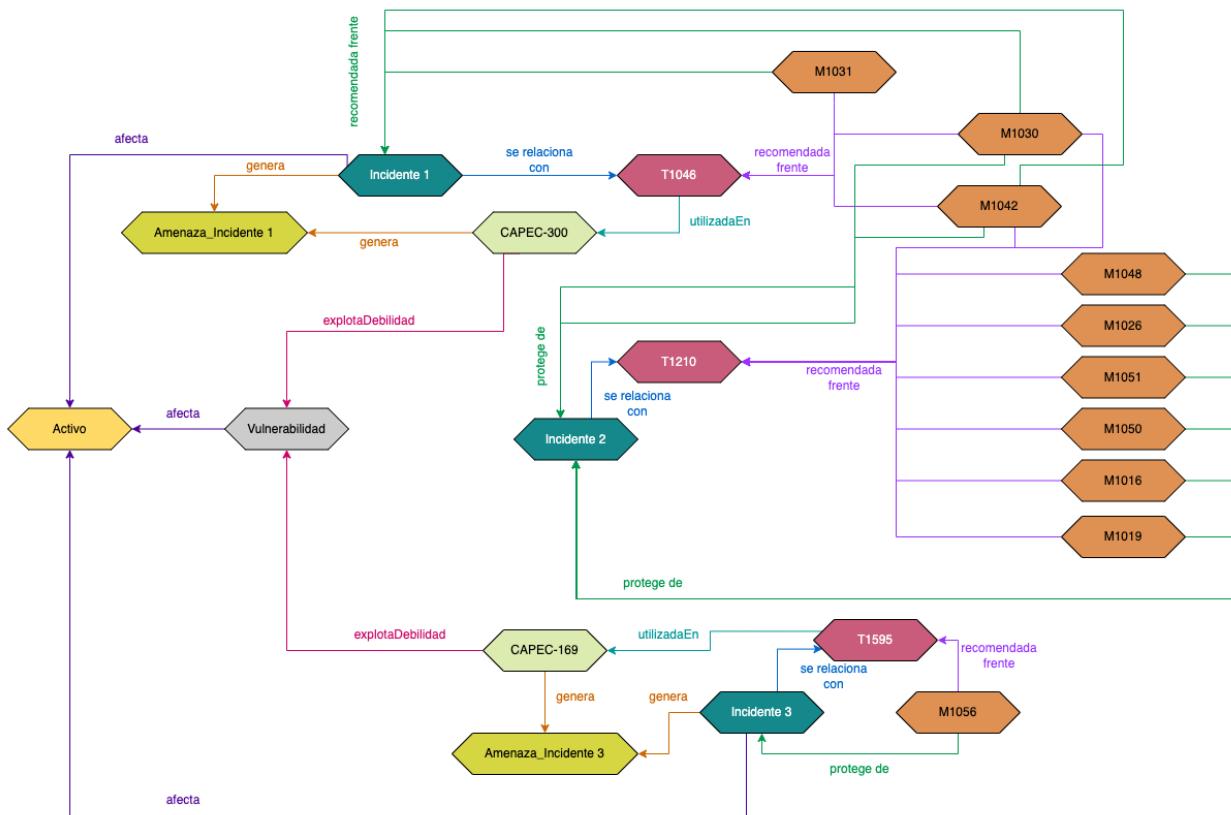


Figura 8.24: Relaciones entre los individuos del caso de uso



Figura 8.25: Estado inicial de la ontología para el caso de uso

El siguiente paso será registrar los incidentes, que ya han atravesado el modelo entrenado y tienen una técnica asignada. En la Figura 8.26 se muestran los tres incidentes y los datos iniciales de uno de ellos.

La creación de estos individuos desencadena la inferencia de conocimiento en la ontología. La relación entre los incidentes y las técnicas debe realizar automáticamente los siguientes pasos:

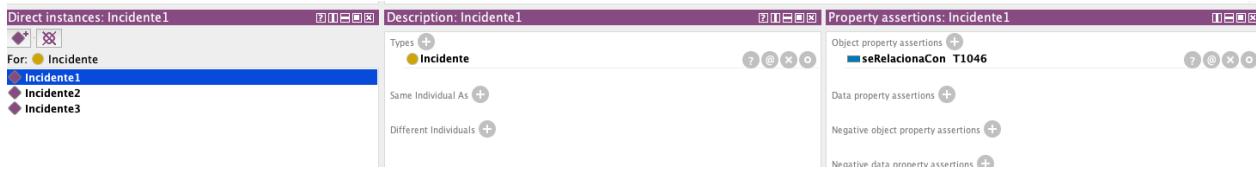


Figura 8.26: Creación de los incidentes en la ontología del caso de uso

- Verificar las contramedidas recomendadas para la técnica asociada, y relacionarlas con el incidente (Figura 8.27).
- Comprobar si existe algún patrón de ataque registrado en el que se aplique la técnica, y crear una amenaza que se asocia con el incidente original y que hereda información del patrón de ataque (Figura 8.28).
- Si existe el patrón de ataque, y está relacionado con una vulnerabilidad que afecte a un activo, el incidente original debe afectar a este activo. Además, el incidente recibe información de otras contramedidas según la recomendación CAPEC (Figura 8.29).



Figura 8.27: Mitigaciones asociadas con cada incidente

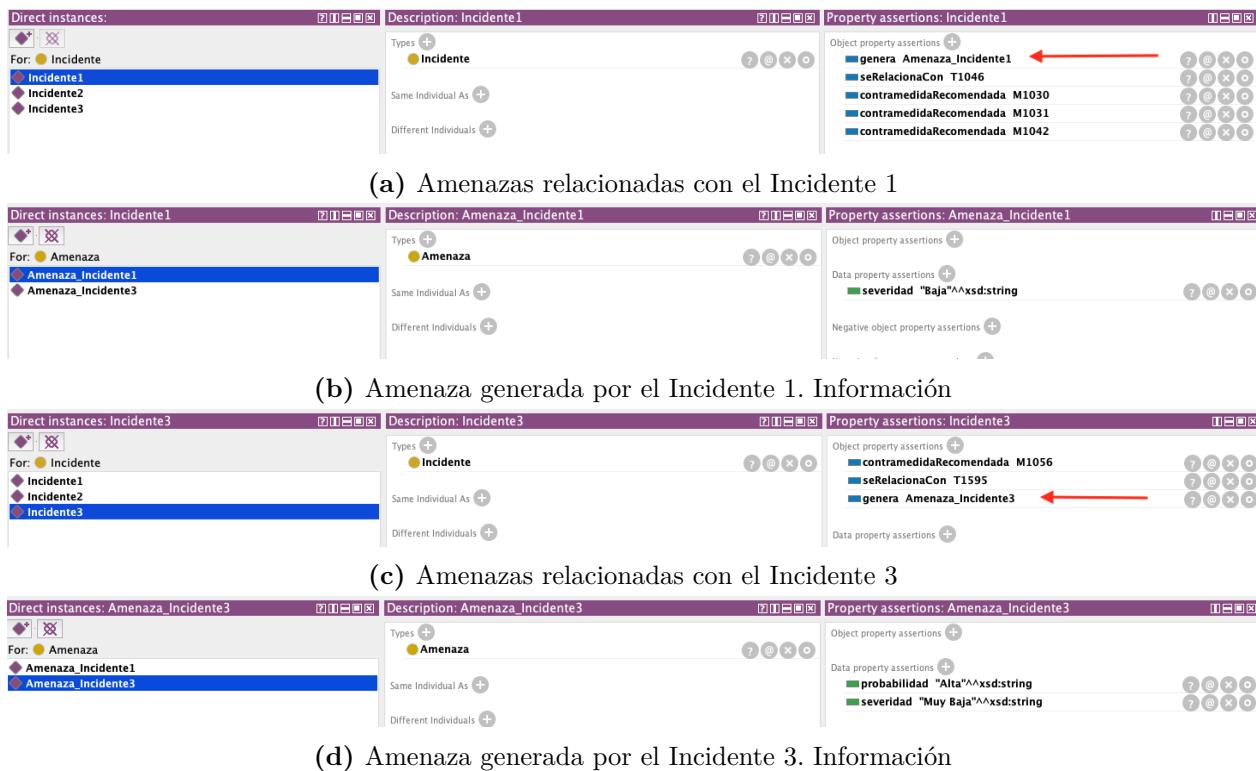


Figura 8.28: Amenazas generadas por cada incidente



Figura 8.29: Activo relacionado con los incidentes

En la última captura (Figura 8.29) se presenta la situación final de la ontología.

- El Incidente 1 tiene tres mitigaciones asociadas que permiten reducir el riesgo que supone. Además, al estar relacionado con un patrón de ataque, genera una amenaza de severidad baja y afecta al activo expuesto a la debilidad que explota este patrón.
- El Incidente 2, que no está relacionado con ningún patrón de ataque, únicamente se relaciona con las ocho contramedidas recomendadas por ATT&CK.
- El caso del Incidente 3, similar al Incidente 1, se asocia con una mitigación, afecta al activo expuesto a la debilidad y genera una amenaza con probabilidad alta y severidad muy baja que le aporta recomendaciones de mitigación opcionales.

8.6 Conclusiones

Dada la importancia de la protección dinámica de los sistemas frente a amenazas y riesgos, la caracterización de técnicas y tácticas permite complementar la información obtenida a partir de estudios expertos y ofrecer las protecciones óptimas en cada situación, completando la caracterización de los ciberataques. Este proceso implica una comprensión más profunda de los elementos de la matriz MITRE ATT&CK para poder definir reglas que identifiquen estos comportamientos. Sin embargo, estos criterios para la detección no pueden abarcar las técnicas más complejas, donde la mejor oportunidad se encuentra en las ventajas de la IA y el aprendizaje automático.

Entre los modelos entrenados, cuyos resultados se presentan en la Tabla 8.15, todos los algoritmos tienen resultados de exactitud superiores al 98 %, por lo que los tres podrían llevar a cabo la identificación de las técnicas en registros de tráfico. Sin embargo, atendiendo a los tiempos de ejecución, el modelo XGBoost se descarta, eligiendo como el más adecuado para el conjunto de datos de entrenamiento el modelo de árbol de decisión.

Tabla 8.15: Comparación de los modelos entrenados

Modelo	Exactitud entrenamiento	Exactitud validación	Métrica F1	Tiempo de ejecución
Árbol de decisión	0.9913	0.9868	0.9869	0.7292 s
<i>Random Forest</i>	0.9904	0.9878	0.9879	1.7820 s
XGBoost	0.9926	0.9839	0.9840	51.1392 s

Los resultados presentados contribuyen al desarrollo de una estrategia integral que permita abordar la respuesta frente a amenazas a raíz de la caracterización de las técnicas utilizadas en el ciberataque con soluciones innovadoras.

La investigación presentada aprovecha la eficacia del aprendizaje automático aplicándolo en la identificación de técnicas MITRE ATT&CK, permitiendo recomendar contramedidas

específicas y extrayendo información relevante para la gestión de riesgos. Este enfoque aborda la detección de incidentes de ciberseguridad y la respuesta proactiva ante ellos gracias a la capacidad de razonamiento e inferencia de conocimiento de las ontologías basándose en la información propuesta por MITRE.

Los resultados obtenidos permiten mejorar en algunos aspectos los trabajos realizados anteriormente, como se recoge en la Tabla 8.16.

Tabla 8.16: Comparación de trabajos previos con esta propuesta

Publicación	N. Tácticas	N. Técnicas	Métrica
[61]	5	5	No se indica
[126, 127]	6	13	AUC: 0.868
[18]	3	-	Exactitud multiclas: 0.9999
[16]	2	-	<i>Discovery</i> : 0.9991 <i>Reconnaissance</i> : 0.994
[21]	8	-	Exactitud : 0.999936
Esta Propuesta	7	11	Exactitud Técnicas: 0.9868 Exactitud Tácticas: 0.9939 AUC: 0.999

En cuanto a la detección de técnicas, esta propuesta mejora los resultados de área bajo la curva del trabajo presentado en [126, 127], a pesar de tener la capacidad de detectar 4 técnicas menos, y supera al resto de trabajos relacionados. Además, este desarrollo permite la detección de tácticas, superando en número a casi todos los trabajos previos. En el caso de [21], a pesar de detectar una táctica más, no incluye la detección de técnicas, lo que supone una mejora con respecto a ese trabajo. Este sistema podría enriquecerse con un sistema de identificación mediante reglas, para localizar algunas de las técnicas que son indetectables para este modelo.

Además de la cantidad de tácticas y técnicas que la propuesta planteada es capaz de identificar, se pueden comprobar los resultados de este sistema directamente con los obtenidos en [126] utilizando el enfoque mediante reglas, que se recoge en la Tabla 8.17 extraída del mismo documento, donde indican si es posible entrenar un clasificador para detectar la técnica utilizando su conjunto de datos.

Tabla 8.17: Resultados de la investigación [126] aplicando esta propuesta

Táctica	Técnica	Detección mediante reglas	Detección mediante IA
<i>Reconnaissance</i>	T1590	✓	✗ → ✓
<i>Credential Access</i>	T1557	✓	✗ → ✗
<i>Discovery</i>	T1124	✓	✓
	T1135	✓	✓
<i>Lateral Movement</i>	T1021	✓	✗
	T1550	✓	✗
	T1563	✓	✗
	T1570	✓	✗
<i>Command and Control</i>	T1071	✓	✓ → ✓
	T1090	✓	✗
	T1105	✓	✓
	T1571	✓	✓ → ✗
<i>Execution</i>	T1053	✓	✗

En esta tabla, se observan las 13 técnicas identificadas mediante reglas. En cuatro de ellas coinciden ambas propuestas, difiriendo el resultado en dos de ellas. La T1590, que no se podía identificar con modelos de aprendizaje automático, con este desarrollo y el *dataset* de entrenamiento sí se ha podido identificar. Al contrario ocurre con la T1571, que por la limitación del conjunto de datos inicial, no ha podido ser identificada ya que contenía datos erróneos. En la T1071 ambas investigaciones coinciden en que es viable la detección mediante algoritmos, mientras que la T1557, al encontrarse únicamente una muestra en el conjunto de datos de entrenamiento, no pudo ser identificada. Además, la caracterización presentada en este capítulo, aunque no es capaz de identificar las tácticas *Credential Access* o *Execution*, sí incluye *Defense Evasion*, *Initial Access* y *Resource Development*, por lo que el abanico de técnicas identificadas es muy variado.

Como parte del entorno de conciencia cibersituacional, este desarrollo representa la base de la caracterización de ciberataques de la que surge la Hipótesis 2 de esta Tesis Doctoral. La validación de la misma se llevará a cabo en secciones futuras, partiendo de la investigación presentada en este capítulo, ya que la información extraída y el comportamiento de la ontología pueden conducir a mejores resultados en la reducción de riesgos y la recomendación de contramedidas adaptadas.

Capítulo 9

Propuesta de una metodología basada en ontologías para la interoperabilidad de marcos dinámicos de gestión de riesgos

A lo largo de este capítulo se define un módulo que permite llevar a cabo de forma dinámica los procesos de una metodología de gestión de riesgos para completar el funcionamiento del entorno de conciencia cibersituacional que se muestra en la arquitectura global (Figura 6.1). El módulo se resalta sobre esta arquitectura en la Figura 9.1.

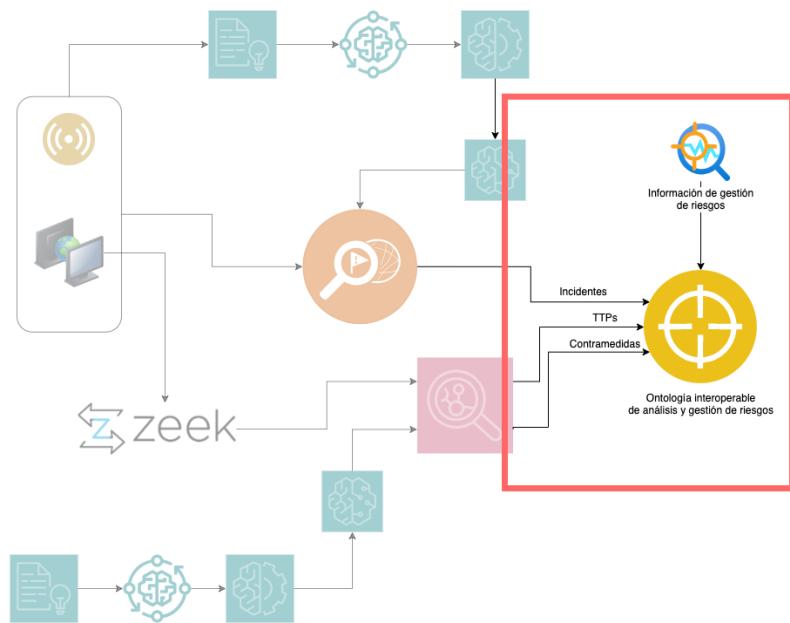


Figura 9.1: Módulo de gestión de riesgos en la arquitectura global propuesta

La metodología presentada aquí es más completa y por tanto mejora la que se utilizó en el capítulo anterior para la recomendación de contramedidas y extracción de información relacionada con las técnicas que identifica el módulo anterior, completando la caracterización de los ciberataques a través de la inferencia de conocimiento. Además, permite la interoperabilidad con otros marcos de gestión.

El estado del arte analizado en el Capítulo 5 es la base para el desarrollo presentado en éste. Tras analizar la literatura relacionada en la Sección 9.2, se identifica la problemática que se intenta abordar mediante la solución propuesta y desarrollada en la Sección 9.3. Finalmente la propuesta se valida mediante unos casos de uso (Sección 9.4), extrayendo las conclusiones que se presentan en la Sección 9.5.

9.1 Introducción

La caracterización de los ciberataques no termina en la identificación de las técnicas que conlleva un incidente, sino que debe completarse con la definición de los riesgos que suponen estas amenazas y la respuesta ante ellos.

Actualmente, cualquier organización se expone a un gran volumen de ataques de ciberseguridad. Las consecuencias de estas amenazas se materializan en grandes pérdidas financieras, daño reputacional o incluso la interrupción del negocio. La conciencia cibersituacional continua del estado del sistema, especialmente el nivel de riesgo al que se expone permite reaccionar de forma dinámica a ellos y proteger información sensible del ciberataque [142].

Distintas agencias de ciberseguridad junto a grupos de expertos en la materia, en su intento de definir un sistema para responder a esta necesidad, han diseñado distintos marcos que definen los procesos a la hora de analizar y gestionar riesgos en la seguridad de la información. Actualmente, éstos se aplican en la mayoría de organizaciones a nivel mundial, con el objetivo de estimar el nivel de riesgo que afecta al sistema y en algunos casos proponer contramedidas para manejarlos, definiendo el nivel de riesgo residual. Algunas metodologías comparten nomenclatura y propósito, pero los procedimientos y algunos conceptos por lo general difieren. Esto provoca que el intercambio y comparación de información entre cada marco sea una tarea tediosa, especialmente en el cálculo del nivel de riesgo.

La necesidad de unificación e interoperabilidad en el dominio de la ciberseguridad que se ha identificado en esta Tesis Doctoral como uno de los focos de investigación principales en la actualidad, es una de las motivaciones de esta propuesta. Ya existen desarrollos para unificar los términos del dominio y los conceptos que soportan la conciencia cibersituacional a través de la aplicación de una ontología y reglas de inferencia [135].

El ámbito en el que se enmarca la Tesis Doctoral se caracteriza por la información heterogénea, donde propuestas como la citada anteriormente demuestran la capacidad de las ontologías para gestionar este tipo de datos, dándoles sentido y extrayendo información mediante reglas en lenguajes como SPIN o SWRL.

La novedad de esta propuesta es proporcionar un marco único para la gestión de riesgos que acepte como entrada distintas metodologías, llevando a cabo la evaluación de los riesgos

mediante una ontología interoperable.

En este campo es esencial la anticipación a los ciberataques que puedan surgir, identificando y gestionando los riesgos a los que se exponen las organizaciones y aplicando medidas para reducir su impacto. Para ello, el sistema desarrollado recoge información para llevar a cabo estos procesos, mapea conceptos de las metodologías de entrada en un marco interoperable, identificando amenazas para conocer el nivel de riesgo, que es vital para responder a contingencias y establecer medidas preventivas.

Las metodologías propuestas y analizadas en el Capítulo 5 se encuentran entre las que tienen mayor tasa de adopción, centrando el diseño en marcos europeos como EBIOS, MAGERIT, MONARC, ITSRM o CRAMM. Esta propuesta es un primer paso hacia la consecución de esta interoperabilidad, ya que permite transformar datos de estos marcos en un cálculo común, facilitando así comparaciones o intercambio de información. Así, se obtiene un modelo dinámico de gestión de riesgos compatible con los estándares más aplicados. Este tipo de desarrollos se validan mediante casos de uso y el desarrollo de un prototipo.

9.2 Trabajos relacionados

Se han identificado varios trabajos previos centrados en el uso y la creación de modelos ontológicos en el dominio de la ciberseguridad, como las vulnerabilidades. Sin embargo, existen pocas investigaciones centradas en el modelado de amenazas y la seguridad operacional, incluyendo la inteligencia de amenazas (*Cyber Threat Intelligence*, CTI), lo que demuestra el reto que supone su desarrollo.

En la búsqueda de modelos de información, primero se analizaron los relacionados con CTI. De esta forma, en [11] se introduce un enfoque ontológico para analizar el diseño de un sistema y la creación de patrones de seguridad utilizando fuentes de amenazas y contramedidas para el modelado automático de amenazas en sistemas basados en la nube. Los autores en [77] describen un modelo CTI para analizar y compartir información sobre amenazas de forma eficaz utilizando ontologías. También se encontró una ontología de vulnerabilidades en [134] diseñada para un sistema de alerta de vulnerabilidades y contramedidas utilizando reglas SWRL. En [54] se presenta un sistema que reúne y gestiona conocimiento CTI de fuentes abiertas (OSCTI). Esta información se extrae mediante un modelo de aprendizaje profundo, a partir de lo que genera un modelo de información basado en ontologías y un grafo de conocimiento.

En los últimos años se han multiplicado las investigaciones basadas en la gestión de riesgos, especialmente utilizando ontologías [22, 23, 5, 13, 143]. Su fortaleza en este tipo de trabajos, permitiendo la interoperabilidad entre diferentes dominios, se pone de manifiesto en [123, 124], donde los autores aplican ontologías para identificar amenazas y riesgos utilizando un razonador e información de fuentes externas, como bases de datos de vulnerabilidades. Además, validan los resultados con un caso de uso. Sin embargo, destacan la imposibilidad de crear nuevas instancias por inferencia. Esta limitación, en determinados contextos, como el de esta Tesis Doctoral, es una debilidad de las reglas SWRL, ya que la gestión de riesgos no sólo pretende modificar atributos o crear relaciones, sino que los incidentes recibidos deben

desencadenar la creación automática de amenazas y el cálculo de riesgos.

No obstante, también existen ejemplos previos de gestión de riesgos donde las ontologías no son la tecnología central de la propuesta como [56], donde los autores introducen un método de evaluación de riesgos y gestión de respuestas validado a través de un caso de uso, destacando la aplicación de generadores de grafos de ataque para definir posibles escenarios de riesgo. Una metodología similar se utiliza en [137] para evaluar los riesgos utilizando el modelado de vectores de ataque. Los autores de [9] aplican el aprendizaje automático (redes neuronales bayesianas) para predecir la gravedad de los riesgos futuros basándose en evaluaciones anteriores. En [64] se aplica una metodología de identificación de riesgos de ciberseguridad en sub-estaciones digitales para reconocer ciberataques potenciales, evaluando los riesgos y sus impactos y definiendo planes de mitigación según MITRE ATT&CK.

Por otra parte, el gran número de propuestas basadas en ontologías permite considerarlas como una de las tecnologías preferidas para esta tarea, gracias a su capacidad de razonamiento. Se utilizan para intercambiar información, como puede observarse en [73], centrado en la gestión del riesgo operacional mediante ontologías. Facilitan el intercambio de información a través de una organización y áreas de negocio heterogéneas, apoyando la toma de decisiones con la ayuda de reglas de razonamiento. En [57], los autores definen una ontología de ciberseguridad para ayudar en los procedimientos de recopilación de información y evaluación de riesgos en sistemas complejos.

Así, los autores de [46] presentan una técnica basada en una ontología para el cálculo de la propagación del riesgo, midiendo el impacto en diferentes activos. El riesgo se expresa en función del impacto en la tríada CIA (confidencialidad, integridad, disponibilidad). Asimismo, en [115] se presenta un marco para evaluar los riesgos de seguridad en plataformas de computación en la nube, donde el riesgo se define como una combinación de la probabilidad y el impacto de un evento. Los riesgos e impactos se categorizan siguiendo diferentes objetivos de seguridad definidos, donde las escalas de impacto se basan en el modelo FIPS. En [94] se presenta una ontología especializada en la evaluación de riesgos, que utiliza un método basado en amenazas para evaluar la peligrosidad de los ataques APT desde diferentes perspectivas (riesgos tácticos, riesgos de activos y respuestas a los ataques APT), y los autores de [82] proponen una metodología para evaluar el riesgo en los sistemas de información, incluyendo como novedad vulnerabilidades conocidas y desconocidas, que no suelen tenerse en cuenta. Centrado en el marco MITRE ATT&CK, la investigación de [8] trata de abordar los riesgos producidos por APTs mediante el estudio de técnicas y tácticas de los adversarios. A través de la caracterización de los actores de amenazas en ontologías y grafos de ataque, los autores llevan a cabo una evaluación del riesgo, estimando la probabilidad del ataque. En [90] se propone una metodología de análisis de riesgos considerando el marco MITRE ATT&CK, que permite explicar el impacto de una acción sobre otra y la correlación de acciones con fuentes de datos, defensas, configuraciones y otras contramedidas utilizadas para seguridad centrado en un caso de uso concreto.

La necesidad de unificación de conceptos e interoperabilidad presentada en [135] también es la motivación en trabajos más recientes como [26], destacando la urgencia de utilizar terminología estandarizada en ciberseguridad, especialmente en la gestión de riesgos.

Sin embargo, no existen muchos ejemplos de ontologías basadas en normas de gestión de riesgos, como es el caso de esta propuesta. Los autores de [80] muestran un ejemplo de ontología para modelar datos CTI y reglas de razonamiento para la supervisión de riesgos en tiempo real basada en las definiciones de riesgo de la norma ISO 27005 y en las dependencias entre éstos, amenazas, vulnerabilidades y activos. Destacan la importancia de utilizar el razonamiento semántico en un sistema de toma de decisiones de ciberseguridad. Además, en [33] se presentan modelos de riesgo gráficos para desarrollar algoritmos de evaluación basados en la ISO 31000. También, en [141] los autores presentan una solución para gestionar la norma ISO 27005:2011 para ayudar a entender los conceptos utilizando una ontología y un escenario de uso como ejemplo. El modelo captura los conceptos centrales y las relaciones de la metodología de gestión de riesgos.

Con un enfoque diferente, los autores de [12] desarrollan una ontología para seleccionar la metodología más adecuada para la gestión de riesgos en función de las características de la organización. Se basa en las normas ISO 31000, ISO 31010 e ISO 73 sobre identificación, análisis y evaluación de riesgos para ayudar en la elección de las técnicas de gestión de riesgos. Asimismo, en [140] se desarrolla una extensión de MAGERIT para incluir conceptos difusos. Por último, como trabajo preliminar, es necesario mencionar [51], donde los autores presentan una definición formalizada de los riesgos de ciberseguridad relacionada con el documento [40] de metodologías de gestión de riesgos de ENISA, y donde proponen como trabajo futuro un marco para gestionar la implementación del análisis de riesgos según varios estándares.

Como primera solución al problema planteado en algunos de los trabajos de investigación anteriores sobre creación de instancias y procesamiento en tiempo real, se ha validado el uso de reglas SPIN en un sistema dinámico en la investigación presentada en [114]. La conclusión que se extrae del análisis de estos trabajos de investigación es, por tanto, que la necesidad de interoperabilidad existente no está totalmente cubierta y es un foco de trabajo en curso. La mayoría de las ontologías de ciberseguridad analizadas son específicas de un dominio, por lo que su reutilización implica importantes modificaciones para obtener la generalización necesaria para la interoperabilidad. Además, como se ha mostrado, existen algunas iniciativas que están tratando de unificar el dominio de la ciberseguridad en sus diferentes aspectos, pero en cuanto al análisis y gestión de riesgos, se encuentran trabajos teóricos como los mencionados anteriormente, pero no existen sistemas que permitan esta interoperabilidad.

9.3 Diseño de la solución

Para afrontar el problema de la interoperabilidad entre los marcos de gestión de riesgos, y dado que las ontologías permiten llevar a cabo estas tareas y han demostrado en trabajos anteriores su eficacia, se propone el desarrollo de un modelo de información con reglas de comportamiento basado en las características analizadas en el Capítulo 5. Por lo tanto, la metodología propuesta que trata de completar este análisis será basada en activos y realiza una evaluación de riesgos tanto cuantitativos como cualitativos.

El sistema propuesto recibe como entrada el estudio llevado a cabo por expertos en cada metodología, extrayendo los activos propios de cada organización, estudiando las vulnerabilidades asociadas, información relativa al sistema, contramedidas disponibles y un análisis de los

posibles escenarios de riesgo. Además, los incidentes que se registran en el sistema en tiempo real, identificados mediante un IDS y caracterizados previamente, también se almacenan para analizar su efecto sobre el sistema.

Estos datos se recogen en un gestor encargado de cargar los catálogos procedentes de una de las metodologías interoperables con el sistema, poblando las clases definidas en la ontología. También gestionará la ontología y las reglas de inferencia que permitirán estimar los valores de probabilidad e impacto de cada amenaza. Asimismo, en función de los escenarios de riesgo definidos, en el gestor se realiza el cálculo del nivel de riesgo y la propagación del efecto de estos incidentes sobre los activos del sistema, así como la selección de las contramedidas más adecuadas para protegerlos, calculando el riesgo residual para cada amenaza identificada.

ENISA ha realizado trabajos previos a este desarrollo, proponiendo un método [3] para la traducción entre otras metodologías de gestión de riesgos y la propuesta por ella como metodología estándar (ITSRM), con el objetivo de compartir información y compararlas. Siguiendo las directrices marcadas en dicho documento, el sistema conjunto tiene como objetivo recopilar información para realizar el análisis y gestión de riesgos de forma automática, efectuando cálculos en una escala estandarizada, pero también tiene la capacidad de normalizar valores de otros informes de gestión de riesgos realizados por otras metodologías, con el fin de comparar resultados. Además, se incluye en el sistema una lógica de apoyo a la toma de decisiones para la elección de las contramedidas óptimas.

Por tanto, partiendo de la base teórica que proporciona ENISA se obtiene un sistema funcional e interoperable que permite la gestión y evaluación dinámica de riesgos en tiempo real. En la Figura 9.2 se muestra un resumen de los procedimientos y la arquitectura. Las amenazas previas, que se incluyen en el gestor a través de una flecha de color rojo indica que es una entrada opcional. Las flechas azules indican relaciones entre los distintos catálogos, mientras que el resto de flechas, en negro, representan los procesos propios de la gestión de riesgos (carga de la información, inferencia para generar amenazas a partir de los incidentes, cálculo del nivel de riesgo y propuesta de contramedidas).

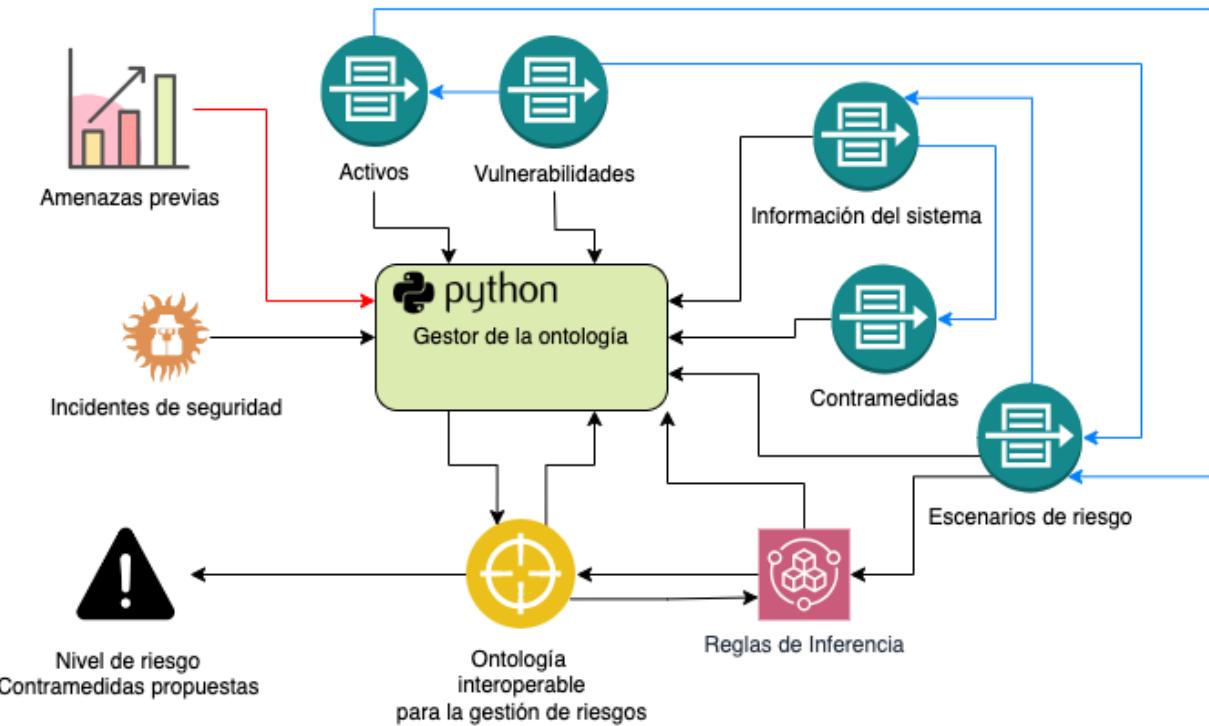


Figura 9.2: Escenario de la propuesta

El sistema que gestiona la ontología también se encarga de poblarla a través de catálogos en formato JSON, con la estructura presentada en el Anexo A, teniendo en cuenta las características principales de cada entrada según las metodologías:

- **Información del sistema**: Este archivo contiene datos generales sobre el sistema analizado.
- **Activos**: Contiene una lista de los activos primarios del sistema y sus activos de soporte.
- **Vulnerabilidades**: En este documento se listan las vulnerabilidades que afectan a los activos anteriores.
- **Amenazas previas**: Este archivo, que es opcional, contiene información de amenazas previas que hayan ocurrido en el sistema generadas por otros procesos de gestión, si el administrador quiere importarla. El sistema acepta definiciones de amenazas de otras metodologías (palabras en otros idiomas o distintas denominaciones).
- **Escenarios de Riesgo**: Aquí se almacena el resultado principal del análisis experto, donde se identifican posibles activos -primarios o de soporte- atacados, las vulnerabilidades a las que se exponen y el tipo y propiedades de las amenazas que se generan en caso de que en el sistema se reciba un incidente que encaje con el escenario.
- **Incidentes de seguridad**: La información generada por los sistemas de detección, incluyendo la caracterización de las técnicas de los ciberataques identificados.

- **Contramedidas:** Catálogo de mitigaciones y otras respuestas disponibles en el entorno, incluyendo información para seleccionar la óptima y estimar el riesgo residual según el efecto de mitigación.

9.3.1 Diseño de la ontología

Teniendo en cuenta los resultados reflejados en la literatura previa, para resolver la problemática de la interoperabilidad entre metodologías de gestión de riesgos se propone el diseño de una ontología capaz de seguir los procesos definidos en las distintas metodologías. Es adaptable, modelando a nivel genérico una metodología dinámica de gestión de riesgos que pueda aplicarse a cualquier metodología de las presentadas en el Capítulo 5. A pesar de que ya existen varias ontologías centradas en la gestión de riesgos, identificadas en la Sección 9.2, se ha decidido diseñarla desde cero en base a la guía de ENISA, en lugar de reutilizar una debido a la necesidad de interoperabilidad, ya que las existentes previamente son, por lo general, muy específicas e incluyen conceptos y relaciones propios del escenario para el que fueron diseñadas.

En la Figura 9.3 se representa la estructura de clases y relaciones identificadas para describir una metodología de gestión de riesgos. Las líneas del mismo color representan el mismo tipo de relación entre clases, y las líneas discontinuas reflejan que las clases destino son sub-clases de las origen.

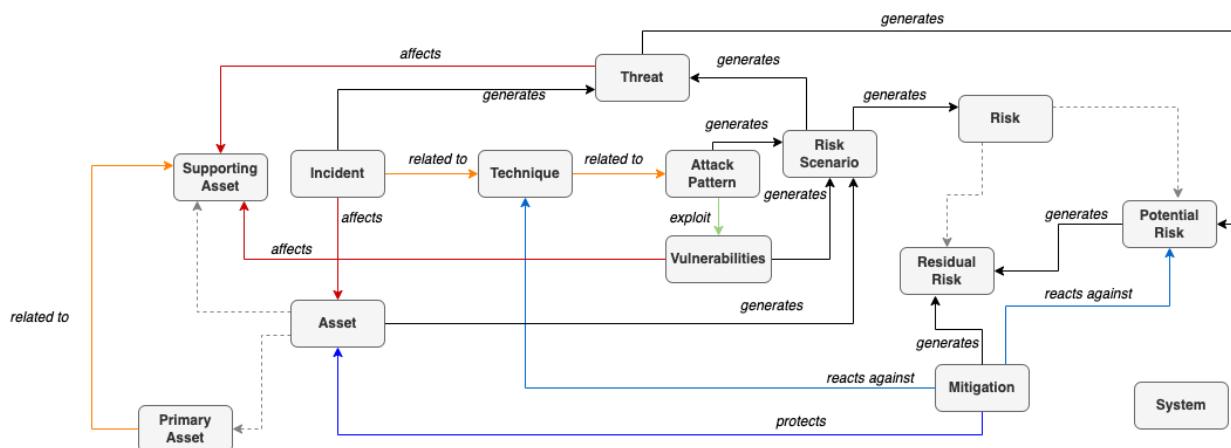


Figura 9.3: Esquema de la ontología

Las propiedades de las clases descritas a continuación permiten la creación de individuos a partir de las distintas entradas del sistema. La elección de atributos y relaciones se basa en trabajos previos y contribuciones propias:

- **System:** Esta clase representa propiedades del sistema global, como la metodología de gestión de riesgos original o el nivel de aceptación del riesgo. En relación con los cálculos, entre los atributos se encuentra el nivel de riesgo global calculado tanto con el método interoperable como con la metodología original, y finalmente el nivel de riesgo residual tras aplicar las contramedidas óptimas para cada caso.

- *Asset*: En esta clase, que modela el catálogo de activos, se definen propiedades comunes a las subclases: tipo de activo, dimensiones de seguridad (confidencialidad, integridad y disponibilidad) y descripción. Se relaciona con la clase escenario de riesgo (*Risk Scenario*) mediante la propiedad “*generates*”.
 - *Primary Asset*: Esta sub-clase define activos primarios, cruciales para la actividad de la organización. Los valores de la propiedad tipo de activo están restringidos a ‘Information and Data’ y ‘Processes, Functions and Services’. Se relacionan con al menos un activo de soporte (*Supporting Asset*) a través de la propiedad “*related_to*”.
 - *Supporting Asset*: Sub-clase que representa activos de soporte, utilizados en la gestión de los primeros. Los tipos en este caso se limitan a ‘Hardware, Devices and Equipment’, ‘Infrastructure’, ‘Location and Utilities’, ‘Personnel’ y ‘Software and Applications’. En este caso, las relaciones con otras clases serán las inversas a las que se describen más adelante, ya que en la Figura 9.3 son únicamente el destino de las flechas.
- *Incident*: Esta clase representa los incidentes de seguridad y ciberataques detectados y caracterizados en el sistema. Las propiedades mínimas serán: identificador, descripción y fecha. En cuanto a los enlaces con otras clases, se relaciona con una técnica (*Technique*) mediante la relación “*related_to*”, con al menos un activo (*Asset*) con la relación “*affects*” y con amenazas (*Threat*) a través de la relación “*generates*”.
- *Technique*: Representa las técnicas definidas en el marco MITRE ATT&CK identificadas en registros de tráfico maliciosos. Como propiedades, identifica la táctica a la que pertenece. Además, se relacionan con un patrón de ataque (*Attack Pattern*) utilizando la propiedad “*related_to*”.
- *Attack Pattern*: Aquí se representan los CAPECs relacionados con las técnicas identificadas. Como propiedades recogen la severidad, probabilidad, debilidades y posibles mitigaciones o recomendaciones. Los individuos de esta clase se relacionan con escenarios de riesgo (*Risk Scenario*) a través de la propiedad “*generates*” y mediante la propiedad “*exploit*” con vulnerabilidades (*Vulnerabilities*).
- *Vulnerabilities*: Las vulnerabilidades se definen mediante atributos como un identificador (CVE), tipo, descripción, debilidades (CWE), *Common Platform Enumeration* (CPE) y CVSS. Esta clase se relaciona con los activos de soporte (*Supporting Asset*) mediante la propiedad “*affect*” y con los escenarios de riesgo (*Risk Scenario*) utilizando la relación “*generates*”
- *Threats*: Las amenazas se definen a través de atributos como descripción, origen y dos conjuntos de propiedades que definen a qué dimensión de la seguridad afectan y en qué medida. Interactúan con las clases de activos de soporte (*Supporting Asset*) utilizando la relación “*affects*” y con el riesgo potencial (*Potential Risk*) con la propiedad “*generates*”. Tiene un conjunto muy amplio de sub-clases, agrupadas en ‘Errors and unintentional failure’, ‘Industrial’, ‘Natural’, ‘Willful attacks’ y ‘Service-related threats’, cada uno con las sub-clases identificadas por ENISA en [3].

- *Risk Scenario*: Esta clase define las amenazas generadas a raíz de un incidente o ciberataque caracterizado sobre un activo concreto. Se caracteriza por las propiedades identificador, impacto y probabilidad; y las relaciones con las clases amenaza (*Threat*) y riesgo (*Risk*) a través de “*generates*”.
- *Risk*: La propiedad principal de esta clase es el nivel de riesgo, donde se almacena el cálculo a través de la metodología de gestión original. Como sub-clases se definen los riesgos potenciales (*Potential Risk*) y residuales (*Residual Risk*), que se relacionan a través de la propiedad “*generates*” con origen en el primero.
- *Mitigation*: Esta clase almacena las contramedidas disponibles en el escenario, definidas por un identificador, descripción, tipo, palabras clave, coste y esfuerzo de despliegue, complejidad de instalación, tiempo en estar activo y disponible, impacto, efectividad y factor de mitigación. Además tiene propiedades para indicar si, por alguna razón, la respuesta no está disponible; y para indicar si ha sido seleccionada y propuesta como óptima en algún caso. En relación con otras clases, influye sobre los riesgos residuales (*Residual Risk*) con la propiedad “*generates*”, con las clases riesgo potencial (*Potential Risk*) y técnica (*Technique*) mediante la propiedad “*reacts against*” y con los activos (*Assets*) a través de “*protects*”.

9.3.2 Diseño del gestor de la ontología

Este módulo trata de automatizar el cálculo de riesgo y la evaluación utilizando la metodología original y utilizando el método interoperable, que se explicarán más adelante. De esta forma, cuando se detecta un incidente que afecta a un activo de la organización, se generan automáticamente amenazas, nuevas relaciones y cambios en los valores de riesgo y evaluación de activos. Además, se recomiendan contramedidas de los catálogos y de las propuestas por MITRE ATT&CK según el nivel de riesgo calculado, y el criterio de aceptación de riesgo.

El objetivo es conseguir una ontología aplicable en casos de uso prácticos y entornos de conciencia cibersituacional como el de esta Tesis Doctoral, donde la gestión de riesgos se base en alguno de los marcos analizados en el Capítulo 5. Para ello, se desarrolla un sistema de gestión y se analiza el comportamiento del modelo, creando un conjunto de restricciones y reglas de razonamiento. Previamente, un grupo de expertos debe analizar el entorno y mediante un estudio, definir los activos, vulnerabilidades y contramedidas y componer los escenarios de riesgo, generando los catálogos definidos anteriormente.

En la Figura 9.4 se presentan los pasos principales que sigue la metodología presentada para llevar a cabo la gestión dinámica de riesgos, aceptando información de otras metodologías y buscando contribuir a la interoperabilidad de estos marcos.

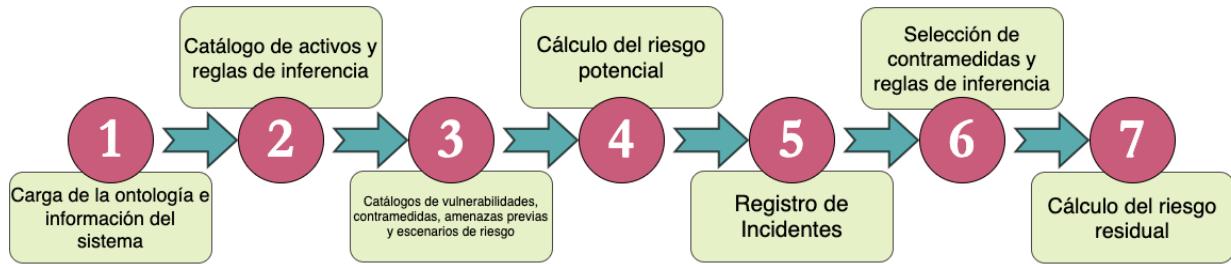


Figura 9.4: Secuencia de procesos del sistema

En el primer paso del gestor se carga la ontología y se incluye la información de contexto: identificador del sistema, la metodología de análisis de riesgos anterior y el nivel de aceptación del sistema, a partir del cual no se tratan los riesgos. A continuación se carga el catálogo de activos. Esto implica la creación de los individuos y las relaciones entre activos primarios y de soporte que se definen en el archivo de entrada. Después se ejecuta una regla de inferencia en formato SPIN que actualiza las valoraciones de los activos primarios para las dimensiones de seguridad en función de los activos de soporte con los que tienen relación. Se calcula la media de los valores de cada dimensión entre todos los *Supporting Assets* que se relacionan con el activo primario, y estos valores se asocian con este activo.

```

DELETE {
?pa uri:confidentiality ?c.
?pa uri:integrity ?i.
?pa uri:availability ?a.
} INSERT {
?pa uri:confidentiality ?nc.
?pa uri:integrity ?ni.
?pa uri:availability ?na.
} WHERE {
?asset a uri:Supporting_Asset.
?asset uri:confidentiality ?c.
?asset uri:integrity ?i.
?asset uri:availability ?a.
?pa a uri:Primary_Asset.
?pa o:related_to ?asset.
BIND(AVG(?c) as ?nc)
BIND(AVG(?i) as ?ni)
BIND(AVG(?a) as ?na)
} group by ?pa
    
```

Regla de inferencia: Actualización de la valoración de los activos primarios

El tercer paso es la carga del catálogo de vulnerabilidades y el de mitigaciones, creando nuevas instancias relacionadas con activos que representan las contramedidas disponibles y

qué activo protegen y las vulnerabilidades identificadas y a cuál afectan. Además, si existen amenazas previas almacenadas, se pueden importar en el sistema para conocer el nivel de riesgo en un momento determinado o el efecto de las contramedidas en una situación concreta. Este es el primer proceso interoperable, ya que los datos definidos en el catálogo utilizan los términos de la metodología de origen y se asocian a los tipos genéricos de la ontología tras una agrupación de amenazas similares de todos los marcos con los que se trabaja. Por ejemplo, si la metodología original es MAGERIT, las amenazas utilizarán denominaciones en castellano correspondientes con el catálogo (por ejemplo, ‘Corte de suministro’) y se convertirán al término más similar en ITSRM (en este caso, ‘*Power Interruption*’). Estas amenazas también afectarán a los activos de soporte, reduciendo el valor de las dimensiones de seguridad según la consecuencia y, por tanto, este impacto se extenderá a los activos primarios de la misma forma que en el paso 1. Aquí también se crean las instancias de escenarios de riesgo, con información suficiente para generar amenazas si se registra un incidente que encaje con los campos del escenario. Esto se genera a partir de un estudio del entorno y de las probabilidades y consecuencias de cada amenaza.

En este punto de la metodología ya se han introducido en el sistema todos los datos anteriores y permanece a la espera de que se identifique y caracterice algún nuevo incidente. Cuando se registra uno nuevo, el sistema genera una instancia asociada. Las técnicas, vulnerabilidades, patrones de ataque y activos con los que se relaciona se comparan con los escenarios de riesgo existentes. Si hay una coincidencia se define una amenaza con los datos del escenario.

A continuación, se lleva a cabo el primer cálculo de riesgo, siguiendo las indicaciones definidas según la metodología original. Se generan instancias de riesgo en el sistema asociadas a las amenazas anteriores y se les asigna un impacto y una probabilidad, a partir de los que se obtiene el nivel de riesgo según sus definiciones y según un estándar que permita compararlos (en este caso, ITSRM). Con cada nuevo incidente los niveles de riesgo se actualizan. Se debe tener en cuenta que los escenarios de riesgo podrían estar definidos según la metodología original y deben traducirse para llevar a cabo todo el proceso.

Las escalas de riesgo de cada metodología son las siguientes:

- EBIOS tiene tres niveles de riesgo (*Low (L)*, *Medium (M)*, *High (H)*) que se alcanzan con las combinaciones definidas en el siguiente mapa de calor, donde el impacto y la probabilidad toman valores enteros del 1 al 4 (Figura 9.5).

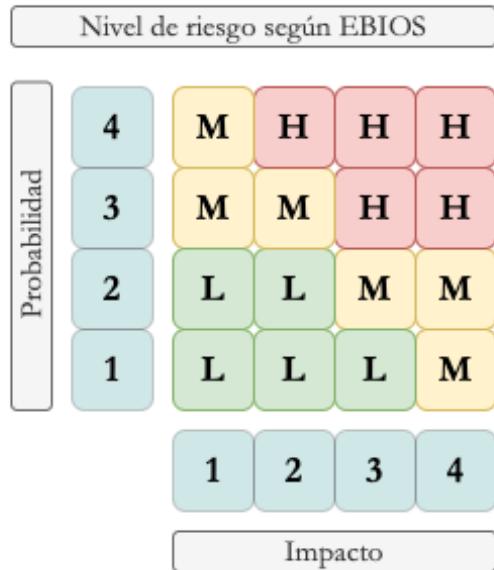


Figura 9.5: Mapa de calor para el cálculo del riesgo en EBIOS

- En el caso de MAGERIT los niveles son Muy Bajo (MB), Bajo (B), Medio (M), Alto (A) y Muy Alto (MA), obtenidos a partir del mapa presentado en la Figura 9.6.

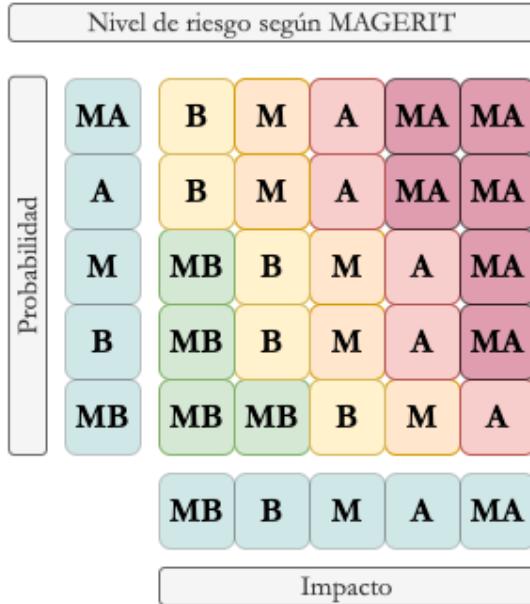


Figura 9.6: Mapa de calor para el cálculo del riesgo en MAGERIT

- MONARC, sin embargo, define tres niveles de riesgo (*Low (L)*, *Medium (M)*, *High (H)*) a partir de la probabilidad y el impacto, que toman valores continuos entre 0 y 4. La escala de riesgo se define como se muestra en la Figura 9.7.

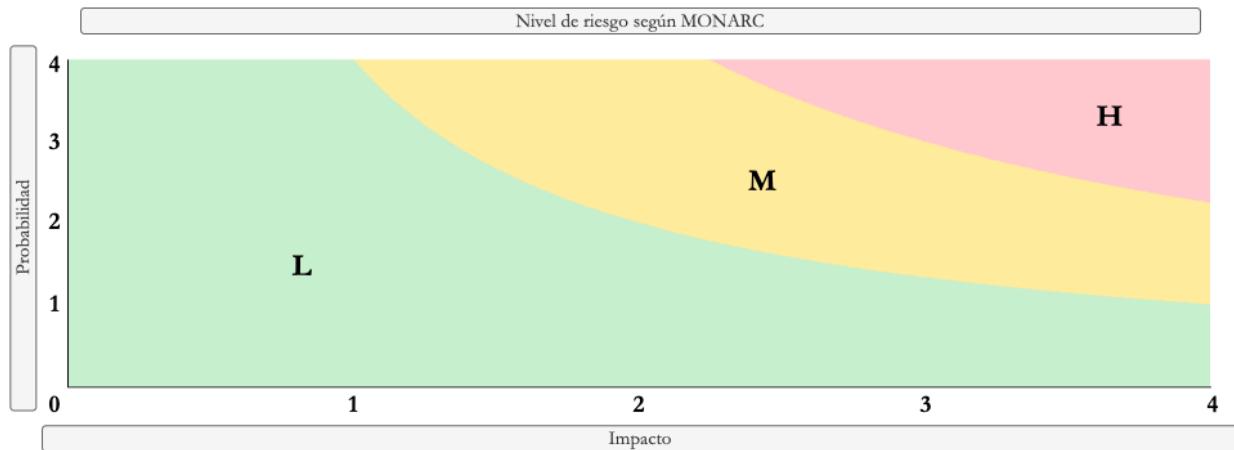


Figura 9.7: Mapa de calor para el cálculo del riesgo en MONARC

- ITSRM también tiene cinco niveles de riesgo (*Very Low (VL)*, *Low (L)*, *Medium (M)*, *High (H)*, *Very High (VH)*) que se calculan de la siguiente forma (Figura 9.8). Este es el método que se utilizará como estándar para poder comparar los resultados de las distintas metodologías.

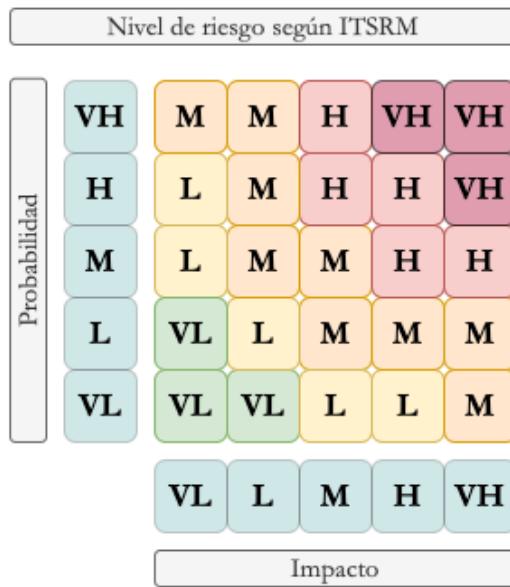


Figura 9.8: Mapa de calor para el cálculo del riesgo en ITSRM

- CRAMM se centra en la valoración de los activos, la severidad de las vulnerabilidades y la consecuencia de las amenazas para calcular el riesgo, en lugar de la probabilidad e impacto como ocurría en los casos anteriores. La escala de riesgo en esta metodología se define por números enteros entre 1 y 7, como se observa en la Figura 9.9.

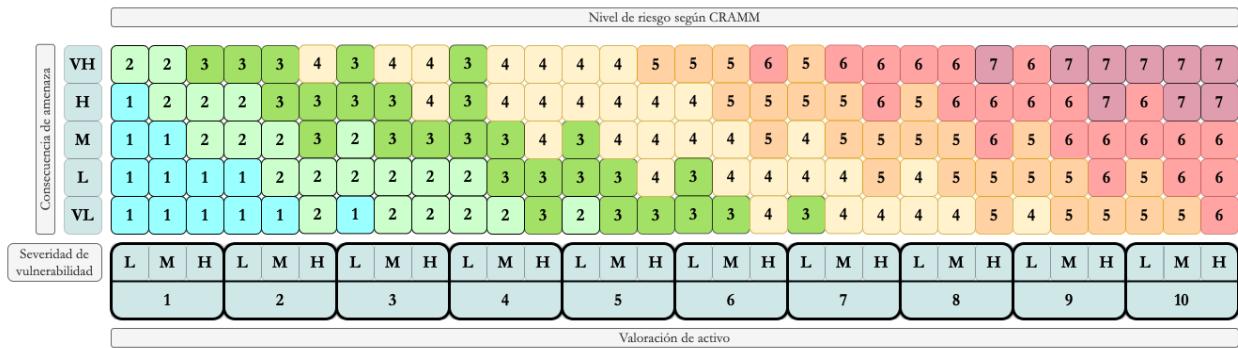


Figura 9.9: Mapa de calor para el cálculo del riesgo en CRAMM

Para diferenciar ambos cálculos, aquí se obtiene un nivel según el marco de gestión de riesgos, asociado a individuos de la clase *Risk*, y otro según el cálculo estándar, como atributo de instancias de la clase *Potential Risk*. En ambos casos, además, en el individuo que representa la información del sistema se presenta el nivel de riesgo global, pudiendo también comparar los resultados.

Como parte del sistema de soporte a la toma de decisiones que se integra en el entorno de conciencia cibersituacional, este módulo analiza las contramedidas y mitigaciones, las organiza según sus propiedades y elige la más adecuada, a partir de la que estima el riesgo residual según ITSRM.

En primer lugar, se hace una primera selección de las contramedidas que pueden estar relacionadas con cada riesgo potencial, utilizando reglas de inferencia. Por ejemplo, la que se muestra a continuación compara las palabras clave de una mitigación con el tipo de amenaza asociada al riesgo para proponerla. También se proponen aquellas recomendadas por MITRE ATT&CK para hacer frente a una técnica concreta.

```

CONSTRUCT {
?m uri:reacts_against ?risk.
?risk uri:is_affected_by ?m.
} WHERE {
?risk a uri:Potential_Risk.
?threat a uri:Threat.
?threat rdf:type ?x.
?threat uri:generates ?risk.
?asset a uri:Assets.
?threat uri:affects ?asset.
?m a uri:Mitigations.
?m uri:protects ?asset.
?m uri:keywords ?k.
FILTER CONTAINS(str(?x), str(?k))
}

```

Regla de inferencia: Pre-selección de contramedidas

Tras esta pre-selección, el sistema descarta aquellas que tienen el atributo *disabled* a *True*, ya que por algún motivo no están disponibles en el momento de la elección, y aquellas cuyo impacto es mayor que el riesgo al que se intenta responder. El resto se evalúan de la siguiente forma:

1. Se extrae el valor asociado al coste de despliegue de cada contramedida.
2. Se obtiene el atributo correspondiente al tiempo hasta estar disponible en cada una.
3. Se calcula una puntuación para cada mitigación, escalando entre el 0 y el 10 los parámetros que se van a indicar a continuación, y restando los aspectos negativos de los positivos. Si alguna de las contramedidas está recomendada por ATT&CK, se le proporciona mayor peso a su efectividad en el cómputo que a las demás.
 - Parámetros negativos: Esfuerzo de despliegue, complejidad de instalación y complejidad de operación.
 - Parámetros positivos: Efectividad de la contramedida y factor de mitigación.
4. Con los tres valores (coste, tiempo y puntuación) calculados, se localizan las contramedidas que tengan mayor puntuación, menor coste y menor tiempo. Si al menos dos de las tres coinciden, en la instancia de esta mitigación se actualiza el parámetro *enabled* a *True*. Si han salido tres mitigaciones distintas en el análisis de cada parámetro, se da prioridad a la que tiene mayor puntuación, y luego a la que tiene menor coste, y también se modifica el atributo *enabled* a *True*.

Finalmente, se calcula el riesgo residual teniendo en cuenta el factor de mitigación (Figura suponiendo que la contramedida ha sido desplegada para hacer frente a aquellos riesgos potenciales cuyo nivel está por encima del criterio de aceptación del riesgo definido en el entorno. También se estima un riesgo residual global, y se incluye en la instancia del sistema.

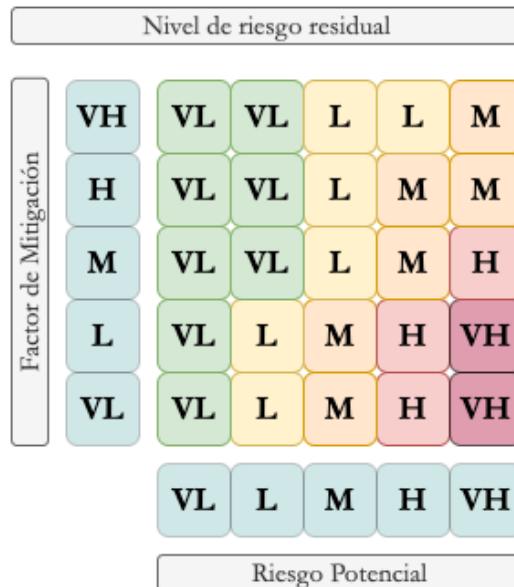


Figura 9.10: Mapa de calor para el cálculo del riesgo residual

En resumen, en la Figura 9.11 se presenta el flujo de trabajo de una ejecución en el módulo de gestión de la ontología cuando se recibe un nuevo incidente. Sin embargo, para llevar a cabo el análisis, gestión y evaluación de los riesgos de forma dinámica, este proceso se repetiría en bucle: tras la selección de la contramedida óptima para el incidente recibido y el cálculo del riesgo residual, el sistema vuelve al recuadro de espera (en color rosa en la figura) hasta que el entorno de conciencia cibersituacional vuelva a identificar y caracterizar un ciberataque. Cabe destacar que el estudio de los escenarios de riesgo debe incluir amenazas no conocidas, de manera que únicamente en caso de error en el incidente el sistema volverá al estado de espera.

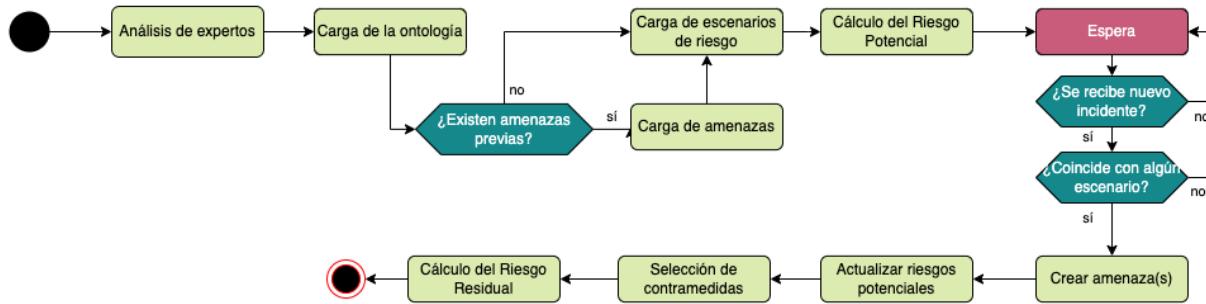


Figura 9.11: Flujo de trabajo del gestor de la ontología

9.4 Validación

9.4.1 Metodología de validación

Para validar esta propuesta de interoperabilidad se utilizan distintas aproximaciones. En primer lugar, se verifican los aspectos de la ontología y la metodología propuesta en relación con los aspectos de los distintos marcos presentados en el Capítulo 5.

Por otra parte, se presenta la realización del conjunto de casos de uso descritos en [3], comprobando si se obtienen los resultados esperados. Además, se presenta otro caso de uso centrado en validar el funcionamiento completo del sistema y la metodología propuesta para la gestión de riesgos. Para ello, se diseña un prototipo en Python y *Protégé* para crear, gestionar y visualizar la ontología.

El primer caso de uso pone a prueba que el sistema acepte información en el formato de un marco de gestión de riesgos específico (MONARC y MAGERIT), verificando que el sistema lleva a cabo el cálculo según esa metodología y según el estándar ITSRM, y permite compararlos. Para ello se crea un escenario simulado: el catálogo de activos se completa con ocho instancias (tres activos primarios y cinco secundarios), el de vulnerabilidades con tres individuos relacionados con los activos anteriores (Tabla 9.1), y la amenaza para llevar a cabo la traducción será denegación de servicio (*Denial of Service*).

Tabla 9.1: Catálogo de vulnerabilidades. Caso de uso 1

CVE	Tipo	CWE	CPE	CVSS	Activo
CVE-2023-0001	<i>Overflow</i>	CWE-1, CWE-2	CPE-1	<i>High</i>	SA(2)
CVE-2023-0002	<i>SQL Injection</i>	CWE-003	CPE-2, CPE-3	<i>Medium</i>	SA(2)
CVE-2023-0003	<i>File Inclusion</i>	CWE-004, CWE-005	CPE-1	<i>Critical</i>	SA(1), SA(2)

Por otro lado, en el último caso que se presenta, se prueban y validan todos los procesos identificados en la Figura 9.11.

9.4.2 Validación de los aspectos funcionales

Las metodologías de gestión de riesgo tienen un conjunto de aspectos funcionales, introducidos en la Sección 5.2.6, que se validarán en este apartado: Taxonomía y Evaluación de los activos, Catálogos de amenazas y vulnerabilidades y Cálculo del riesgo.

- Taxonomía de activos: En este aspecto se evalúa la capacidad de la metodología de clasificar activos. En la propuesta presentada existen dos sub-clases principales, correspondientes a los activos primarios (que contiene información/datos y procesos/funciones/servicios) y los activos secundarios (donde se incluyen activos de tipo hardware, infraestructura, localización, personal y software). Además, con esta estructura y el código desarrollado en el gestor, los activos del catálogo de entrada en la taxonomía propia de cada marco de gestión se traducen a alguna de estas sub-clases, contribuyendo a la interoperabilidad e intercambio de catálogos, como se comprobará en los casos de uso.
- Valoración de activos: Esta característica considera la capacidad de evaluar los activos según un criterio definido. En este sistema se valoran en tres de las dimensiones de seguridad (confidencialidad, integridad y disponibilidad) en una escala entre 1 y 10, que además varía dinámicamente según las amenazas a las que se exponen y las contramedidas que los protegen. Además, estas dimensiones se utilizan en la mayoría de marcos, por lo que puede heredar información de otros sistemas o comenzar la valoración desde cero.
- Catálogo de amenazas: El tipo de amenazas que considera el sistema es una lista pre-definida por la metodología ITSRM. En el código del gestor también se incluyen traducciones automáticas entre las amenazas definidas en otras metodologías, y se asocian con su equivalente en el marco ITSRM.
- Catálogo de vulnerabilidades: Este aspecto es una entrada opcional en la mayoría de las metodologías analizadas. En el sistema, se incluye el catálogo como entrada con datos específicos de CVE.
- Cálculo de riesgo: Cada metodología tiene un procedimiento para calcular el nivel de riesgo, como se muestra en los mapas de calor anteriores. Esta propuesta mantiene los

métodos propios de los marcos incluidos, además del procedimiento estándar basado en ITSRM para poder comparar los resultados o intercambiar información.

Los distintos procesos que componen la metodología se llevan a cabo en el gestor de la ontología y la definición de los catálogos, finalizando en la mayor parte de los casos con el cálculo de riesgo.

9.4.3 Caso de uso 1: MONARC y MAGERIT

En el primer caso de uso presentado se pone a prueba la interoperabilidad del sistema con los marcos MONARC y MAGERIT como metodología original. Para ello, se crearán dos casos de uso reducidos sobre el mismo entorno y se comparará su resultado.

En primer lugar, el sistema tiene como datos de entrada los procedentes del marco MONARC. Su forma de calcular el riesgo y la estándar (ITSRM) difieren en la cantidad de niveles (3 en MONARC y 5 en ITSRM), por lo que inicialmente no es comparable un riesgo clasificado alto o bajo en MONARC con uno clasificado igual en ITSRM, en el primer caso se refiere a los límites de la escala mientras que en el segundo son niveles intermedios.

Los datos de activos que se introducirán en el escenario simulado para este caso de uso se detallan en la Tabla 9.2, incluyendo la valoración de los activos secundarios.

Tabla 9.2: Catálogo de activos - MONARC (PA: Activo principal, SA: Activo secundario)

Activo	PA/SA	Tipo	C	I	A
Asset_Monarc_1	PA(1)	<i>Information</i>	-	-	-
Asset_Monarc_4	SA(1)	HW	5	3	2
Asset_Monarc_5	SA(2)	SW	10	6	3
Asset_Monarc_2	PA(2)	<i>Functions</i>	-	-	-
Asset_Monarc_6	SA(3)	<i>Personnel</i>	7	9	3
Asset_Monarc_7	SA(4)	SW	9	8	7
Asset_Monarc_3	PA(3)	<i>Information</i>	-	-	-
Asset_Monarc_8	SA(5)	SW	1	2	3

Las vulnerabilidades descritas anteriormente (Tabla 9.1) también se introducen como entrada. Como la finalidad de este caso de uso es validar el cálculo de riesgo, se introduce en el sistema una amenaza de denegación de servicio (*Denial of Service* (DoS)) con los mayores valores

para la probabilidad e impacto en la metodología MONARC (4 sobre 4), que se representa en la Tabla 9.3.

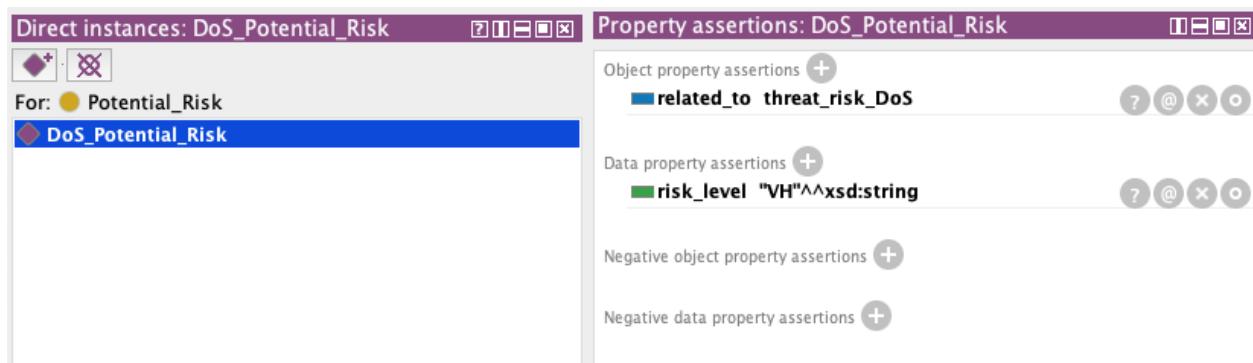
Tabla 9.3: Amenaza DoS - MONARC

Características de la amenaza DoS	Valor
Activos principales	Asset_Monarc_2
Activos secundarios	Asset_Monarc_6
Vulnerabilidad	CVE-2023-0001
Categoría general	<i>Willful</i>
Categoría específica	<i>Denial of Service (DoS)</i>
Dimensiones de seguridad afectadas	C,I,A
Efecto sobre las dimensiones de seguridad	0.7,0.6,0.4
Origen	<i>Deliberated</i>
Impacto	4
Probabilidad	4

Cuando la herramienta se ejecuta, se espera obtener según la metodología MONARC (Figura 9.7) un nivel de riesgo alto (*High*, H), que es el máximo de esta escala, por lo que en la metodología ITSRM debería corresponder a la máxima categoría también, muy alto (*Very High*, VH) en lugar de H. En la Figura 9.12 se muestran los resultados obtenidos en la herramienta. El resultado coincide con el esperado, y además se ha comprobado que los catálogos de entrada con términos de MONARC son compatibles con el sistema.

The screenshot shows the MONARC tool's interface with two main panels. The left panel, titled 'Direct instances: DoS_Risk', lists an instance named 'DoS_Risk' under a 'Risk' category. The right panel, titled 'Property assertions: DoS_Risk', displays several assertions. One assertion is highlighted in green: 'original_risk_value "H"^^xsd:string'. Other assertions include 'Object property assertions' and 'Data property assertions'. The bottom of the right panel has buttons for '?', '@', 'X', and 'O'.

(a) Cálculo de riesgo según MONARC



(b) Cálculo de riesgo según ITSRM

Figura 9.12: Comparación del cálculo de riesgo en el escenario de MONARC

En segundo lugar, se genera un caso similar al anterior, pero con los datos de la metodología de entrada procedentes del marco MAGERIT y, por tanto, en castellano. Su escala de riesgos sí es similar a la de ITSRM, ambos tienen cinco niveles, aunque las condiciones y cálculos para pertenecer a cada uno de ellos, difieren.

El escenario simulado anterior se adapta a la nueva metodología, como se muestra en la Tabla 9.4: los tipos de activos están en castellano y no hay una diferenciación explícita entre activos primarios y de soporte.

Tabla 9.4: Catálogo de activos - MAGERIT

Activo	Tipo	C	I	A
Asset_Magerit_1	Datos	-	-	-
Asset_Magerit_4	HW	5	3	2
Asset_Magerit_5	SW	10	6	3
Asset_Magerit_2	Servicios	-	-	-
Asset_Magerit_6	Personal	7	9	3
Asset_Magerit_7	SW	9	8	7
Asset_Magerit_3	Datos	-	-	-
Asset_Magerit_8	SW	1	2	3

Las vulnerabilidades que afectan a los activos son las mismas que en la simulación anterior (Tabla 9.1). La amenaza que se valida en este caso, DoS, se define con una probabilidad

de 3 sobre 5, categoría media (M), y un impacto de 4 sobre 5, categoría alta (A) según la metodología MAGERIT, como se refleja en la Tabla 9.5.

Tabla 9.5: Amenaza DoS - MAGERIT

Características de la amenaza DoS	Valor
Activos primarios	Asset_Magerit_1
Activos secundarios	Asset_Magerit_4
Vulnerabilidades	CVE-2023-0001
Categoría general	Deliberada
Categoría específica	Denegación de Servicio
Dimensiones de seguridad afectadas	C,I,A
Efecto sobre las dimensiones de seguridad	0.7,0.6,0.4
Origen	<i>Deliberated</i>
Impacto	A
Probabilidad	M

Según la Figura 9.6, los resultados esperados en el sistema y la ontología serán que mediante el cálculo de la metodología original, el nivel de riesgo es alto (A) y utilizando el proceso estándar de ITSRM, es alto también (H). En la Figura 9.13 se muestran los resultados obtenidos, que coinciden con lo esperado y, de nuevo, los términos de MAGERIT, introducidos en el sistema en castellano son compatibles con la herramienta.

The screenshot shows the MAGERIT interface. On the left, under 'Direct instances: DoS_Risk', there is a list of instances. One instance, 'DoS_Risk', is selected and highlighted in blue. On the right, under 'Property assertions: DoS_Risk', there are four sections: 'Object property assertions' (empty), 'Data property assertions' (containing a single assertion 'original_risk_value "A"^^xsd:string'), 'Negative object property assertions' (empty), and 'Negative data property assertions' (empty). A toolbar with various icons is visible at the top and bottom of the interface.

(a) Cálculo de riesgo según MAGERIT

The screenshot shows the ITS-RM interface. On the left, under 'Direct instances: DoS_Potential_Risk', there is a list of instances. One instance, 'DoS_Potential_Risk', is selected and highlighted in blue. On the right, under 'Property assertions: DoS_Potential_Risk', there are four sections: 'Object property assertions' (empty), 'Data property assertions' (containing a single assertion 'related_to threat_risk_DoS_3 risk_level "H"^^xsd:string'), 'Negative object property assertions' (empty), and 'Negative data property assertions' (empty). A toolbar with various icons is visible at the top and bottom of the interface.

(b) Cálculo de riesgo según ITS-RM

Figura 9.13: Comparación del cálculo de riesgo en el escenario de MAGERIT

En el conjunto de este caso de uso se han podido validar los procesos de traducción de términos y el cálculo de riesgos mediante MONARC, MAGERIT e ITSRM. Pero el objetivo principal, que era poder comparar los riesgos, también se verifica: en el escenario de MONARC, el riesgo según la metodología era alto (H), al igual que en escenario de MAGERIT (A), teniendo dos amenazas con propiedades distintas. Aparentemente, estas dos podrían parecer igualmente importantes, pero al compararlas en una escala estándar, ITSRM, se observa que el riesgo en el caso del escenario de MONARC es muy alto (VH), mientras que el de MAGERIT sigue siendo alto (H), por lo que es un riesgo que está un nivel por debajo. Además, este sistema permitiría compartir información desde otras metodologías para hacer frente a los riesgos, como catálogos de contramedidas que hayan sido efectivas previamente.

9.4.4 Caso de uso 2: Escenario general

Para la validación de este caso de uso se utiliza EBIOS como metodología original para comprobar que también es compatible con el sistema mientras se verifica el funcionamiento de los procesos descritos en el capítulo anterior, evaluando la operación completa de la herramienta.

Catálogos de entrada

Como primer paso, para esta simulación se han creado unos catálogos que contienen la siguiente información:

- Activos: En este escenario se está trabajando con unos dispositivos hardware que están controlados por una aplicación y generan información. Por tanto, el *Primary Asset 1*, que representa la información, se relaciona con dos activos de soporte, *Supporting Asset 1* (hardware) y 2 (software). Por otro lado, el *Primary Asset 2* que representa el servicio que se proporciona, se asocia con los *Supporting Asset 3* (hardware) y 4 (software). Las dimensiones de seguridad de los activos primarios se calculan a partir de los de soporte con los que tienen relación (Tabla 9.6). Además, en la Figura 9.14 se muestra como ejemplo la carga del primer activo primario en la ontología tras la ejecución de la regla de inferencia correspondiente, que calcula su valoración en las dimensiones de seguridad según los activos de soporte con los que tiene relación, coincidiendo con los resultados mostrados en la tabla.

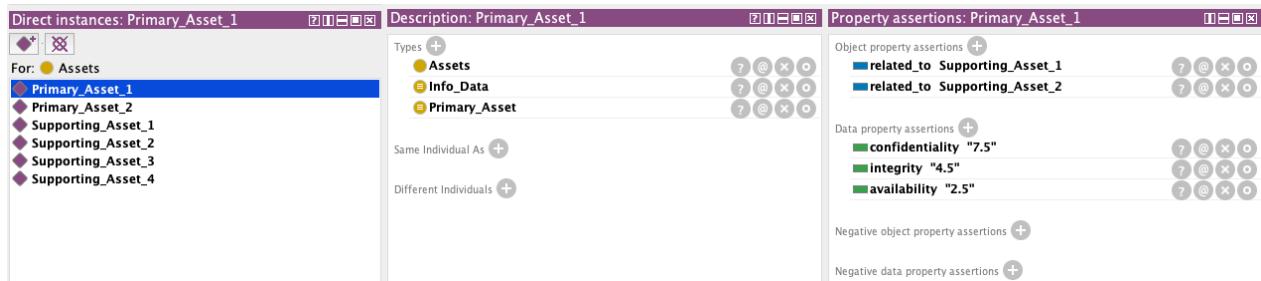


Figura 9.14: *Primary Asset 1* en la ontología

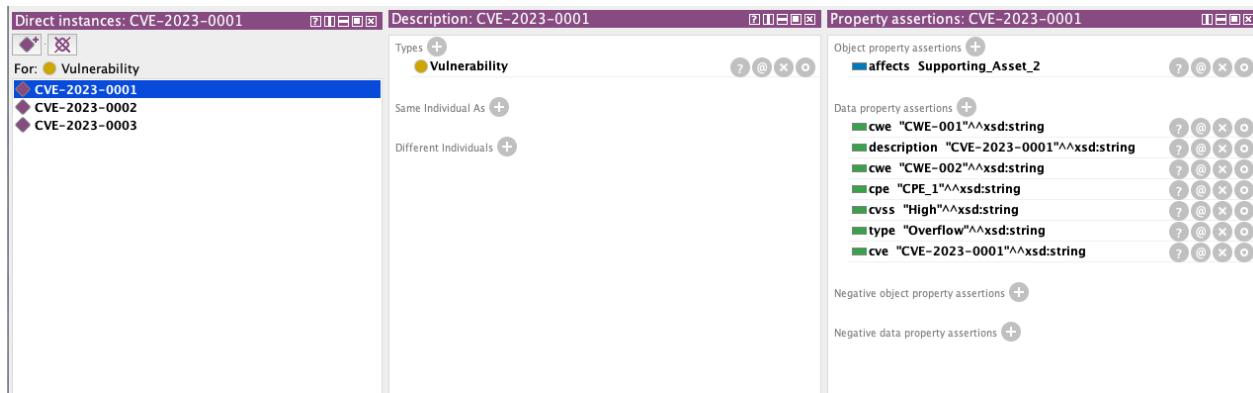
Tabla 9.6: Catálogo de activos. Caso de uso 2

Activo	Confidencialidad	Integridad	Disponibilidad
<i>Primary Asset 1</i>	7.5	4.5	2.5
<i>Supporting Asset 1</i>	5	3	2
<i>Supporting Asset 2</i>	10	6	3
<i>Primary Asset 2</i>	5	5	5
<i>Supporting Asset 3</i>	1	2	3
<i>Supporting Asset 4</i>	9	8	7

- Las vulnerabilidades, presentadas en la Tabla 9.7, se consideran ejemplos genéricos, ya que no se incluye información específica de los activos que permita identificar las vulnerabilidades a las que se exponen. Un ejemplo se muestra en la Figura 9.15.

Tabla 9.7: Catálogo de vulnerabilidades. Caso de uso 2

CVE	CWE	CPE	CVSS	Asset
CVE-2023-0001	CWE-{001,002}	CPE-1	High	S.A. 3
CVE-2023-0002	CWE-003	CPE-{2,3}	Medium	S.A. 2
CVE-2023-0003	CWE-{004,005}	CPE-1	Critical	S.A. 1,2

**Figura 9.15:** Vulnerabilidades en la ontología

- Las amenazas previas a la ejecución del sistema se presentan en la Tabla 9.8, mostrando en la Figura 9.16 un ejemplo de la amenaza no intencionada (*Threat3*) incluida en la ontología.

Tabla 9.8: Catálogo de amenazas. Caso de uso 2

ID	Tipo	Sub-tipo	Consecuencia sobre C,I,A	Activos
<i>Threat1</i>	<i>Natural</i>	<i>Flood</i>	(0, 0, 0.2)	S.A. 1
<i>Threat2</i>	<i>Industrial</i>	<i>Fire</i>	(0, 0, 0.8)	S.A. 1
<i>Threat3</i>	<i>Unintentional failure</i>	<i>User Error</i>	(0.1, 0.4, 0.6)	S.A. 1
<i>Threat4</i>	<i>Willful attack</i>	<i>DoS</i>	(0, 0, 0.7)	S.A. 1
<i>Threat5</i>	<i>Service related</i>	<i>Lock-In</i>	(0, 0, 0.9)	S.A. 4
<i>Threat6</i>	<i>Service related</i>	<i>Lock-In</i>	(0.7, 0.2, 0.9)	S.A. 2

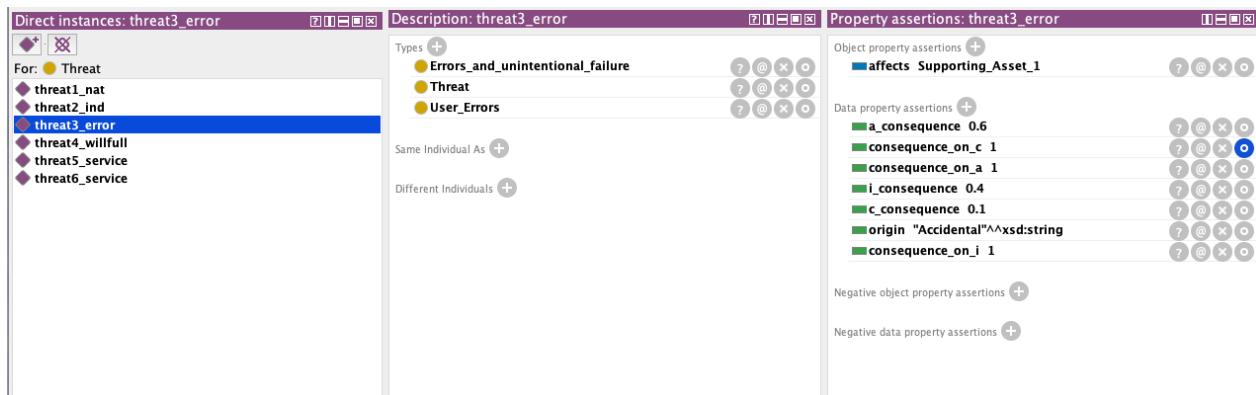


Figura 9.16: Ejemplo de amenazas

Además, estas amenazas tienen un impacto en las dimensiones de los activos. Estos cambios se presentan en la Tabla 9.9, mostrando lo vulnerables que son estos activos a las amenazas de este entorno. En la Figura 9.17, en comparación con la Figura 9.14, se puede observar esta consecuencia y la actualización de los valores de los activos primarios mediante la ejecución de la primera regla de inferencia.

Tabla 9.9: Catálogo de activos afectado por las amenazas. Caso de uso 2

Activos	Confidencialidad	Integridad	Disponibilidad
<i>Primary Asset 1</i>	$7.5 \Rightarrow 3.75$	$4.5 \Rightarrow 3.3$	$2.5 \Rightarrow 0.169$
<i>Supporting Asset 1</i>	$5 \Rightarrow 4.5$	$3 \Rightarrow 1.8$	$2 \Rightarrow 0.0384$
<i>Supporting Asset 2</i>	$10 \Rightarrow 3$	$6 \Rightarrow 4.8$	$3 \Rightarrow 0.3$
<i>Primary Asset 2</i>	5	5	$5 \Rightarrow 1.85$
<i>Supporting Asset 3</i>	1	2	3
<i>Supporting Asset 4</i>	9	8	$7 \Rightarrow 0.7$

Direct instances: Supporting_Asset_1 Property assertions: Supporting_Asset_1

For: Assets

- Primary_Asset_1
- Primary_Asset_2
- Supporting_Asset_1**
- Supporting_Asset_2
- Supporting_Asset_3
- Supporting_Asset_4

Object property assertions +

- is_affected_by threat4_willfull
- is_affected_by threat2_ind
- is_affected_by threat1_nat
- is_affected_by threat3_error

Data property assertions +

- integrity 1.7999999999999998
- availability 0.03839999999999999
- confidentiality 4.5

Negative object property assertions +

Negative data property assertions +

(a) Efecto de la amenaza sobre un activo de soporte

Direct instances: Primary_Asset_1 Property assertions: Primary_Asset_1

For: Assets

- Primary_Asset_1**
- Primary_Asset_2
- Supporting_Asset_1
- Supporting_Asset_2
- Supporting_Asset_3
- Supporting_Asset_4

Object property assertions +

- related_to Supporting_Asset_1
- related_to Supporting_Asset_2

Data property assertions +

- confidentiality "3.75"
- availability "0.1691999999999999"
- integrity "3.3"

Negative object property assertions +

Negative data property assertions +

(b) Efecto de la amenaza sobre un activo primario

Figura 9.17: Ejemplo del efecto de las amenazas sobre los activos

- Por último, las mitigaciones (Tabla 9.10). En este caso, aún ninguna se ha desplegado y no hay ninguna deshabilitada, pero esta situación cambiará a raíz de la selección de contramedidas. La integración de éstas en la ontología se muestra en la Figura 9.18.

Tabla 9.10: Catálogo de contramedidas. Caso de uso 2

ID	<i>Mitigation 1</i>	<i>Mitigation 2</i>	<i>Mitigation 3</i>	<i>Mitigation 4</i>	<i>Mitigation 5</i>
Palabras clave	<i>SW Tampering</i> <i>Destructive attack</i> <i>User Error</i> CVE-2023-0001	<i>Destructive Attack</i>	<i>User Error</i>	<i>User Error</i>	<i>User Error</i>
Complejidad operacional	1	2	7	2	5
Complejidad instalación	4	8	8	8	7
Coste de despliegue	2000	10000	10	100	120
Esfuerzo de despliegue	5	9	9	9	9
Tiempo. disp.	1	365	365	36	120
Impacto	VH	VL	VH	VL	VL
Factor de Mitigación	0.6	0.9	0.6	0.6	0.9
Disabled	<i>False</i>	<i>False</i>	<i>False</i>	<i>False</i>	<i>False</i>
Enabled	<i>False</i>	<i>False</i>	<i>False</i>	<i>False</i>	<i>False</i>
Activo Protegido	S.A. 2	S.A. 3	S.A. 1	S.A. 1	S.A. 1

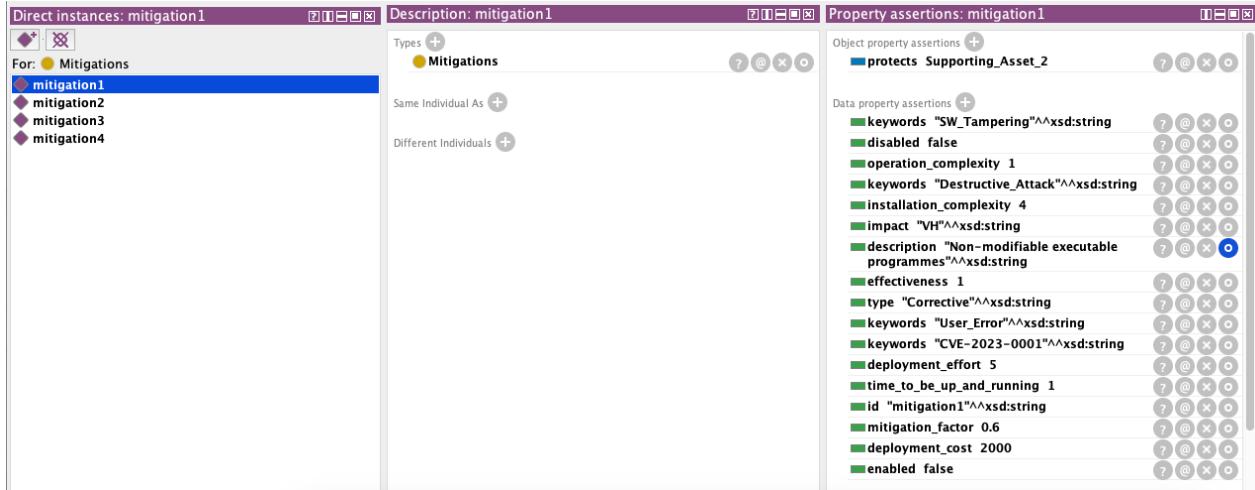


Figura 9.18: Ejemplo de contramedida

Escenarios de riesgo e incidentes

A partir del análisis previo llevado a cabo por expertos se identifica un conjunto de escenarios de riesgo, definidos por los incidentes que se reciben en el sistema, las vulnerabilidades que explotan en algunos activos, y las amenazas que se generan a partir de esta situación (Tabla 9.11). En la Figura 9.19 se puede observar un ejemplo de instancia de escenario de riesgo, sus atributos y sus relaciones con individuos de vulnerabilidades y activos. Hasta que no se registre un incidente que coincida con el escenario, no se crea la instancia de amenaza relacionada.

Tabla 9.11: Escenarios de Riesgo. Caso de uso 2

ID	Activos	Vulnerabilidad	Amenaza	Probabilidad	Impacto
<i>Risk Scenario 1</i>	S.A. 1, P.A. 1	CVE-2023-0001	<i>Deliberated Malware Diffusion</i>	1	3
<i>Risk Scenario 2</i>	S.A. 3, P.A. 2	CVE-2023-0001	<i>Destruction of Media</i>	2	4

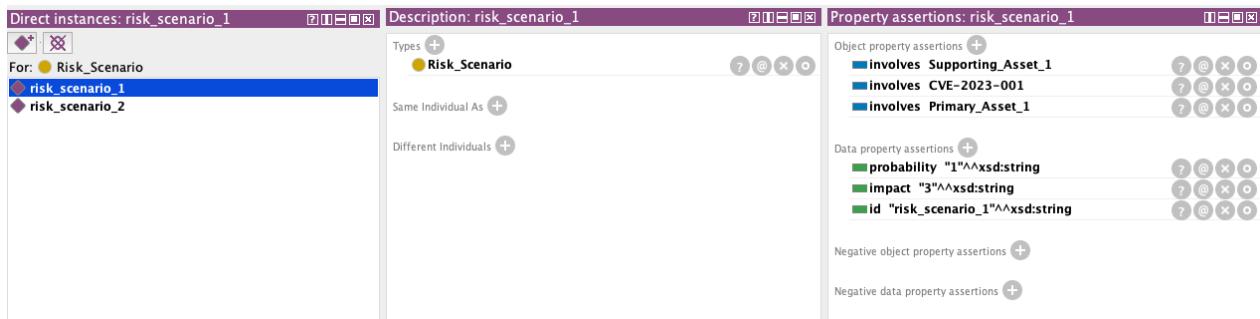


Figura 9.19: Ejemplo de escenario de riesgo

De esta forma, cuando se recibe y caracteriza un incidente como los que se presentan en la Tabla 9.12, se comprueba el activo que afecta a partir de la explotación de una vulnerabilidad, se compara con los escenarios de riesgo definidos y se asocia con los que encajen.

Tabla 9.12: Incidentes registrados. Caso de uso 2

ID	TTPs	Activos	Vulnerabilidad	Escenario de riesgo asociado
Incident 1	T1046	S.A. 1, S.A. 4	CVE-2023-0002	Ninguno
Indicent 2	T1595	S.A. 3, S.A. 4	CVE-2023-0001 CVE-2023-0003	Risk Scenario 2

En este caso de uso, el primer incidente no encaja con los escenarios definidos, por lo tanto se considera una amenaza no conocida y se trata como tal. Sin embargo, el segundo sí encaja con el segundo escenario de riesgo presentado en la Tabla 9.11. Este ciberataque, por tanto afecta al *Supporting Asset 3* a través de la vulnerabilidad CVE-2023-0001, generando un ataque destructivo (*Destruction of Media*) de tipo deliberado (*Willful*), donde el impacto que tiene es sobre la dimensión de disponibilidad del activo y es igual a 0.4, como se puede observar en la Figura 9.20. Las dimensiones de seguridad del activo afectado se modifican de acuerdo con la consecuencia de la amenaza. En la Tabla 9.13 se muestran los valores finales de las dimensiones como evolución de las Tablas 9.6 y 9.9.

(a) Ejemplo de incidente en la ontología

The screenshot shows the ontology editor interface with three panels:

- Direct instances: incident2**: Shows a tree view of instances under the type **Incident**. Selected is **incident2**.
- Description: incident2**: Shows the type **Incident** selected. Below it are buttons for "Same Individual As" and "Different Individuals".
- Property assertions: incident2**: Shows object and data property assertions for **incident2**. Object properties include **affects Supporting_Asset_4**, **affects Supporting_Asset_3**, **generates threat_risk_2**, **exploits CVE-2023-0001**, **exploits CVE-2023-0003**, and **related_to T1595**. Data properties include **id "incident2"^^xsd:string**, **description "Incident 2"^^xsd:string**, and **date "2023-11-24"^^xsd:string**.

(b) Ejemplo de escenario de riesgo que genera amenaza

The screenshot shows the ontology editor interface with three panels:

- Direct instances: risk_scenario_2**: Shows a tree view of instances under the type **Risk_Scenario**. Selected is **risk_scenario_2**.
- Description: risk_scenario_2**: Shows the type **Risk_Scenario** selected. Below it are buttons for "Same Individual As" and "Different Individuals".
- Property assertions: risk_scenario_2**: Shows object and data property assertions for **risk_scenario_2**. Object properties include **involves CVE-2023-0001**, **generates threat_risk_2**, **involves Supporting_Asset_3**, and **involves Primary_Asset_2**. Data properties include **probability "2"^^xsd:string**, **impact "4"^^xsd:string**, and **id "risk_scenario_2"^^xsd:string**.

(c) Ejemplo de amenaza generada por escenario de riesgo

The screenshot shows the ontology editor interface with three panels:

- Direct instances: threat_risk_2**: Shows a tree view of instances under the type **Threat**. Selected is **threat_risk_2**.
- Description: threat_risk_2**: Shows the type **Threat** selected. Below it are buttons for "Same Individual As" and "Different Individuals".
- Property assertions: threat_risk_2**: Shows object and data property assertions for **threat_risk_2**. Object properties include **affects Supporting_Asset_3**. Data properties include **consequence_on_c 0**, **consequence_on_a 1**, **origin "Deliberate"^^xsd:string**, **i_consequence 0**, **a_consequence 0.4**, **c_consequence 0**, and **consequence_on_j 0**.

Figura 9.20: Ejemplo de la generación de amenazas a partir de los escenarios de riesgo

Tabla 9.13: Catálogo de activos afectado por la amenaza del escenario de riesgo. Caso de uso 2

Activo	Confidencialidad	Integridad	Disponibilidad
<i>Primary Asset 1</i>	3.75	3.3	0.169
<i>Supporting Asset 1</i>	4.5	1.8	0.0384
<i>Supporting Asset 2</i>	3	4.8	0.3
<i>Primary Asset 2</i>	5	5	$1.85 \Rightarrow 1.25$
<i>Supporting Asset 3</i>	1	2	$3 \Rightarrow 1.8$
<i>Supporting Asset 4</i>	9	8	0.7

Cálculo del riesgo potencial

El siguiente paso será estimar el nivel de riesgo según las amenazas incluidas en el sistema. Este cálculo se lleva a cabo siguiendo por un lado la metodología original y el procedimiento interoperable (ITSRM). A modo de ejemplo se presenta el cálculo de riesgos para la amenaza de error de usuario (*User Error*). La única amenaza de este tipo que existe en el sistema es la amenaza previa con identificador *Threat 3* (Tabla 9.8). Según la metodología EBIOS (Figura 9.5), esta amenaza tiene impacto 2 y probabilidad 1, lo que implica un nivel de riesgo bajo (L). Siguiendo el método interoperable (ITSRM, Figura 9.8), la misma amenaza tiene un impacto bajo (L) y una probabilidad muy baja (VL), lo que implica un nivel de riesgo muy bajo (VL). En este caso, los niveles no coinciden, y la priorización de la amenaza sería distinta en cada caso. En la Figura 9.21, se reflejan los resultados de estos cálculos en el sistema.

The figure consists of three side-by-side screenshots of the EBIOS interface, each showing a different view of the risk calculation process for the 'User_Error_Risk' instance:

- Direct instances: User_Error_Risk**: Shows a list of direct instances under the 'Risk' type. The 'User_Error_Risk' instance is selected and highlighted in blue.
- Description: User_Error_Risk**: Shows the description of the 'User_Error_Risk' instance. It includes a 'Types' section with 'Risk' selected, and two 'Same Individual As' buttons: one for 'Same Individual As' and one for 'Different Individuals'.
- Property assertions: User_Error_Risk**: Shows the property assertions for the 'User_Error_Risk' instance. It includes sections for 'Object property assertions' (with a single assertion: 'original_risk_value "L"^^xsd:string'), 'Data property assertions' (with a single assertion: 'original_risk_value "L"^^xsd:string'), 'Negative object property assertions' (empty), and 'Negative data property assertions' (empty).

(a) Cálculo de riesgo de la amenaza *User Error*. Metodología EBIOS



(b) Cálculo de riesgo potencial de la amenaza *User Error*. Metodología ITSRM

Figura 9.21: Ejemplo de cálculos de riesgo para la amenaza *User Error*

En la instancia que representa el sistema se almacena el cálculo global para ambas metodologías considerando todas las amenazas que se han creado en el caso de uso. En total, según EBIOS, el riesgo global es medio (M), mientras que según ITSRM es bajo (L) (Figura 9.22). Tanto en este caso como en el de la amenaza de *User Error* el cálculo interoperable refleja una escala menor. Esto se debe a que EBIOS tiene solo tres niveles, mientras que ITSRM cuenta con cinco.

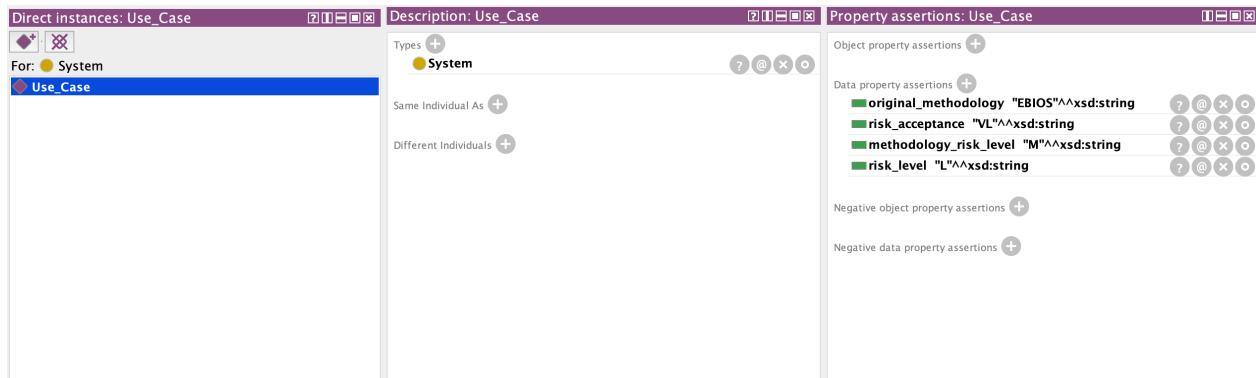


Figura 9.22: Cálculo del riesgo potencial global

Selección de contramedidas y cálculo del riesgo residual

En este paso se llevan a cabo distintas comprobaciones. En la Tabla 9.10 se mostró que ninguna contramedida estaba activa, ya que los activos no habían sido atacados. Tras la detección y caracterización de los incidentes, las mitigaciones 3, 4 y 5, que reaccionan frente al mismo tipo de ataque (en las palabras clave aparece *User Error*) se proponen para la optimización. Entre éstas, la mitigación 3 se descarta ya que su impacto (VH) es más alto que el nivel de riesgo asociado a la amenaza (VL). Según el algoritmo definido, la contramedida óptima es la mitigación 4, que por tanto se activa (atributo *enable* igual a *True*), al contrario que en la mitigación 5, como se muestra en la Figura 9.23.

(a) Mitigación 4 - Seleccionada y activada

(b) Mitigación 5 - Seleccionada pero no activada

Figura 9.23: Metodologías para la amenaza de *User Error*

Como se ha visto, la contramedida 4 genera un riesgo residual para la amenaza de *User Error*. A la vez, la mitigación 2 se selecciona para reaccionar a las amenazas de *Destructive Attack* directamente, ya que no existen otras alternativas (Figura 9.24).

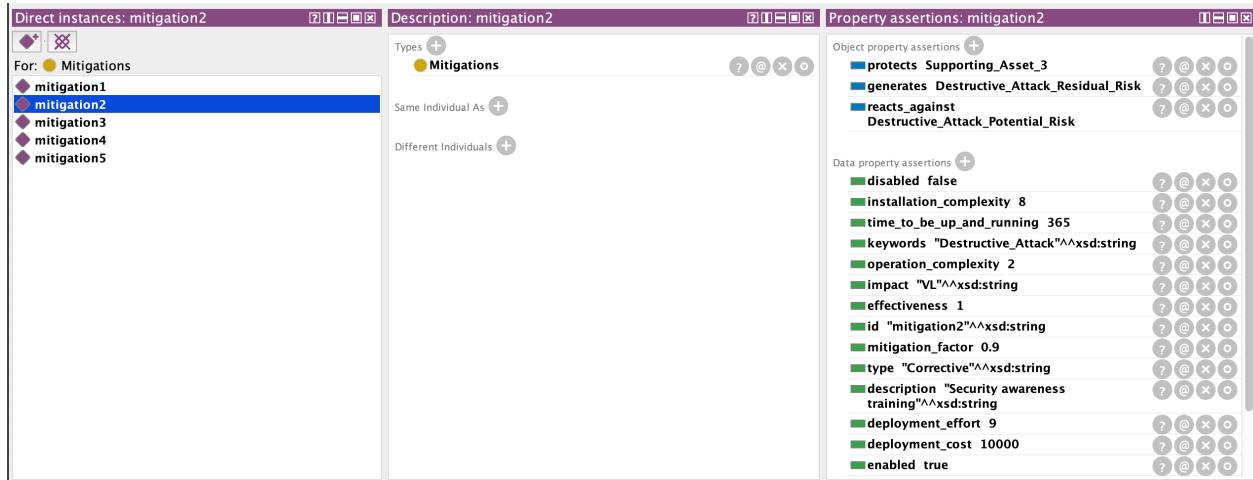
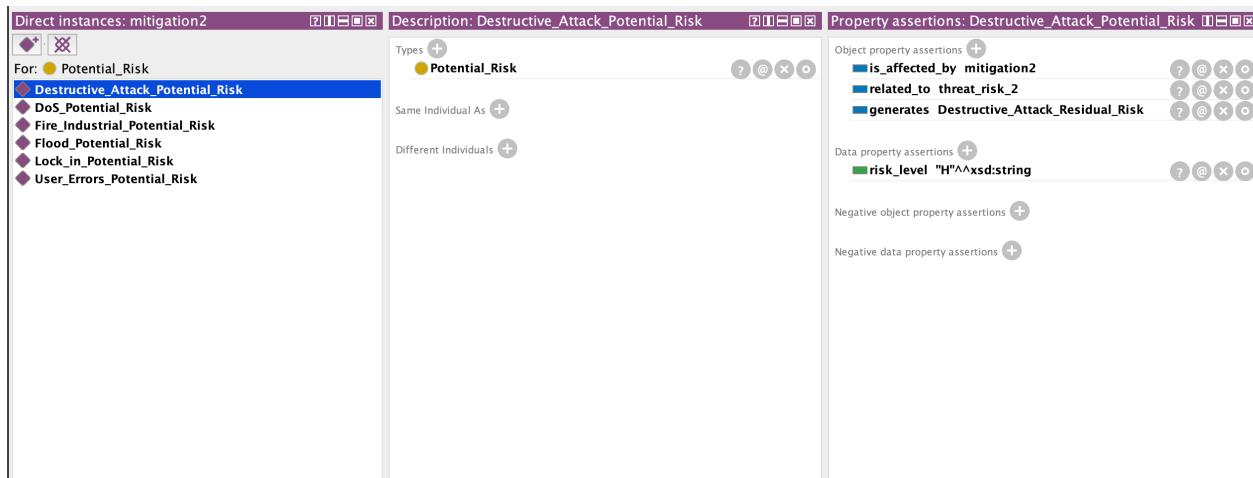
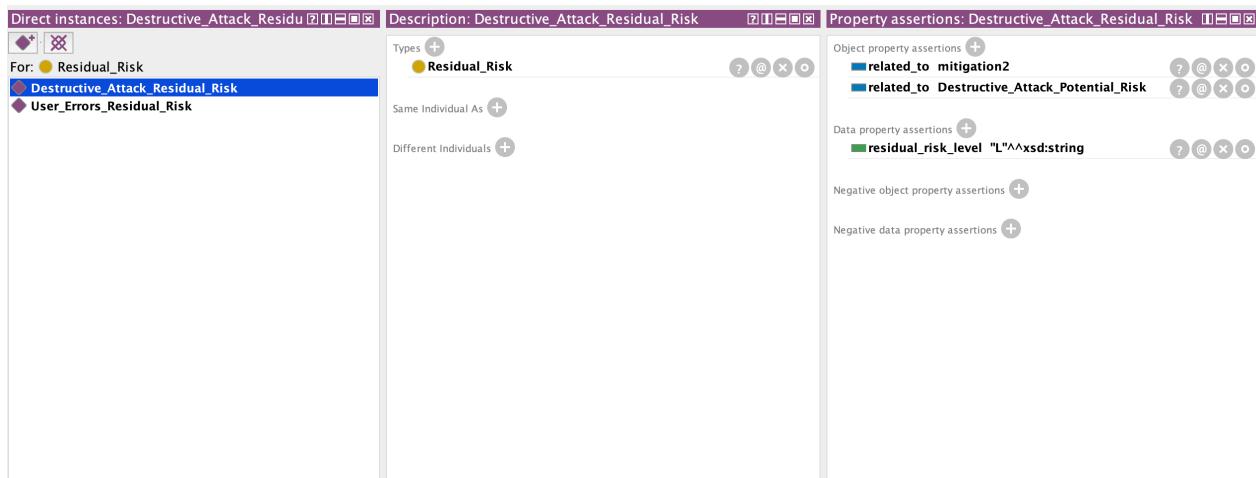


Figura 9.24: Mitigación 2 - *Destructive Attack*

Para el cálculo del riesgo residual únicamente se consideran los niveles de riesgo potenciales, creados por el procedimiento interoperable. El riesgo residual de la amenaza *User Error* era muy bajo (VL) y por tanto el efecto de la contramedida no puede reducir el nivel (riesgo residual muy bajo (VL)). Por otro lado, la amenaza *Destructive Attack* tenía un riesgo alto (H), y a partir de la aplicación de la contramedida con un factor de mitigación de 0.9 (Figura 9.10), el riesgo residual se reduce a bajo (L). Este cambio se puede observar en la Figura 9.25.



(a) Riesgo potencial de la amenaza *Destructive Attack*



(b) Riesgo residual de la amenaza *Destructive Attack*

Figura 9.25: Efecto de la contramedida sobre el riesgo de la amenaza *Destructive Attack*

Además, el riesgo residual del sistema global se calcula. Como en este caso la reducción de riesgos no es significativa en comparación con los que no se pueden mitigar, el riesgo potencial y el residual coinciden (Figura 9.26).

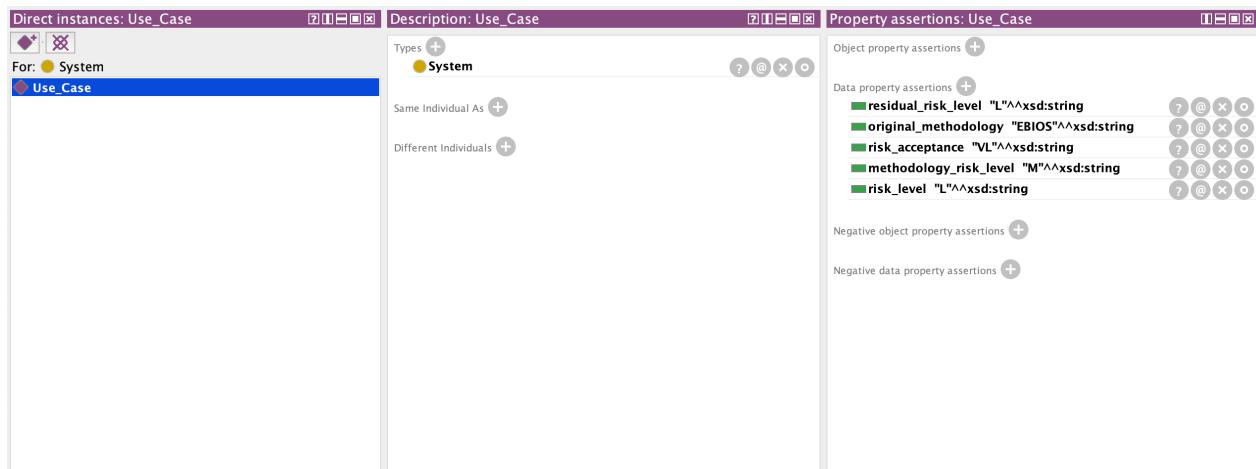


Figura 9.26: Riesgos calculados en el sistema: Metodología EBIOS, Riesgo potencial ITSRM y Riesgo residual ITSRM

Finalmente, en las Tablas 9.14 y 9.15 se resume la evolución del nivel de riesgo que se ha producido en el sistema.

Tabla 9.14: Amenazas de ejemplo - Cálculo del riesgo

Amenaza	Nivel de riesgo EBIOS	Nivel de riesgo potencial interoperable	Nivel de riesgo residual interoperable
<i>User Error</i>	L	VL	VL
<i>Destructive Attack</i>	M	H	L

Tabla 9.15: Cálculo de riesgo global

Tipo de cálculo	Valor obtenido
Nivel de riesgo EBIOS	<i>Medium</i> (M)
Nivel de riesgo potencial interoperable (ITSRM)	<i>Low</i> (L)
Nivel de riesgo residual interoperable (ITSRM)	<i>Low</i> (L)

9.5 Conclusiones

Teniendo en cuenta la relevancia de los procesos de análisis y gestión de riesgos en ciberseguridad y la cantidad de amenazas que reciben los sistemas continuamente, las metodologías son uno de los focos de atención en la actualidad, especialmente su estudio y comparación. La necesidad de llevar a cabo estos procesos en tiempo real y de intercambio de información explotan las ventajas que tienen las ontologías en este campo para generar información automáticamente, como se ha puesto de manifiesto en las investigaciones previas analizadas.

Las metodologías de gestión de riesgos más aceptadas a día de hoy a nivel europeo plantean distintos enfoques a la hora de tratar el riesgo o los catálogos de información que utilizan en el análisis, por lo que es difícil comparar los resultados, comprobar cual se adapta mejor a una situación y priorizar riesgos o compartir información sobre la eficiencia de las respuestas ante los ciberataques.

El desarrollo que se ha presentado en este capítulo trata de conseguir una metodología interoperable, que además sea dinámica y proporcione análisis en tiempo real. Por otro lado, las soluciones encontradas en la literatura, que se centran en el análisis de un entorno o caso específico, no se consideren interoperables según las propuestas de ENISA en este aspecto.

Como se ha demostrado, las ontologías permiten modelar los procesos de cualquier marco de trabajo enfocado en la gestión de riesgos, pero además tienen la capacidad de definir el comportamiento esperado para calcular el nivel de riesgo o proporcionar contramedidas para responder dinámicamente a los ciberataques.

Para validar la metodología propuesta y el diseño de la ontología y del gestor se han planteado dos tipos de casos de uso. El primero se centra en verificar la capacidad de interoperabilidad: se introducen en un prototipo del sistema amenazas definidas en dos metodologías distintas (MONARC y MAGERIT), se lleva a cabo el cálculo del nivel de riesgo según sus propias escalas y una estándar, basada en ITSRM para poder comparar las amenazas de cada escenario. Por otro lado, se propone una simulación para evaluar el funcionamiento completo del sistema, comprobando además la interoperabilidad con la metodología EBIOS y la elección de la contramedida óptima y su efecto para la reducción de riesgo en el entorno. Los resultados en ambos casos han sido positivos: los niveles de riesgos de las metodologías originales y el cálculo estándar basado en ITSRM no siempre coinciden pero, al trasladar los riesgos a esta escala común, la propuesta permite comparar los estados de riesgo de distintos sistemas que utilizan marcos diferentes no comparables a priori.

El entorno de conciencia cibersituacional que se presenta a lo largo de la Tesis Doctoral se complementa gracias a la gestión de riesgos planteada en ese capítulo, que aprovecha la caracterización de ataques para optimizar el análisis y evaluación de riesgos y la recomendación de contramedidas. La caracterización de ciberataques enriquece el proceso de gestión de riesgos en el que también se apoya la Hipótesis 2 de esta investigación, que se ha podido validar parcialmente en este capítulo y se completará en el siguiente, verificando que la elección de contramedidas basada en la información del ciberataque puede ser más precisa que las recomendaciones generales.

Capítulo 10

Validación global

En este capítulo se propone la validación del entorno de conciencia cibersituacional presentado a lo largo de la Tesis Doctoral, mostrado en la Figura 6.1: las anomalías identificadas a partir de los sistemas heterogéneos que se presentó en el Capítulo 7 y los ciberataques caracterizados en tráfico de red de los que se extraen técnicas según se desarrolló en el Capítulo 8, se recogen en el sistema dinámico e interoperable de gestión de riesgos definido en el capítulo anterior para proporcionar a los analistas de ciberseguridad información automática sobre el estado del sistema según se detecta un incidente, proponiendo respuestas para hacerles frente.

En este capítulo, en primer lugar, se presenta la metodología que se sigue a la hora de validar el sistema en conjunto (Sección 10.1). A continuación, dos casos de uso (Secciones 10.2 y 10.3). Finalmente, las conclusiones extraídas se presentan en la Sección 10.4.

10.1 Metodología de validación del sistema

Para validar la propuesta, además de las evaluaciones parciales que ya se han presentado en cada módulo individual, se define un conjunto de pruebas. La metodología está basada en un caso de uso sobre un escenario simulado que permite analizar el funcionamiento global del entorno y valorar la contribución de cada uno de los módulos para la conciencia cibersituacional en un entorno controlado. Además, las contribuciones de esta Tesis Doctoral se han validado individualmente a través de la contribución en los proyectos PLICA, de ámbito nacional para el Ministerio de Defensa, y ECYSAP, para la Comisión Europea.

Para evaluar el sistema se define un conjunto de datos procedentes de tres de los sensores diseñados para identificar anomalías, analizando el riesgo generado a través de la ontología. Para poder validar la Hipótesis 2, basada en la capacidad de la caracterización de ciberataques para mejorar la gestión de riesgos, no todos los casos de estudio harán uso del módulo de identificación de técnicas, pudiendo comparar así la gestión y evaluación de riesgos obtenida en cada uno. Finalmente, para analizar el resultado del sistema en conjunto y validar o refutar la Hipótesis 3, se presenta un caso de uso más detallado en el sistema.

El escenario de la simulación está compuesto de los activos definidos en la Tabla 10.1, organizados según se presenta en la Figura 10.1. Para simplificar lo más posible el conjunto

de pruebas se representa una ubicación o edificio donde se sitúan los sensores, los datos que generan, y que se almacenan en un servidor. Además, existe un usuario con un dispositivo móvil y un ordenador con los que se conecta al servicio de correo electrónico y gestiona datos críticos, todo ello almacenado también en el servidor. Las dimensiones de seguridad de las categorías de datos y servicios se heredan a partir de los activos de los que dependen.

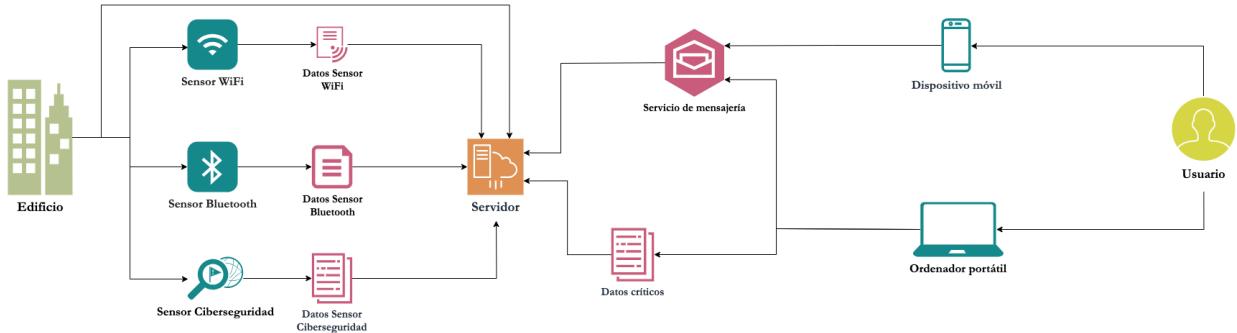


Figura 10.1: Activos del escenario de validación

Tabla 10.1: Catálogo de activos. Validación

Activo	Tipo	C	I	A
Edificio	Ubicación	2	3	6
Sensor de Wi-Fi	Hardware	8	5	9
Datos Wi-Fi	Datos	5.67	5.34	7
Sensor de Bluetooth	Hardware	9	7	7
Datos Bluetooth	Datos	6	6	6.34
Sensor de Ciberseguridad	Software	8	6	7
Datos Ciberseguridad	Datos	5.67	5.67	6.34
Servidor	Hardware	7	8	6
Usuario	Personal	8	8	9
Móvil	Hardware	6	7	9
PC	Hardware	9	9	10

Tabla 10.1: Catálogo de activos. Validación

Activo	Tipo	C	I	A
Datos críticos	Datos	6.5	7	7.75
Correo electrónico	Servicio	7.5	8	8.5

Por ello, partiendo de estos activos y para validar la funcionalidad del entorno de conciencia cibersituacional, se plantean dos simulaciones:

- En el primer caso de uso se gestiona en primer lugar información sin caracterizar, obteniendo una evaluación de riesgos potenciales y residuales en función de las respuestas seleccionadas, que en este caso serán genéricas y, a continuación, se pondrá a prueba la eficacia de la caracterización en la gestión de riesgos, extrayendo información de los ciberataques para enriquecer el proceso y proponer contramedidas adecuadas.
- El segundo caso de uso se centrará en detallar los procesos de cada uno de los módulos para evaluar el entorno de conciencia cibersituacional completo. Con ejemplos simulados de datos de los sensores, éstos se analizarán para identificar anomalías. Además se utilizará el segundo módulo para identificar técnicas en los registros de tráfico y poder aplicar la caracterización de ataques en la gestión de riesgos, evaluando además el funcionamiento del sistema interoperable con la metodología CRAMM.

10.2 Caso de uso 1: Evaluación de la caracterización de ciberataques

Como ya se ha introducido, esta simulación trata de analizar el valor de la caracterización de ciberataques. Para ello, el escenario estará definido por los activos presentados en la Tabla 10.1. Además, para poder aislar y evaluar el riesgo y el efecto de la caracterización, el catálogo de amenazas previas está vacío y la información del sistema se presenta en la Tabla 10.2, indicando el nivel de aceptación de riesgos, que será bajo (B), ya que la metodología en la que se definen los datos de entrada es MAGERIT. A partir de aquí, se plantean dos posibles situaciones, según si tiene lugar o no la caracterización de ciberataques.

Tabla 10.2: Información del sistema. Validación - Caso de uso 1

Campo	Valor
Identificador	Validación
Nivel de aceptación del riesgo	B
Metodología de entrada	MAGERIT

10.2.1 No se produce la caracterización de ciberataques

Esta condición se da cuando no se activa el módulo de identificación de técnicas, por lo que el tráfico generado por los sensores se analiza para identificar anomalías y se incluye directamente en la ontología para la gestión de riesgos. Además de los catálogos presentados hasta ahora, se tendrán en cuenta un conjunto de escenarios de riesgo y mitigaciones. Para facilitar el ejemplo no se describen las vulnerabilidades de los activos, ya que principalmente se usan en la caracterización, por lo que se incluirán en la siguiente situación.

En primer lugar, los incidentes que se identifican mediante el IDS de sensores heterogéneos se reflejan en la ontología con los siguientes datos (Tabla 10.3). Se van a analizar anomalías detectadas por los sensores de Wi-Fi, *Bluetooth* y ciberseguridad, que son los que mejor se adaptan al entorno.

Tabla 10.3: Incidentes registrados. Validación - Caso de uso 1.1

Identificador	Sensor	Fecha	Activo atacado
Incidente_1	Wi-Fi	10/02/2024 13:49	PC
Incidente_2	Wi-Fi	10/02/2024 05:17	Móvil
Incidente_3	Wi-Fi	10/02/2024 19:26	PC
Incidente_4	<i>Bluetooth</i>	25/03/2024 02:14	Móvil
Incidente_5	<i>Bluetooth</i>	25/03/2024 10:56	Móvil
Incidente_6	Ciberseguridad	04/04/2024 11:03	PC
Incidente_7	Ciberseguridad	04/04/2024 15:26	PC
Incidente_8	Ciberseguridad	04/04/2024 02:45	PC
Incidente_9	Ciberseguridad	04/04/2024 22:19	PC
Incidente_10	Ciberseguridad	04/04/2024 23:36	PC
Incidente_11	Ciberseguridad	04/04/2024 03:46	PC

Estos incidentes se comparan con los escenarios de riesgo que se han identificado previamente tras analizar el entorno (Tabla 10.4). Se ha definido uno por cada tipo de sensor y activo que afecta, y un caso por si el incidente no cuadra con ninguno (anomalías Wi-Fi que afectan a los dispositivos móviles).

Tabla 10.4: Escenarios de riesgo. Validación - Caso de uso 1.1

Identificador	Amenaza	Probabilidad	Impacto	Activo afectado
Escenario Wi-Fi	Escucha	B	MA	PC
Escenario <i>Bluetooth</i>	DoS	B	A	Móvil
Escenario ciberseguridad	Difusión de SW dañino	MA	MB	PC
Desconocido	<i>Unknown</i>	M	MA	Cualquiera

Los incidentes presentados y los escenarios definidos ponen al sistema bajo un nivel de riesgo determinado. La ontología interoperable realiza el cálculo para la metodología MAGERIT e ITSRM, obteniendo los resultados de la Tabla 10.5. A pesar de que el nivel de cada tipo de amenaza no coincide entre las dos metodologías, el riesgo global sí coincide. La diferencia principal es que el orden de prioridad de las amenazas según MAGERIT sería '*Eavesdropping*, *Unknown* - DoS - *Deliberated Malware Diffusion*', mientras que para ITSRM es prioritario el riesgo desconocido, y los otros tres tipos están igualados. Esta situación será crítica a la hora de mitigar el riesgo.

Tabla 10.5: Cálculos de riesgo potencial. Validación - Caso de uso 1.1

Riesgo	Nivel MAGERIT	Nivel ITSRM
<i>Unknown</i>	MA	H
<i>Deliberated Malware Diffusion</i>	B	M
DoS	A	M
<i>Eavesdropping</i>	MA	M
Riesgo Global	A	H

En la Figura 10.2 se muestra el cálculo de riesgo global en ambos casos.

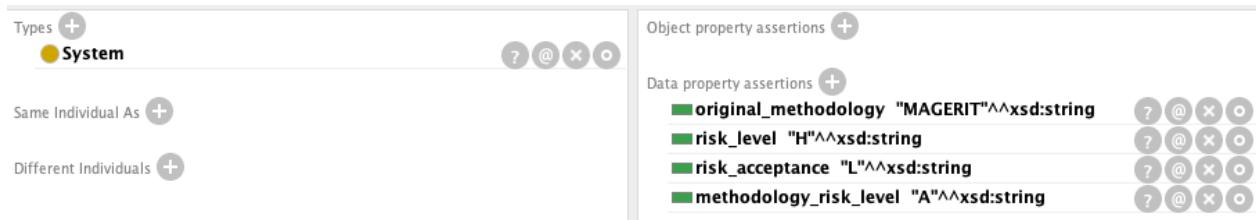


Figura 10.2: Nivel de riesgo potencial global del sistema. Validación - Caso de uso 1.1

En este momento se analizan las mitigaciones definidas en el sistema (Tabla 10.6) y se eligen las más adecuadas para responder al incidente. Teniendo en cuenta que son contramedidas genéricas, su eficacia es menor. En la situación planteada, todas las amenazas tienen una mitigación posible. El sistema las asocia y luego descarta la mitigación 4 (*Use of External Information Systems*) para responder al ataque de *Malware Diffusion* ya que el impacto de la contramedida (MA) es mayor que el nivel de riesgo asociado a esa amenaza (M). DoS y *Eavesdropping* únicamente tienen una mitigación disponible (*Non-modifiable executable programmes* para DoS y *Remote Access* para la escucha), que se selecciona, mientras que las desconocidas tienen dos (*Information Sharing* y *Security awareness training*) y, según la puntuación que les asigna el algoritmo, la mitigación 2 (*Security awareness training*) es más adecuada para hacer frente a este riesgo.

Tabla 10.6: Catálogo de mitigaciones. Validación - Caso de uso 1.1

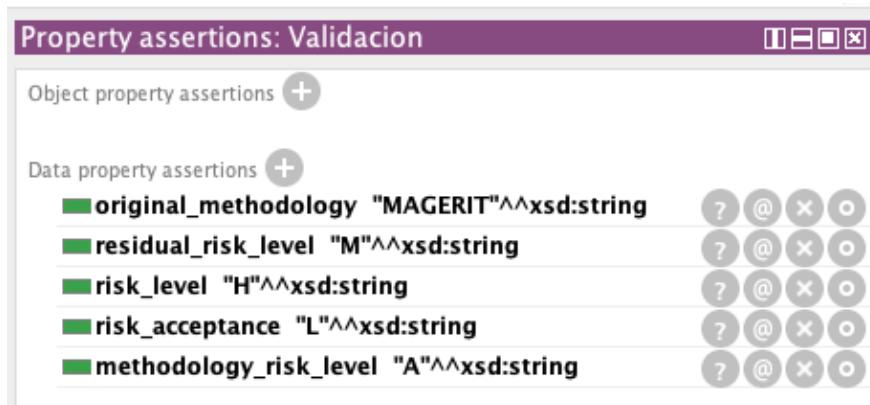
Identificador	Palabras clave	Coste	Esfuerzo	Impacto	Complejidad de instalación	Complejidad de operación	Factor de mitigación	Eficacia	Tiempo
Mitigación 1	DoS	340	8	B	6	1	0.8	0.3	500
Mitigación 2	Unknown	10000	7	MB	1	1	0.6	0.5	365
Mitigación 3	Eavesdropping	128	9	M	8	6	0.7	0.6	30
Mitigación 4	Malware_Diffusion	100	5	MA	7	2	0.6	0.2	75
Mitigación 5	Unknown	12054	9	MB	1	1	0.4	0.6	365

Tras este análisis, se plantean contramedidas para hacer frente a las amenazas de DoS, *Eavesdropping* y desconocidas. Dado que ninguno de los riesgos potenciales estimados para ellas (Tabla 10.5) es menor que el nivel de aceptación definido para el sistema (Tabla 10.2), las tres contramedidas se aplican para reducir el riesgo, obteniendo los siguientes riesgos residuales (Tabla 10.7).

Tabla 10.7: Cálculos de riesgo residual. Validación - Caso de uso 1.1

Riesgo	Potencial	Residual
<i>Unknown</i>	H	H
<i>Deliberated Malware Diffusion</i>	M	-
DoS	M	M
<i>Eavesdropping</i>	M	L
Riesgo Global	H	M

A pesar de que los dos primeros riesgos no se reducen tras la aplicación de las contramedidas, los riesgos residuales obtenidos a raíz del efecto de las mitigaciones genéricas consigue reducir en un nivel el riesgo global del sistema (de alto (H) a medio (M)). Como se muestra en la Figura 10.3, el riesgo según la metodología MAGERIT no se mitiga, tal y como se presentó en el Capítulo 9.

**Figura 10.3:** Nivel de riesgo residual global del sistema. Validación - Caso de uso 1.1

10.2.2 Se produce la caracterización de ciberataques

La propuesta de esta Tesis Doctoral se basa en la eficacia de la caracterización de ciberataques para complementar procesos de ciberseguridad y facilitar la labor de los analistas. Para verificarlo, sobre el caso anterior, el tráfico de red atraviesa el sistema de identificación de técnicas que permite extraer información para enriquecer la gestión de riesgos.

En esta ocasión, se produce la identificación manual de vulnerabilidades relacionadas con los activos definidos en el escenario (Tabla 10.1), para poder establecer relaciones con las técnicas de los ciberataques. Las vulnerabilidades identificadas se presentan en la Tabla 10.8.

Tabla 10.8: Catálogo de vulnerabilidades. Validación - Caso de uso 1.2

Identificador	CWE	CVSS	Activo
CVE-2022-29527	CWE-732	<i>High</i>	PC
CVE-2017-7440	CWE-1021	<i>Medium</i>	PC
CVE-2019-6260	CWE-1317	<i>Critical</i>	PC
CVE-2019-10482	CWE-208	<i>Medium</i>	Móvil

Además, se tienen en cuenta los incidentes de ciberseguridad procedentes de fuentes heterogéneas que detecta el IDS (no se consideran amenazas previas). De la Tabla 10.3 se mantienen los incidentes del sensor Wi-Fi y *Bluetooth*, mientras que los incidentes de ciberseguridad atraviesan el módulo de caracterización de técnicas y, además de la información presentada en esa tabla, se indica la técnica utilizada, los posibles patrones de ataque y mitigaciones con las que responder al incidente (Tabla 10.9).

Tabla 10.9: Incidentes caracterizados. Validación - Caso de uso 1.2

Identificador	Sensor	Fecha	Activo atacado	Técnica	Patrón de ataque	Mitigaciones
Incidente_1	Wi-Fi	10/02/2024 13:49	PC	-	-	-
Incidente_2	Wi-Fi	10/02/2024 05:17	Móvil	-	-	-
Incidente_3	Wi-Fi	10/02/2024 19:26	PC	-	-	-
Incidente_4	<i>Bluetooth</i>	25/03/2024 02:14	Móvil	-	-	-
Incidente_5	<i>Bluetooth</i>	25/03/2024 10:56	Móvil	-	-	-
Incidente_6	Ciberseguridad	04/04/2024 11:03	PC	T1071	-	M1031
Incidente_7	Ciberseguridad	04/04/2024 15:26	PC	T1548	CAPEC-114, CAPEC-115, CAPEC-122, CAPEC-233, CAPEC-654	M1047, M1038, M1026, M1022, M1052, M1018
Incidente_8	Ciberseguridad	04/04/2024 02:45	PC	T1548	CAPEC-114, CAPEC-115, CAPEC-122, CAPEC-233, CAPEC-654	M1047, M1038, M1026, M1022, M1052, M1018
Incidente_9	Ciberseguridad	04/04/2024 22:19	PC	T1548	CAPEC-114, CAPEC-115, CAPEC-122, CAPEC-233, CAPEC-654	M1047, M1038, M1026, M1022, M1052, M1018
Incidente_10	Ciberseguridad	04/04/2024 23:36	PC	T1190	-	M1048, M1050, M1030, M1026, M1051, M1016
Incidente_11	Ciberseguridad	04/04/2024 03:46	Móvil	T1592	CAPEC-169, CAPEC-541	M1056

La caracterización de esos ciberataques permite definir escenarios de riesgo más precisos que los presentados en la Tabla 10.4, ya que ahora pueden depender de los patrones, debilidades o técnicas identificadas. Los escenarios establecidos para esta situación son los siguientes (Tabla

10.10), manteniendo los de los sensores Wi-Fi y *Bluetooth* y segmentando el de ciberseguridad en distintos casos:

Tabla 10.10: Escenarios de riesgo. Validación - Caso de uso 1.2

Identificador	Amenaza	Probabilidad	Impacto	Activo afectado	Patrón de ataque	Vulnerabilidad
Escenario Wi-Fi	Escucha	B	MA	PC	-	-
Escenario <i>Bluetooth</i>	DoS	B	A	Móvil	-	-
Escenario ciberseguridad 1	Acceso no autorizado	MA	MB	PC	CAPEC-122	CVE-2022-2952
Escenario ciberseguridad 2	Destrucción accidental de información	M	B	PC	CAPEC-654	CVE-2017-7440
Escenario ciberseguridad 3	Alteración de secuencia	MB	MA	PC	CAPEC-122	CVE-2019-6260
Escenario ciberseguridad 4	Manipulación de equipos	A	A	Móvil	CAPEC-541	CVE-2019-10482
Desconocido	<i>Unknown</i>	M	MA	Cualquiera	-	-

Este análisis es más exhaustivo, definiendo distintos tipos de ciberataques para un mismo sensor, y clasificando como amenazas desconocidas las que no coinciden con los patrones anteriores. Además considera amenazas por error del usuario, que tengan origen en una acción deliberada del atacante.

Al llevar a cabo en la ontología el cálculo del estado de riesgo del sistema según la simulación planteada, la situación difiere en gran medida de la presentada en la Tabla 10.5. Sin la caracterización de ataques el riesgo global del sistema era alto (A/H) en ambos casos, considerándose cuatro tipos de amenazas (*Unknown*, *Deliberated Malware Diffusion*, *Eavesdropping* y DoS), cuyos niveles no coincidían según el mapa de ambos marcos. Al caracterizar los ciberataques, el riesgo de las anomalías Wi-Fi o *Bluetooth* debería ser el mismo, sin embargo, aparecen nuevas amenazas (*HW Tampering*, *Sequence Alteration*, *Unauthorised Access* y *Unintentional Destruction of Information*). Ahora los niveles globales ya no coinciden, siendo mayor el obtenido por MAGERIT (Figura 10.4).

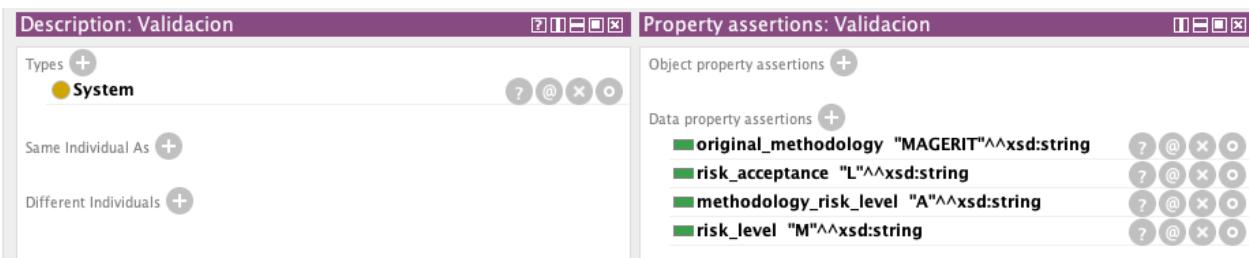


Figura 10.4: Nivel de riesgo potencial global del sistema. Validación - Caso de uso 1.2

En la Tabla 10.11 se recogen los resultados obtenidos. Prestando atención a la prioridad de las amenazas, los resultados que se obtuvieron sin la caracterización fueron los siguientes:

- Metodología MAGERIT: *Eavesdropping*, Desconocidos - DoS - *Deliberated Malware Diffusion*.

- Metodología ITSRM: Desconocidos - *Eavesdropping*, DoS, *Deliberated Malware Diffusion*.

Sin embargo, mediante la identificación de técnicas, se señala como diferencia principal, además del número de entradas, que una de las amenazas detectadas por el sensor de ciberseguridad, que en el caso anterior tenía la prioridad más baja, debe ser antepuesto a otros:

- Metodología MAGERIT: *Eavesdropping*, Desconocidos, *HW Tampering* - DoS, *Sequence Alteration* - *Unauthorised Access*, *Unintentional Destruction of Information*.
- Metodología ITSRM: Desconocidos, *HW Tampering* - *Sequence Alteration*, *Unauthorised Access*, *Unintentional Destruction of Information*, DoS, *Eavesdropping*.

Tabla 10.11: Cálculos de riesgo potencial. Validación - Caso de uso 1.2

Riesgo	Nivel MAGERIT	Nivel ITSRM
<i>Unknown</i>	MA	H
<i>HW Tampering</i>	MA	H
<i>Sequence Alteration</i>	A	M
<i>Unauthorised Access</i>	B	M
<i>Unintentional Destruction of Information</i>	B	M
DoS	A	M
<i>Eavesdropping</i>	MA	M
Riesgo Global	A	M

Con estos resultados, se procede a elegir las respuestas óptimas ante los ciberataques recibidos. El catálogo de contramedidas (Tabla 10.6) se completa según la información obtenida en la caracterización, resultando en las propuestas presentadas en la Tabla 10.12.

A través del algoritmo se plantea la siguiente asignación:

- Amenazas desconocidas: Además de las dos mitigaciones que se plantearon inicialmente (*Security awareness training* e *Information Sharing*), a través de la identificación de técnicas también se puede hacer frente a este tipo de amenazas con mitigaciones específicas del TTP, como M1031, M1048, M1050, M1030, M1051, M1016 y M1026. El riesgo potencial asignado según ITSRM fue alto (H), por lo que de estas contramedidas, únicamente se descarta la M1030, que tiene un impacto mayor. Las demás se ordenan, seleccionando como óptima la M1050 - *Exploit Protection*, con la que se calcula el riesgo residual, pero manteniendo el resto como respuestas recomendadas.

- DoS: En este caso, se mantiene la mitigación 1 (*Non-modifiable executable programmes*), recomendada sin la caracterización de ataques ya que, en esta simulación, este ataque se lleva a cabo a través de señales *Bluetooth*, que no se analizan.
- Amenaza de *Eavesdropping*: Como en el caso anterior, se mantiene la misma respuesta (mitigación 3, *Remote Access*), ya que las señales Wi-Fi a partir de las que se identifica esta amenaza tampoco se caracterizan.
- Amenaza de *HW Tampering*: Esta amenaza, debido a la técnica identificada, únicamente tiene una contramedida posible, M1056 - *Pre-compromise*, que es la identificada como óptima.
- Amenaza de *Unauthorised Access*: Tras la modificación de las palabras clave de la mitigación 4 (*Use of External Information Systems*), ésta se plantea como posible respuesta al riesgo junto a las mitigaciones M1047, M1038, M1028, M1026 y M1018. Entre todas estas recomendaciones, se elige como óptima la M1047 - *Audit*.
- Amenaza de *Sequence Alteration*: Para ese caso, únicamente se proponen medidas extraídas de la caracterización, como M1047, M1038, M1028, M1026 y M1018, siendo esta última elegida como óptima.
- Amenaza de *Unintentional Destruction of Information*: Esta amenaza también se incluye como palabra clave en la mitigación 4, recomendándose además a raíz de la caracterización las mitigaciones M1026, M1018, M1022, M1052. La elección óptima, como en el caso anterior, es M1018 - *User account management*.

Tabla 10.12: Catálogo de mitigaciones. Validación - Caso de uso 1.2

Identificador	Palabras clave	Coste	Esfuerzo	Impacto	Complejidad de instalación	Complejidad de operación	Factor de mitigación	Eficacia	Tiempo
Mitigación 1	DoS	340	8	B	6	1	0.8	0.3	500
Mitigación 2	Unknown	10000	7	MB	1	1	0.6	0.5	365
Mitigación 3	Eavesdropping	128	9	M	8	6	0.7	0.6	30
Mitigación 4	Unauthorised_Access Destruction_Info	100	5	MA	7	2	0.6	0.2	75
Mitigación 5	Unknown	12054	9	MB	1	1	0.4	0.6	365
M1047	T1548	100000	2	B	2	2	0.8	1	365
M1038	T1548	3427	7	A	4	8	0.74	0.81	52
M1028	T1548	3470	8	B	9	6	0.7	0.95	41
M1026	T1548, T1190	3000	6	B	7	5	0.5	0.7	45
M1022	T1548	10	7	M	6	9	0.75	0.87	30

Tabla 10.12: Catálogo de mitigaciones. Validación - Caso de uso 1.2

Identificador	Palabras clave	Coste	Esfuerzo	Impacto	Complejidad de instalación	Complejidad de operación	Factor de mitigación	Eficacia	Tiempo
M1052	T1548	3	5	B	8	7	0.6	0.7	27
M1018	T1548	3	2	MB	7	4	0.6	0.7	15
M1056	T1592	12	9	MB	4	5	0.6	1	365
M1031	T1071	50000	9	M	9	3	0.9	1	600
M1048	T1190	25000	8	M	8	2	0.7	0.9	127
M1050	T1190	70	5	M	3	2	0.9	1	32
M1030	T1190	1000	10	MA	10	1	0.85	1	1000
M1051	T1190	20	3	B	5	3	0.7	1	365
M1016	T1190	5000	8	A	7	1	0.8	1	600

Como ninguno de los riesgos potenciales estimados es menor que el nivel de aceptación presentado en la Tabla 10.2, todas las amenazas se mitigan. Los riesgos residuales finales se presentan en la Tabla 10.13.

Tabla 10.13: Cálculos de riesgo residual. Validación - Caso de uso 1.2

Riesgo	Potencial	Residual
<i>Unknown</i>	H	L
<i>HW Tampering</i>	H	M
<i>Sequence Alteration</i>	M	L
<i>Unauthorised Access</i>	M	L
<i>Unintentional Destruction of Information</i>	M	L
<i>DoS</i>	M	M
<i>Eavesdropping</i>	M	L
Riesgo Global	M	L

La caracterización de ciberataques permite mitigar en dos niveles el riesgo calculado para las amenazas desconocidas, que es una de las principales preocupaciones de los analistas de

seguridad hoy en día. Los riesgos asociados a DoS y *Eavesdropping* no varían de un caso al otro, ya que no les influye la detección de técnicas. La mitigación de las amenazas asociadas a los patrones de ataque es exitosa. En conjunto, el nivel total se reduce en un nivel de la escala, tal y como se muestra en la Figura 10.5.

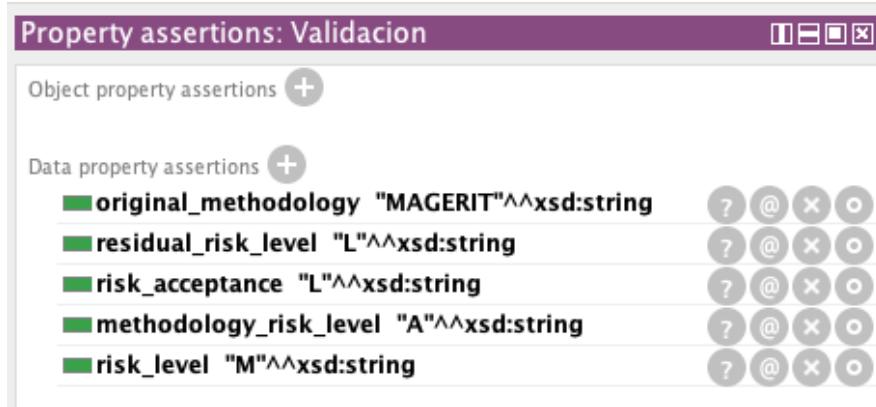


Figura 10.5: Nivel de riesgo residual global del sistema. Validación - Caso de uso 1.2

En conjunto, la recomendación de contramedidas a partir de las técnicas identificadas en la caracterización de ciberataques permite una gestión de riesgos más adaptada a los incidentes y que es capaz de mitigar el riesgo de forma adecuada, validándose la Hipótesis 2 de esta Tesis Doctoral.

10.3 Caso de uso 2: Evaluación del entorno de conciencia cibersituacional

En este segundo caso de uso, el objetivo de la simulación es alejarse de demostrar la eficacia de la caracterización de ciberataques para analizar y validar el sistema completo definido en esta Tesis Doctoral. Hasta el momento, se ha comprobado la funcionalidad individual de cada módulo (IDS, sistema de caracterización de técnicas y ontología para la gestión dinámica de riesgos), obteniendo resultados válidos con casos de uso independientes.

Para comprobar el funcionamiento global del entorno, se plantea una simulación que sigue la creación de unos datos desde los sensores físicos de Wi-Fi, *Bluetooth* y el sensor software de tráfico (que forman parte del conjunto de activos presentado en la Figura 10.1) a través de los distintos módulos definidos, con el objetivo de evaluar la gestión global y los resultados proporcionados por el sistema en conjunto a la conciencia cibersituacional. En este escenario se parte del sistema “en blanco” para que no afecte ninguna configuración al resultado final, aunque los modelos del IDS están entrenados con el comportamiento normal del sistema y los que realizan la identificación de técnicas, con el conjunto de datos escogido. La ontología está construida y su comportamiento definido, pero no existen datos previos almacenados en el sistema.

10.3.1 Generación de datos heterogéneos

En primer lugar, los sensores Wi-Fi y *Bluetooth* capturan toda la información relativa a estas conexiones que se detecta en el entorno. Por otra parte, el módulo Zeek, que captura y complementa los datos de tráfico de red, también proporciona información para analizar en la simulación global. A continuación se presentarán los datos capturados a través de cada fuente.

Dispositivos Wi-Fi

Este sensor captura información sobre cualquier dispositivo que tenga tecnología Wi-Fi desarrollada y aparezca en el radio de detección del sensor. Para la simulación se van a utilizar los siguientes datos (Tabla 10.14), con el formato descrito en la Tabla 7.5, correspondientes a portátiles y teléfonos que podrían estar o no entre los activos de cualquier organización.

Tabla 10.14: Registros Wi-Fi capturados - Caso de uso 2

Campos	Registro 1	Registro 2	Registro 3	Registro 4
<i>Time</i>	1711034763	1711277204	1711355929	1711595567
userid	2A:4F:5E:7D:9B:1C	8F:3B:6A:0D:5E:2C	B1:7C:9A:4E:6F:2D	B1:7C:9A:4E:6F:2D
<i>footprint</i>	69:BF:70:D6:AC:84	33:0C:C5:2D:6A:80	1A:E3:34:30:F8:B8	0C:E2:6D:24:C5:1D
tseen	10800	900	18000	300
tacum	10800	900	18000	300
<i>visits</i>	0	0	0	1
pwr	-61	-91	-77	-87
tx_packets	8	1	1	1
tx_bytes	1118	91	135	121
rx_packets	1	0	0	0
rx_bytes	133	0	0	0
apwr	-44	-1	-1	-1
<i>type</i>	MAL	LMA	MAL	MAL

Son tres dispositivos localizados a lo largo de una semana a distintos horarios. En el módulo IDS se analizarán sus características para determinar si corresponden o no con posibles ciberataques.

Dispositivos *Bluetooth*

De forma similar, el sensor *Bluetooth* es capaz de detectar todos los dispositivos en su radio de acción. En la simulación, los dispositivos conectados han sido los siguientes (Tabla 10.15) y se analizarán en el próximo módulo.

Tabla 10.15: Registros *Bluetooth* capturados - Caso de uso 2

Campos	Registro 1	Registro 2	Registro 3
<i>Time</i>	25/03/2024 09:03:06	25/03/2024 12:43:28	26/03/2024 02:49:31
<i>status</i>	Conectado	Conectado	Conectado
<i>classic_mode</i>	Sí	Sí	Sí
<i>le_mode</i>	No	No	No
<i>lmp_version</i>	-	-	-
<i>address</i>	3C:8A:2F:6D:9B:1E	A7:2B:4F:5E:1D:8C	5F:9D:0A:3B:7C:2E

Información de tráfico de red

Para obtener registros de los incidentes de tráfico, el sistema captura información y se complementa mediante Zeek, para obtener las columnas con las que se entrenó el sistema de identificación de técnicas, como se recoge en la Tabla 10.16.

Tabla 10.16: Registros de tráfico de red capturados - Caso de uso 2

Campos	Registro 1	Registro 2	Registro 3	Registro 4
<i>ts</i>	1664067292.732	1640830086.632	1662241013.164	1663273273.117
<i>datetime</i>	1709903567	1709212367	1712322767	1663273272
<i>orig_ip_bytes</i>	1529	158	192	474
<i>resp_ip_bytes</i>	1312	0	327	1223
<i>resp_bytes</i>	940	0	243	955
<i>duration</i>	0.31130599	1.40666961e-05	0.00043702	0.16599108
<i>orig_bytes</i>	897	102	108	154
<i>orig_pkts</i>	12	2	3	6
<i>history</i>	13	0	1	11
<i>resp_pkts</i>	7	0	3	5
<i>conn_state</i>	4	2	4	4

10.3.2 Identificación de anomalías Wi-Fi y *Bluetooth*

Los registros capturados en estos sensores se analizan utilizando los modelos entrenados y probados como se desarrolló en el Capítulo 7. Los datos se pre-procesan correctamente según su formato antes de atravesar los modelos.

Anomalías Wi-Fi

El módulo de detección de anomalías Wi-Fi compara los nuevos registros con el comportamiento “normal” del sistema. De los cuatro registros capturados (Tabla 10.14), el análisis obtenido es el siguiente:

- El **Registro 1** es un dispositivo que se reconoció durante el entrenamiento, por lo que es habitual. Además, el horario de detección es dentro de la jornada laboral y entre semana, por lo que **no** se considera **anomalía**.
- El **Registro 2**, aunque se recibe en horario laboral, corresponde a un fin de semana, por lo que **sí** se considera **anómalo**.
- El **Registro 3** corresponde a un dispositivo que no se había identificado anteriormente, y por tanto se marca como **anómalo** aunque el sello temporal sea correcto.
- El **Registro 4** se refiere al mismo dispositivo pero realizando una conexión fuera del horario laboral, por lo que también es **anómalo**.

Aunque el tercer registro no represente una amenaza en sí mismo (podría tratarse de un trabajador nuevo), al no conocerse en el sistema, la configuración lo detecta como posible incidente.

Anomalías *Bluetooth*

Similar al caso anterior, este sub-sistema compara los registros obtenidos (Tabla 10.15) con el comportamiento estándar del tráfico del sensor, obteniendo así los siguientes resultados:

- El **Registro 1** corresponde a un dispositivo que ya se había visto, por lo que **no** se considera **anómalo**.
- El **Registro 2** es un dispositivo nuevo, que se considera **anomalía** según el algoritmo entrenado.
- Como en el caso anterior, el **Registro 3** representa tráfico a horas poco comunes, que también se considera **anomalía**.

De nuevo, el hecho de que los dispositivos no reconocidos se consideren un posible ciberataque es una medida de seguridad, ya que el modelo no es capaz de diferenciar si el activo pertenece a un empleado o a un atacante. Los falsos positivos en este caso pueden dispararse, ya que con el desarrollo de IoT, cada vez más dispositivos llevan *Bluetooth* incorporado.

10.3.3 Caracterización de técnicas en registros de tráfico

El tráfico de red que se captura y enriquece mediante Zeek, tras un pre-procesado más sencillo que en el caso de los datos de sensores físicos, se analiza para identificar si representan o no tráfico normal, y en caso negativo, clasificarlos según la técnica más probable que puedan utilizar, como se presentó en el Capítulo 8.

Como estos modelos se basan en un *dataset* de entrenamiento, no es sencillo explicar el motivo de su decisión, como ocurría en los casos anteriores. Por esta razón, en la Tabla 10.17 únicamente se presenta el resultado de esta caracterización.

Tabla 10.17: Caracterización de registros de tráfico capturados - Caso de uso 2

Caracterización	Registro 1	Registro 2	Registro 3	Registro 4
Clasificación binaria	Ciberataque	Tráfico normal	Ciberataque	Ciberataque
Táctica	<i>Discovery</i>	-	<i>Reconnaissance</i>	<i>Reconnaissance</i>
Técnica	T1046	-	T1590	T1595
Patrones de ataque	CAPEC - 300	-	CAPEC - 309	CAPEC - 169
Mitigaciones	M1042, M1031, M1030	-	M1056	M1056

10.3.4 Ontología y gestión dinámica de riesgos

Tras la identificación de las posibles anomalías y ciberataques, se procede al análisis y gestión de los riesgos que suponen estos incidentes para el sistema. Para ello, como en casos de uso anteriores, se puebla la ontología con el siguiente orden.

En primer lugar, se crea una instancia que representa el sistema, donde se almacene información propia de entrada y la que se genere a lo largo del proceso, como los niveles de riesgo global. Inicialmente, contendrá la información recogida en la Tabla 10.18. Para este caso de uso se utilizará la metodología CRAMM, que aún no había sido validada y presenta la mayor diferencia con respecto a ITSRM de entre todas las consideradas, ya que su aproximación para calcular el riesgo es distinta, se centra en los activos afectados en lugar de en las amenazas en sí mismas.

Tabla 10.18: Información del sistema. Validación - Caso de uso 2

Campo	Valor
Identificador	Validación_2
Nivel de aceptación del riesgo	L
Metodología de entrada	CRAMM

En cuanto a los activos, en comparación con la información proporcionada en la Figura 10.1, se tiene en cuenta que en la caracterización de los ciberataques de sensores físicos se han identificado varios dispositivos (al menos tres distintos) y se tiene en cuenta la taxonomía definida en CRAMM para clasificarlos en datos, aplicaciones y activos físicos. Los activos sobre los que se lleva a cabo la evaluación de los riesgos se presentan en la Tabla 10.19, incluyendo la valoración de los activos físicos, de la que se extrae la de los datos y aplicaciones.

Tabla 10.19: Catálogo de activos. Validación - Caso de uso 2

Activo	Tipo	C	I	A
Sensor de Wi-Fi	Activo físico	8	5	9
Datos Wi-Fi	Datos	5.67	5.34	7
Sensor de <i>Bluetooth</i>	Activo físico	9	7	7
Datos <i>Bluetooth</i>	Datos	6	6	6.34
Sensor de Ciberseguridad	Aplicación	8	6	7
Datos Ciberseguridad	Datos	5.67	5.67	6.34
Servidor	Activo físico	7	8	6
Móvil 1	Activo físico	8	8	9
Móvil 2	Activo físico	6	7	9
PC 1	Activo físico	9	9	10
PC 2	Activo Físico	5	7	4
Datos críticos	Datos	5.75	6.75	6.5
Correo electrónico	Aplicación	7	7.8	7.6

Tras un estudio exhaustivo, se han identificado algunas vulnerabilidades que pueden ser explotadas en estos activos, recogidas en la Tabla 10.20.

Tabla 10.20: Catálogo de vulnerabilidades. Validación - Caso de uso 2

Identificador	CWE	CVSS	Activo
CVE-2021-25476	CWE-200	<i>Medium</i>	Móvil 1, Móvil 2
CVE-2022-0708	CWE-200	<i>Medium</i>	Email, PC 1, PC 2
CVE-2022-31162	CWE-200	<i>High</i>	Servidor

El listado inicial de amenazas comienza vacío, pero se definen un conjunto de escenarios de riesgo basados en la información que se puede extraer de la caracterización de los ataques:

- Si el incidente procede de una anomalía de los sensores Wi-Fi o *Bluetooth*, puede darse por estar fuera del horario laboral o porque el sensor no ha visto anteriormente el dispositivo. En función de esto, se definen dos escenarios de riesgo por cada tipo, ya que no tienen la misma probabilidad e impacto.
- A través de la información extraída de los registros de tráfico, se pueden definir escenarios más concretos de ciberataques a través de la red, como se mostró en el caso de uso anterior.
- Es imprescindible definir un escenario en caso de que el incidente no encaje con ninguno de los anteriores. Los ataques desconocidos, como pueden ser los de día cero, son una de las principales amenazas de los sistemas en la actualidad.

Con esta información, los escenarios de riesgo establecidos son los siguientes (Tabla 10.21):

Tabla 10.21: Escenarios de riesgo. Validación - Caso de uso 2

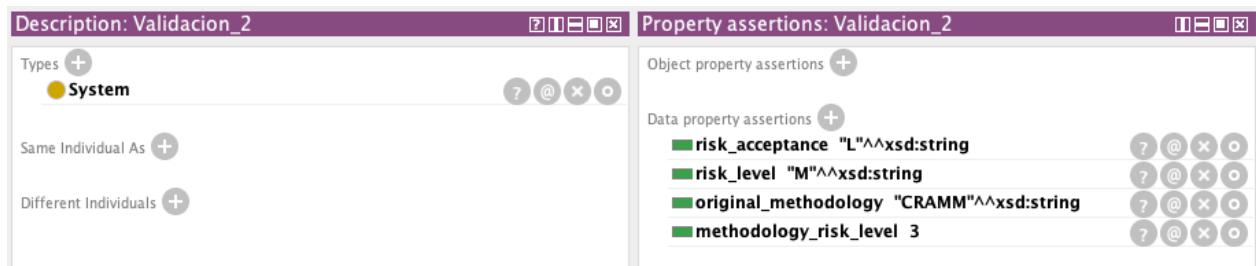
Identificador	Amenaza	Probabilidad	Impacto	Activo afectado	Patrón de ataque	Vulnerabilidad
Escenario WF	Suplantación de identidad	VH	L	PCs, Móviles	-	-
Escenario WF temporal	Escucha	L	H	PCs, Móviles	-	-
Escenario BT	Suplantación de identidad	VH	L	PCs, Móviles	-	-
Escenario BT temporal	DoS	L	H	PCs, Móviles	-	-
Escenario CS 1	Divulgación de información	H	M	PC	CAPEC-300	CVE-2021-25476
Escenario CS 2	Abuso de privilegios	M	M	PC	CAPEC-309	CVE-2022-0708
Escenario CS 3	Alteración de información	M	VH	PC	CAPEC-169	CVE-2022-31162
Desconocido	<i>Unknown</i>	H	VH	Cualquiera	-	-

Cuando se reciben en la ontología los incidentes identificados por el IDS y el sistema de caracterización de técnicas (Tabla 10.22), éstos se comparan con los escenarios de riesgo y se crean las amenazas correspondientes. A través de la fecha y el sensor, el sistema debe ser capaz de distinguir los escenarios de riesgo producidos por nuevos dispositivos de los temporales.

Tabla 10.22: Incidentes registrados. Validación - Caso de uso 2

Identificador	Sensor	Fecha	Activo atacado	Técnica	Patrón de ataque	Mitigaciones
Incidente_1	Wi-Fi (Temporal)	D 24/03/2024 10:46	Móvil 2	-	-	-
Incidente_2	Wi-Fi	L 25/03/2024 08:38	PC2	-	-	-
Incidente_3	Wi-Fi (Temporal)	J 28/03/2024 03:12	PC2	-	-	-
Incidente_4	Bluetooth	L 25/03/2024 12:43	Móvil 2	-	-	-
Incidente_5	Bluetooth (Temporal)	M 26/03/2024 02:49	PC1	-	-	-
Incidente_6	Ciberseguridad	V 08/03/2024 14:12	Móvil 1	T1046	CAPEC-300	M1042, M1031, M1030
Incidente_7	Ciberseguridad	V 05/04/2024 13:26	PC1	T1590	CAPEC-309	M1056
Incidente_8	Ciberseguridad	V 05/04/2024 22:45	Servidor	T1595	CAPEC-169	M1056

En este caso, todos los incidentes coinciden con algún escenario de riesgo, por lo que no aparecen las amenazas desconocidas. Como la metodología CRAMM realiza el cálculo del riesgo a través de activos, y los activos atacados son activos físicos, el riesgo asociado a esta categoría para los ciberataques registrados será de 3/7, que es aproximadamente un nivel medio-bajo, teniendo en cuenta el mapa de calor que utiliza este marco (Figura 9.9). Sin embargo, según la metodología ITSRM, que actúa de estándar, es un riesgo medio (M). Ambos datos se pueden observar en la Figura 10.6 correspondiente a la ontología en este punto.

**Figura 10.6:** Nivel de riesgo potencial global del sistema. Validación - Caso de uso 2

Siguiendo el estándar ITSRM, sí que se puede observar el nivel de riesgo potencial de cada tipo de amenaza (Tabla 10.23).

Tabla 10.23: Cálculos de riesgo potencial. Validación - Caso de uso 2

Riesgo	Nivel ITSRM
<i>Abuse of Access Privileges</i>	M
<i>Deliberated Alteration of Information</i>	H
<i>Disclosure of Information</i>	H

Tabla 10.23: Cálculos de riesgo potencial. Validación - Caso de uso 2

Riesgo	Nivel ITSRM
DoS	M
<i>Eavesdropping</i>	M
<i>Masquerading of Identity</i>	M
Riesgo Global	M

Por último, aunque el nivel de riesgo no es excesivamente preocupante, es imprescindible evaluar las respuestas disponibles frente a los ciberataques, para poder mitigar este riesgo lo máximo posible. El catálogo de contramedidas disponible para este caso de uso es el siguiente (Tabla 10.24).

Tabla 10.24: Catálogo de mitigaciones. Validación - Caso de uso 2

Identificador	Palabras clave	Coste	Esfuerzo	Impacto	Complejidad de instalación	Complejidad de operación	Factor de mitigación	Eficacia	Tiempo
Mitigación 1	DoS	340	8	L	6	1	0.8	0.3	500
Mitigación 2	<i>Unknown</i>	10000	7	VL	1	1	0.6	0.5	365
Mitigación 3	<i>Eavesdropping</i>	128	9	M	8	6	0.7	0.6	30
Mitigación 4	<i>Unknown</i>	12054	9	VL	1	1	0.4	0.6	365
Mitigación 5	<i>Masquerading of Identity</i>	56	5	M	6	6	0.7	0.8	365
M1031	T1046	50000	9	M	9	3	0.9	1	600
M1030	T1046	1000	10	VH	10	1	0.85	1	1000
M1042	T1046	10	7	L	2	2	0.8	1	43
M1056	T1590, T1595	12	9	VL	4	5	0.6	1	365

Tras la ejecución del algoritmo de soporte a la toma de decisiones, el resultado de contramedidas identificadas y seleccionadas se presenta en la Tabla 10.25:

Tabla 10.25: Soporte a la toma de decisiones. Validación - Caso de uso 2

Riesgo	Contramedidas recomendadas	Contramedida óptima
<i>Abuse of Access Privileges</i>	M1056	M1056
<i>Deliberated Alteration of Information</i>	M1056	M1056
<i>Disclosure of Information</i>	M1030, M1031, M1042	M1042
DoS	Mitigación 1	Mitigación 1
<i>Eavesdropping</i>	Mitigación 3	Mitigación 3
<i>Masquerading of Identity</i>	Mitigación 5	Mitigación 5

Las mitigaciones destinadas a las amenazas desconocidas no se aplican y, dado el nivel de aceptación de riesgo del sistema (bajo), todas las contramedidas se despliegan, obteniendo los siguientes riesgos residuales (Tabla 10.26 y Figura 10.7).

Tabla 10.26: Niveles de riesgo residual. Validación - Caso de uso 2

Riesgo	Riesgo Potencial	Riesgo Residual
<i>Abuse of Access Privileges</i>	M	L
<i>Deliberated Alteration of Information</i>	H	M
<i>Disclosure of Information</i>	H	L
DoS	M	M
<i>Eavesdropping</i>	M	L
<i>Masquerading of Identity</i>	M	L
Riesgo Global	M	L

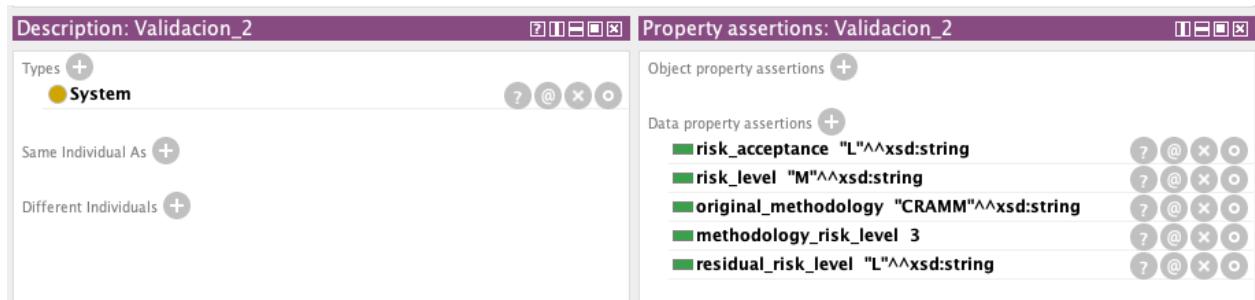


Figura 10.7: Nivel de riesgo residual global del sistema. Validación - Caso de uso 2

Si se realizase una modificación en la configuración del sistema, subiendo el nivel de aceptación del riesgo a alto (H), la situación cambiaría, ya que no todas las contramedidas se aplicarían. Las tablas anteriores se modificarían hasta obtener el siguiente resultado (Tablas 10.27 y 10.28 y Figura 10.8).

Tabla 10.27: Soporte a la toma de decisiones modificada. Validación - Caso de uso 2

Riesgo	Contramedidas recomendadas	Contramedida óptima
<i>Abuse of Access Privileges</i>	M1056	-
<i>Deliberated Alteration of Information</i>	M1056	M1056
<i>Disclosure of Information</i>	M1030, M1031, M1042	M1042
DoS	Mitigación 1	-
<i>Eavesdropping</i>	Mitigación 3	-
<i>Masquerading of Identity</i>	Mitigación 5	-

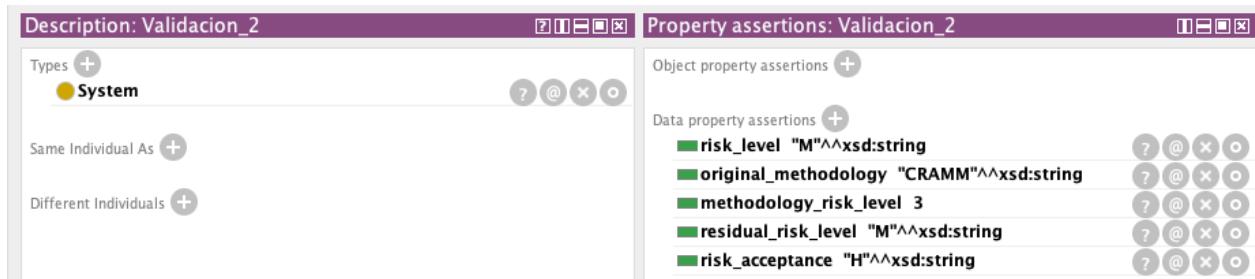


Figura 10.8: Nivel de riesgo residual global del sistema modificado. Validación - Caso de uso 2

Tabla 10.28: Niveles de riesgo residual modificado. Validación - Caso de uso 2

Riesgo	Riesgo Potencial	Riesgo Residual
<i>Abuse of Access Privileges</i>	M	-
<i>Deliberated Alteration of Information</i>	H	M
<i>Disclosure of Information</i>	H	L
DoS	M	-
<i>Eavesdropping</i>	M	-
<i>Masquerading of Identity</i>	M	-
Riesgo Global	M	M

Las contramedidas recomendadas se mantienen igual, pero únicamente se aplica la selección de la respuesta óptima para aquellas amenazas cuyo nivel de riesgo sea inferior al nivel de aceptación (*Deliberated Alteration of Information* y *Disclosure of Information* en este caso). Por este motivo, esta configuración no permite reducir el riesgo global.

10.3.5 Visualización de la información

La información almacenada en la ontología se puede visualizar a través de aplicaciones como *Protégé*, que muestran la información almacenada y razonada, o a través de una consola de mando y control, que permite al analista de seguridad tener toda la información disponible y fácilmente entendible, como el prototipo que se muestra en la Figura 10.9. Se puede observar el nivel de riesgo de los activos, y las relaciones entre ellos, la evolución del riesgo global del sistema de una fecha a otra y el nivel de riesgo asociado a cada amenaza, con las contramedidas recomendadas según el algoritmo, y la elección de la óptima, aunque el administrador puede elegir cuáles despliega o desactiva según la situación en la que se encuentre el sistema.

Esta perspectiva puede modificarse para añadir otra información interesante, como un mapa que asocie amenazas y activos, o vulnerabilidades, escenarios de riesgo, información de ataques identificados y su caracterización, o la asociación con APTs.

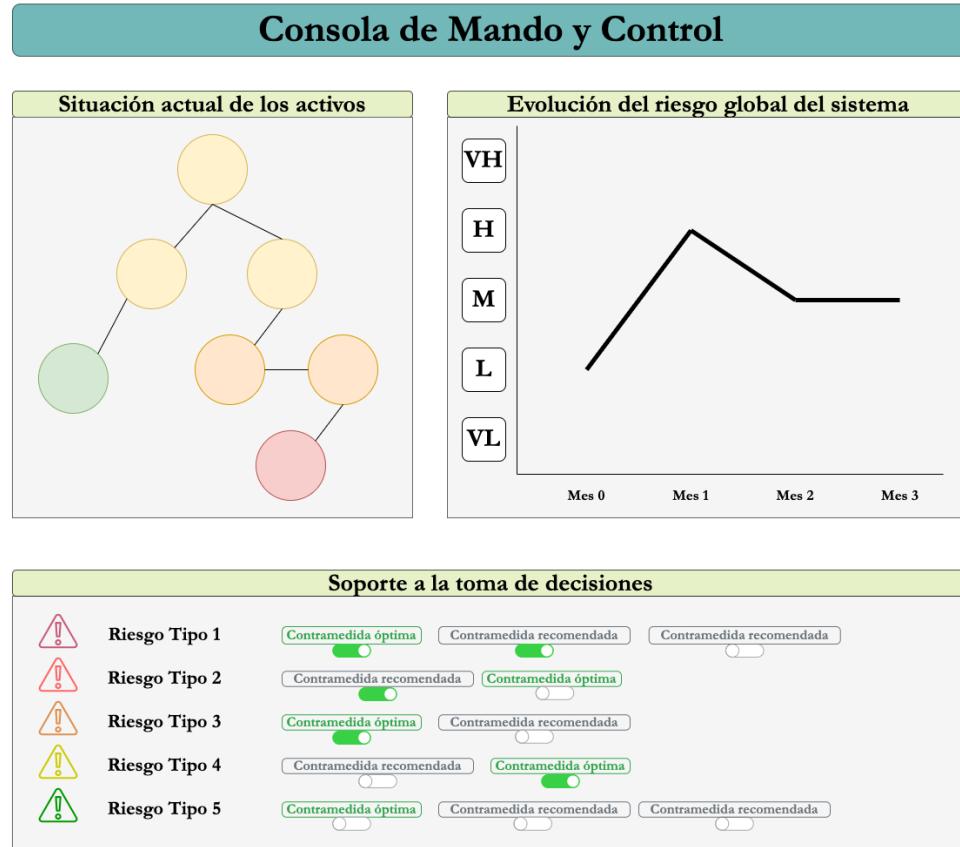


Figura 10.9: Prototipo de consola de mando y control

10.4 Conclusiones

Tras el desarrollo de los casos de uso presentados en este capítulo se ha analizado el resultado de la caracterización de los ciberataques desde distintas perspectivas, como la identificación de anomalías temporales o la identificación de técnicas y patrones de ataque. Además, se ha llevado a cabo la verificación del funcionamiento del entorno de conciencia cibersituacional completo.

La identificación de anomalías en fuentes heterogéneas permite analizar otros orígenes de posibles ciberataques más allá del tráfico de red, que es la más común. Además, por la manera en la que los modelos están entrenados, el resultado permite distinguir si es una anomalía por la información temporal del registro o si es por la primera detección del mismo en el entorno. No obstante, el mayor potencial de esta propuesta radica en la caracterización de técnicas en los registros, ya que permite extraer una gran cantidad de información relacionada gracias a la matriz de MITRE ATT&CK, como los CAPECS asociados o recomendaciones de mitigación.

Este enfoque, con una definición exhaustiva de los escenarios de riesgo y evaluación de las mitigaciones presentes en los activos, permite hacer frente a amenazas desconocidas, fragmentar los escenarios en función de los patrones de ataque identificados, y podría incluso asociar los registros de tráfico de red con distintos APTs que el atacante puede estar llevando

a cabo.

El sistema es altamente configurable y adaptable según la situación enfrentada a través de los catálogos de entrada, además de ser compatible con distintos marcos de gestión de riesgos gracias a su interoperabilidad.

En el primer caso de uso se ha verificado que la identificación de técnicas permite realizar un tratamiento de los riesgos más preciso, dividiendo las amenazas generadas por incidentes de red en distintos casos gracias a la identificación de TTPs y su relación con CAPECs y mitigaciones. Por otro lado, en el segundo caso de uso se ha explorado la posibilidad de extraer distintos análisis a raíz del resultado obtenido del sistema de detección de intrusiones no supervisado, ya que un porcentaje significativo de las anomalías puede deberse a nuevos dispositivos, y por tanto son falsos positivos que elevan el nivel de riesgo de forma irreal. Al considerar esta posibilidad, en lugar de tratar los incidentes de cada sensor en conjunto, se ajusta el nivel de riesgo calculado por el sistema al impacto de los ciberataques registrados.

Tras el análisis llevado a cabo, se han validado las Hipótesis 2 y 3 de esta Tesis Doctoral, comprobando el valor que aporta la caracterización de ciberataques al proceso de gestión de riesgos y al funcionamiento del entorno completo de conciencia cibersituacional.

Capítulo 11

Conclusiones y líneas futuras

En este capítulo se presentan las conclusiones principales de la Tesis Doctoral, que se han ido introduciendo anteriormente al definir las distintas contribuciones realizadas. Éstas se recogen en la Sección 11.1, con el objetivo de resumir la investigación realizada. Además, en la Sección 11.2 se destacan las contribuciones principales de la Tesis Doctoral al estado del arte a día de hoy. Finalmente, en la Sección 11.3 se plantean posibles direcciones para seguir investigando en este campo.

11.1 Conclusiones de la investigación realizada

El objetivo principal de la propuesta presentada a lo largo de esta Tesis Doctoral es la caracterización de ciberataques para entornos de conciencia cibersituacional. Esto implica detectar incidentes y obtener información que permita responder ante ellos de forma adecuada y proporcionar datos actualizados a los analistas de ciberseguridad sobre la situación de riesgo del sistema.

La contextualización de la problemática que trata de abordar esta investigación comienza en el Capítulo 1, exponiendo la necesidad creciente de conocer información sobre los ataques que presenta el ámbito de la ciberseguridad actualmente. Con el desarrollo de las nuevas tecnologías, los ataques se están volviendo más complejos y difíciles de detectar, por lo que las herramientas defensivas deben evolucionar y adaptarse al incidente, dejando atrás las propuestas genéricas que ya no pueden hacerles frente.

La metodología de la propuesta definida en el Capítulo 2 se basa en tres hipótesis que surgen de la motivación y el contexto presentados:

- Hipótesis 1: Los algoritmos de aprendizaje automático no supervisados obtienen mejores resultados en la detección de anomalías de comportamiento en entornos personalizados, con datos heterogéneos, que no se adaptan a ningún conjunto de datos conocido.
- Hipótesis 2: La caracterización de los ciberataques permite obtener resultados más precisos a la hora de calcular el riesgo al que se expone el sistema, adaptándose mejor a la consecuencia que implica cada incidente.

- Hipótesis 3: La gestión global en un entorno de conciencia cibersituacional compuesta por sistemas de detección de intrusiones y un proceso de gestión de riesgos permite establecer un soporte a la toma de decisiones ajustable al entorno y la información extraída de la caracterización de los ataques.

Para poder validar o refutarlas, se definen un conjunto de objetivos que se dividen en tareas a realizar y que marcan el desarrollo de la investigación hasta alcanzarlos.

En primer lugar, es fundamental el estudio del estado del arte. Dado que la propuesta aborda herramientas de distintos ámbitos aplicadas a la ciberseguridad, es necesario conocer la situación en la que se encuentran, sus principales retos y líneas de investigación. Así, en el Capítulo 3 se ahonda en el concepto de *Cyber Threat Hunting* como fundamento de la caracterización de ciberataques. En este proceso de investigación destaca la dificultad de los IDS actuales para identificar ataques de día cero, específicamente, ya que no coinciden con comportamientos vistos anteriormente y por tanto, los modelos de IA no son capaces de asociarlos con ataques según su entrenamiento. Las tendencias actuales se centran en el desarrollo de modelos de comportamiento, que aprenden el funcionamiento estándar del sistema o red, y consideran anómalo cualquier muestra que se diferencie de éste. Estos algoritmos tienen la ventaja de no requerir *datasets*, que pueden encontrarse desactualizados, pero generan una tasa de falsos positivos muy alta, especialmente al principio, hasta que adapte su base de conocimiento al entorno en el que se despliegan. Igualmente, destaca el trabajo de recopilación de información sobre ciberataques llevado a cabo por MITRE, lo que hace esta caracterización más efectiva, pero requiere un equilibrio entre proporcionar información a la comunidad para proteger los sistemas y que los atacantes conozcan las medidas defensivas y preparen los ataques para evitarlas.

La tecnología más utilizada para este tipo de tareas en la actualidad es la inteligencia artificial. Como se describe en el Capítulo 4, ha supuesto una revolución tanto en la forma de realizar ataques como de defender los sistemas. Tanto en la rama de aprendizaje automático como en la de ontologías, la IA presenta ventajas que la convierten en idónea para tratar la problemática identificada. Sin embargo, no puede ser un sistema autónomo, requiere la gestión humana para poder adaptarse al entorno y definir comportamientos eficaces en la protección de activos y reducción de riesgos. La combinación del conocimiento humano en el diseño y la eficiencia de la IA para realizar tareas recurrentes a gran velocidad, con volúmenes de datos mucho más altos de los que una persona podría procesar, permiten desarrollar sistemas dinámicos, capaces de trabajar en tiempo real de forma fiable pero a la vez flexible.

La gestión de riesgos es imprescindible en un sistema orientado a la defensa, especialmente en entornos de conciencia cibersituacional. En el Capítulo 5 se presentan las metodologías más utilizadas a nivel europeo, analizando cómo abordan los riesgos y amenazas. Las elegidas, por su enfoque y la disponibilidad de información sobre los distintos procedimientos que las componen son EBIOS, MAGERIT, MONARC, ITSRM y CRAMM, que han sido definidas por distintos organismos con un objetivo y una propuesta para evaluar el nivel de riesgo, pero que no son interoperables por sí mismas. Para extraer puntos en común sobre distintos aspectos funcionales, se establece una comparación entre ellas, para crear un marco interoperable en la propuesta de esta Tesis Doctoral. La falta de esta capacidad impide llevar a cabo tareas vitales para la seguridad, como el intercambio de información sobre amenazas entre organizaciones.

Conocer cómo otros ha respondido a las mismas amenazas a las que ahora se hace frente puede favorecer la reducción del riesgo que generan, pero no es posible sin una escala común en la que compararlos.

Tras este estudio se localiza la necesidad de contribuir a los entornos de conciencia cibersituacional mediante la caracterización de ataques y una propuesta de metodología dinámica de gestión de riesgos interoperable con los marcos analizados. La propuesta y el modelo de arquitectura que se extraen de la investigación de la Tesis Doctoral se presenta en el Capítulo 6, y cada módulo se detalla en los siguientes capítulos.

- En primer lugar, un sistema de detección de anomalías a partir de datos heterogéneos se plantea en el Capítulo 7, donde la problemática de los conjuntos de datos para entrenar IDS se evita mediante un sistema basado en modelos no supervisados capaces de detectar anomalías en los sensores físicos y lógicos, entrenado en base al comportamiento de la red. Aquí se valida la primera hipótesis de la Tesis Doctoral, consiguiendo que el sistema detecte anomalías según las características de la muestra y del marco temporal en el que se reciban.
- En el Capítulo 8 se desarrolla el núcleo de la caracterización de los ciberataques. Los incidentes llevados a cabo a través del tráfico de red son una de las amenazas más frecuentes de los sistemas actuales, y las técnicas utilizadas por los atacantes cada vez son más variadas, provocando que no sirvan las mismas medidas para responder a todos. La identificación de estas TTPs y las estrategias de los ciberdelincuentes utilizando aprendizaje automático proporciona información muy valiosa para responder a los incidentes, como las mitigaciones recomendadas frente a estas situaciones en lugar de responder con contramedidas definidas en catálogos genéricos o los patrones de ataque conocidos que pueden estar ocurriendo y en qué fase se encuentran a raíz de estas técnicas, o incluso la identificación de APTs.
- Por último, se define una ontología para la gestión dinámica del riesgo interoperable con las otras metodologías y un gestor de información que permite el análisis y evaluación del riesgo en tiempo real en base a la caracterización de los ciberataques y finalizado con un soporte a la toma de decisiones para recomendar la mitigación óptima frente a estos incidentes.

Aunque todos los módulos se validan individualmente, en el Capítulo 10 se plantean casos de uso para poder evaluar el entorno global de conciencia cibersituacional desde dos perspectivas, el efecto de la caracterización frente a los resultados en su ausencia y la implementación del diseño, validando finalmente las Hipótesis 2 y 3 de la Tesis Doctoral.

11.2 Contribución al conocimiento

A raíz de los resultados de las tareas presentadas en el Capítulo 2 en la consecución de los objetivos definidos para la Tesis Doctoral, se obtienen un conjunto de contribuciones de la investigación abordada a la caracterización de ciberamenazas y en global al ámbito de la ciberseguridad. En concreto, se compone de un conjunto de propuestas para mejorar la seguridad de los activos de una organización a través de la identificación de amenazas

procedentes de medios heterogéneos (Wi-Fi, *Bluetooth*, radio frecuencia, redes móviles o tráfico de red y comportamiento del usuario) y su caracterización para obtener información relevante en la gestión de estas ciberamenazas y los riesgos que conllevan.

Las tres contribuciones principales de la Tesis Doctoral se describen a continuación, destacando la problemática identificada y la solución planteada:

- **Contribución 1: Propuesta para la detección automática de anomalías en registros heterogéneos.**

La mayor problemática de la seguridad a día de hoy son los ataques de día cero en entornos con multitud de sensores que generan información heterogénea. Son uno de los tipos de amenazas más identificadas en los últimos años, y preparar sistemas capaces de identificarlos entre el tráfico normal es una tarea complicada. Además, las características del entorno dificultan esta detección, ya que no existen conjuntos de datos públicos con tráfico de todos estos sensores de comunicaciones que permitan entrenar los modelos.

Como respuesta, la primera contribución de esta Tesis Doctoral es el desarrollo de un conjunto de modelos de aprendizaje automático no supervisado entrenados sobre el comportamiento normal de un escenario heterogéneo que componen un IDS capaz de identificar anomalías en las muestras capturadas por los sensores, y caracterizar de forma sencilla estos ataques en función de si la anomalía se debe al sello temporal de la muestra o a sus características. Esta diferenciación permite plantear distintos escenarios de riesgo, evaluando los incidentes con más probabilidad de ser falsos positivos y reduciendo su efecto sobre el cálculo del riesgo para adecuar más este nivel a la realidad y no elevarlo debido a falsas alarmas.

- **Contribución 2: Propuesta para la caracterización de técnicas MITRE ATT&CK en ciberataques de tráfico.**

El medio más frecuente por el que se identifican ciberataques en la actualidad es el tráfico de red, ya que proporciona acceso a información crítica y servicios que son fundamentales para la operación de cualquier organización. Un análisis sobre las técnicas más utilizadas en los ataques conduce a la conclusión de que identificarlas puede permitir no sólo clasificar ese flujo como anómalo, sino proporcionar información imprescindible sobre el ataque que se podría estar llevando a cabo. Existen enfoques en este aspecto centrados en el procesamiento del lenguaje natural, pero el desarrollo sobre la caracterización de registros de tráfico según sus atributos era limitado hasta la fecha.

Por ese motivo, otra contribución de la Tesis Doctoral al ámbito de CTH es un sistema basado en aprendizaje automático, entrenado con registros etiquetados con las técnicas identificadas, capaz de analizar el tráfico recibido y extraer el patrón de ataque relacionado con el TTP o las mitigaciones recomendadas por MITRE. En conjunto estos datos permiten saber en qué estado se encuentra el ataque (qué paso del patrón de ataque representa la técnica) o cómo hacerle frente. Entre las TTPs que el modelo es capaz de caracterizar se encuentran tres de las más identificadas en los informes de análisis de los ciberataques más frecuentes en la actualidad: T1190 - *Exploit Public-Facing Application*, T1587 - *Develop Capabilities* y T1595 - *Active Scanning*.

- **Contribución 3: Propuesta de una metodología basada en una ontología interoperable para marcos de gestión dinámica de riesgos.**

Otra contribución parcial de esta Tesis Doctoral a la caracterización de ciberataques en entornos de conciencia cibersituacional se basa en el desarrollo de un modelo de conocimiento adaptado al dominio de la ciberseguridad y con capacidad de interoperear información de distintas metodologías de gestión de riesgo. Un asunto pendiente actualmente en ciberseguridad es la capacidad de intercambio de información, ya que los atacantes suelen buscar distintos objetivos, y la necesidad de tener una base común, tanto en lo referido a terminología como en procedimientos, crece con el desarrollo tecnológico.

La metodología propuesta traslada a una escala común los procesos de los marcos de gestión de riesgos más utilizados en Europa, permitiendo comparar los resultados de sus planes de respuesta y el intercambio de información. Además, la ontología como base de conocimiento y el conjunto de reglas de comportamiento se complementan para establecer un soporte a la toma de decisiones y recomendar respuestas, eligiendo la contramedida óptima en función de la caracterización de los incidentes registrados.

Las conclusiones y resultados obtenidos de esta investigación han sido compartidos con la comunidad científica, dando como resultado las siguientes publicaciones con índice de impacto:

1. Xavier Larriva-Novo, Carmen Sánchez-Zas, Víctor A. Villagrá, Mario Vega-Barbas, Diego Rivera. “An Approach for the Application of a Dynamic Multi-Class Classifier for Network Intrusion Detection Systems”, en *Electronics*, 9 (2020). [68]
2. Carmen Sánchez-Zas, Xavier Larriva-Novo, Víctor A. Villagrá, Mario Sanz Rodrigo, José Ignacio Moreno. “Design and Evaluation of Unsupervised Machine Learning Models for Anomaly Detection in Streaming Cybersecurity Logs”, en *Mathematics* 10 (2022). [111]
3. Carmen Sánchez-Zas, Víctor A. Villagrá, Mario Vega-Barbas, Xavier Larriva-Novo, José Ignacio Moreno, Julio Berrocal. “Ontology-based approach to real-time risk management and cyber-situational awareness”, en *Future Generation Computer Systems* 141 (2023). [114]
4. Xavier Larriva-Novo, Carmen Sánchez-Zas, Víctor A. Villagrá, Andrés Marín-López, Julio Berrocal. “Leveraging Explainable Artificial Intelligence in Real-Time Cyberattack Identification: Intrusion Detection System Approach”, en *Applied Sciences* 13 (2023). [70]
5. Carmen Sánchez-Zas, Xavier Larriva-Novo, Víctor A. Villagrá, Diego Rivera, Andrés Marín-Lopez. “A methodology for ontology-based interoperability of dynamic risk assessment frameworks in IoT environments” en *Internet of Things*. Enviado, en proceso de revisión.

Esta difusión se complementa con la participación en distintas ediciones de las Jornadas Nacionales de Investigación en Ciberseguridad, generando las siguientes publicaciones sin índice de impacto:

- Carmen Sánchez-Zas, Víctor A. Villagrá, Mario Vega-Barbas, Xavier Larriva-Novo, José Ignacio Moreno, Julio Berrocal. “Sistema de conciencia cibersituacional y gestión dinámica de riesgos basado en ontologías”, en *Actas de las VII Jornadas Nacionales de Investigación en Ciberseguridad*, Bilbao, 2022 [112].
- Carmen Sánchez-Zas, Xavier Larriva-Novo, Víctor A. Villagrá, Mario Sanz Rodrigo, Sonia Solera-Cotanilla. “Desarrollo de una ontología para modelar una metodología interoperable de gestión dinámica de riesgos”, en *Actas de las VIII Jornadas Nacionales de Investigación en Ciberseguridad*, Vigo, 2023 [113].
- Xavier Larriva-Novo, Alba Vara Plaza, Óscar Jover, Carmen Sánchez-Zas, Víctor A. Villagrá. “Simulador de APTs realistas avanzados basado en el marco de MITRE ATT&CK”, en *Actas de las VIII Jornadas Nacionales de Investigación en Ciberseguridad*, Vigo, 2023 [71].
- Carmen Sánchez-Zas, Xavier Larriva-Novo, Víctor A. Villagrá, Diego Rivera, Sonia Solera-Cotanilla. “Sistema de caracterización de técnicas MITRE ATT&CK en incidentes de ciberseguridad”, en *IX Jornadas Nacionales de Investigación en Ciberseguridad*, Sevilla, 2024.

En la Figura 11.1 se representa de manera gráfica la relación de las hipótesis (**Hx**), contribuciones y publicaciones (**Px**, si están indexadas, y **Jx**, si pertenecen a jornadas de investigación) mencionadas sobre la metodología propuesta en la Tesis Doctoral.

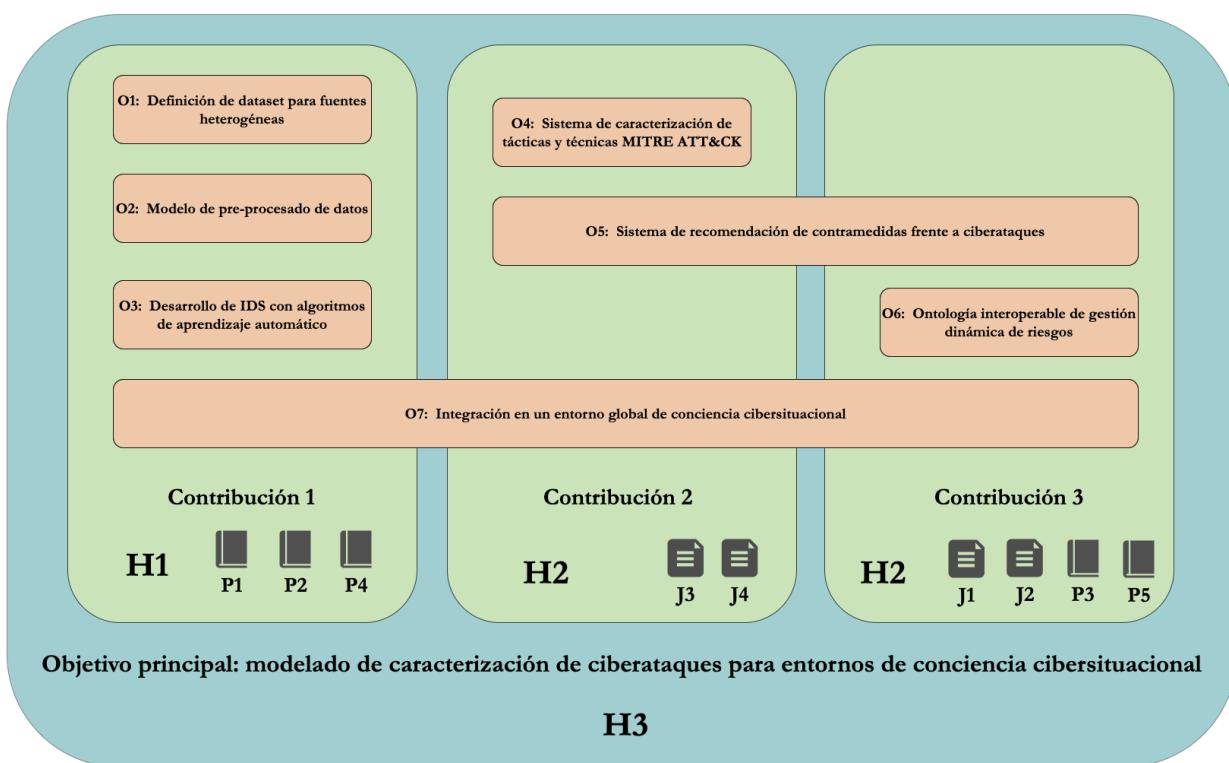


Figura 11.1: Contribuciones y publicaciones en el marco de la propuesta

En ésta se muestran las siguientes relaciones:

- **Contribución 1:**

- Hipótesis: H1, H3.
- Objetivos: O1, O2, O3, O7, Objetivo principal.
- Publicaciones:
 - * P1: “An Approach for the Application of a Dynamic Multi-Class Classifier for Network Intrusion Detection Systems” [68].
 - * P2: “Design and Evaluation of Unsupervised Machine Learning Models for Anomaly Detection in Streaming Cybersecurity Logs” [111].
 - * P4: “Leveraging Explainable Artificial Intelligence in Real-Time Cyberattack Identification: Intrusion Detection System Approach” [70].

- **Contribución 2:**

- Hipótesis: H2, H3.
- Objetivos: O4, O5, O7, Objetivo principal.
- Publicaciones:
 - * J3: “Simulador de APTs realistas avanzados basado en el marco de MITRE ATT&CK” [71].
 - * J4: “Sistema de caracterización de técnicas MITRE ATT&CK en incidentes de ciberseguridad”.

- **Contribución 3:**

- Hipótesis: H2, H3.
- Objetivos: O5, O6, O7, Objetivo principal.
- Publicaciones:
 - * J1: “Sistema de conciencia cibersituacional y gestión dinámica de riesgos basado en ontologías” [112].
 - * J2: “Desarrollo de una ontología para modelar una metodología interoperable de gestión dinámica de riesgos” [113].
 - * P3: “Ontology-based approach to real-time risk management and cyber-situational awareness” [114].
 - * P5: “A methodology for ontology-based interoperability of dynamic risk assessment frameworks in IoT environments”. Enviado, en proceso de revisión.

11.3 Líneas futuras

El desarrollo y la evolución de la tecnología implica que las investigaciones nunca se concluyan por completo. Durante el proceso, la propuesta inicial se expande, surgen ramificaciones o se puede profundizar en algún concepto abordado. Así, de cualquier investigación nacen nuevas líneas, motivando los avances en el estado del arte.

En particular, de la propuesta presentada en esta Tesis Doctoral, existen varios aspectos que podrían extenderse:

- Ampliar el número de técnicas que el sistema de caracterización en registros de tráfico de red puede detectar. Actualmente el sistema es capaz de identificar las técnicas que contiene el *dataset* de entrenamiento. El módulo puede validarse con otros conjuntos de datos, pero principalmente se propone identificar o construir un conjunto ampliado y re-entrenar el modelo de aprendizaje supervisado. Ampliar el abanico de TTPs permite obtener una caracterización más completa, ya que el sistema sería capaz de identificar ataques en distintas fases. Ya se ha probado la eficacia de la caracterización de las técnicas y la cantidad de información que se puede extraer a partir de ello, por lo que ampliar el número repercutirá positivamente en el resultado final.
- Otra opción para elevar la cantidad de técnicas identificadas es la definición de reglas que permitan localizarlas en el tráfico de red. Complementar el sistema basado en modelos con reglas de detección permite establecer un módulo más fiable.
- Desarrollar la identificación posterior de APTs. A raíz de las técnicas identificadas también se puede obtener información sobre campañas en las que se utilizan. Especialmente si en algún APT coinciden varias técnicas identificadas consecutivamente puede ser una señal muy clara de ataque, y conocer el procedimiento del atacante permite al defensor prepararse para contrarrestarlo.
- En la propuesta, la caracterización más profunda se ha limitado al tráfico de red, pero en entornos heterogéneos como el utilizado, las amenazas pueden producirse de múltiples formas. Identificar tipos de ataque en el tráfico de los sensores Wi-Fi, *Bluetooth*, radio frecuencia o redes móviles favorece la definición de escenarios de riesgo más precisos. Principalmente, es necesario el depurado de la detección de anomalías temporales, planteando la posibilidad de dividir el modelo en dos, o implantar un mecanismo de listas blancas y negras para definir horas anómalas y otros tipos de limitaciones.
- Detección de amenazas procedentes de *insiders*. El análisis del comportamiento de los usuarios introducido en esta Tesis Doctoral abre la puerta al estudio de amenazas internas, que no se consideran en la propuesta. Este sensor debe mejorarse, pero la versión inicial ya obtiene resultados prometedores. La detección de *insiders* implica un enfoque de estudio distinto al planteado aquí, ya que este tipo de amenazas se confunden con tráfico normal.
- La interoperabilidad todavía es un aspecto a trabajar en el mundo de la ciberseguridad actualmente. La ontología y el sistema de gestión propuesto pueden mejorarse, incluyendo nuevos marcos de gestión de riesgo y optimizando los procesos ya incluidos.

- Una de las debilidades principales del sistema actual es la identificación manual de vulnerabilidades en los activos del sistema y su efecto sobre las amenazas y activos. Automatizar este proceso evita errores o que algunas se pasen por alto. Es una de las líneas futuras prioritarias en el desarrollo de la ontología para la gestión dinámica de riesgos, junto a la definición de condiciones para la ejecución de las contramedidas. Además, para mejorar el catálogo de activos, se pueden introducir pesos para distinguir la importancia para la organización de cada dimensión de la seguridad.
- El desarrollo de la consola de mando y control, que permite visualizar la información referente al riesgo en el entorno de conciencia cibersituacional, facilita las tareas de supervisión y decisión al analista de ciberseguridad encargado del sistema.

Estas líneas futuras se plantean sobre la investigación realizada y el estado del arte existente. Sin embargo, la tecnología es la herramienta que permite llevar a cabo cualquier avance. Su evolución ha hecho posible desarrollos inimaginables a lo largo de la historia, y puede expandir las líneas futuras de ésta y cualquier investigación hasta que sean innumerables. Sin embargo, la tecnología es una combinación de creatividad e ingeniería por lo que, sin el ser humano, corre el riesgo de estancarse.

Referencias

- [1] “Apache Kafka”. Disponible en: <https://kafka.apache.org/documentationstreams/>. Accedido: 8-mar-2024.
- [2] “Apache Spark - Extracting, transforming and selecting features”. Disponible en: <https://spark.apache.org/docs/latest/ml-features>. Accedido: 8-mar-2024.
- [3] “Interoperable EU Risk Management Toolbox - ENISA”. Disponible en: <https://www.enisa.europa.eu/publications/interoperable-eu-risk-management-toolbox>. Accedido: 13-abr-2024.
- [4] “Trumania”. Disponible en: <https://github.com/RealImpactAnalytics/trumania>. Accedido: 8-mar-2024.
- [5] Temitope Elizabeth Abioye et al. “Toward ontology-based risk management framework for software projects: An empirical study”. En: *Journal of Software: Evolution and Process* 32.12 (2020), e2269. DOI: [10.1002/sm.2269](https://doi.org/10.1002/sm.2269).
- [6] Md Sahrom Abu et al. “Formulation of Association Rule Mining (ARM) for an Effective Cyber Attack Attribution in Cyber Threat Intelligence (CTI)”. En: *International Journal of Advanced Computer Science and Applications* 12.4 (2021). DOI: [10.14569/IJACSA.2021.0120418](https://doi.org/10.14569/IJACSA.2021.0120418).
- [7] Constantin Adam et al. *Attack Techniques and Threat Identification for Vulnerabilities*. 2022. arXiv: [2206.11171 \[cs.CR\]](https://arxiv.org/abs/2206.11171).
- [8] Mohamed Ahmed et al. “MITRE ATT&CK-driven Cyber Risk Assessment”. En: *Proceedings of the 17th International Conference on Availability, Reliability and Security*. Vienna Austria: ACM, 2022, págs. 1-10. DOI: [10.1145/3538969.3544420](https://doi.org/10.1145/3538969.3544420).
- [9] Fahad H. Alshammari. “Design of capability maturity model integration with cybersecurity risk severity complex prediction using bayesian-based machine learning models”. En: *Service Oriented Computing and Applications* 17.1 (2023), págs. 59-72. DOI: [10.1007/s11761-022-00354-4](https://doi.org/10.1007/s11761-022-00354-4).
- [10] Manuel Alvarez-Campana et al. “Smart CEI Moncloa: An IoT-based Platform for People Flow and Environmental Monitoring on a Smart University Campus”. En: *Sensors* 17.12 (2017). DOI: [10.3390/s17122856](https://doi.org/10.3390/s17122856).
- [11] Andrei Brazhuk. “Threat modeling of cloud systems with ontological security pattern catalog”. En: *International Journal of Open Information Technologies* 9.5 (2021), págs. 36-41.
- [12] Silvia Ansaldi et al. “An Ontology for the Identification of the most Appropriate Risk Management Methodology”. En: *On the Move to Meaningful Internet Systems: OTM 2012 Workshops*. Vol. 7567. Series Title: Lecture Notes in Computer Science. Berlin,

- Heidelberg: Springer Berlin Heidelberg, 2012, págs. 444-453. DOI: [10.1007/978-3-642-33618-8\60](https://doi.org/10.1007/978-3-642-33618-8_60).
- [13] Oluwasefunmi T. Arogundade, Adebayo Abayomi-Alli y Sanjay Misra. “An Ontology-Based Security Risk Management Model for Information Systems”. En: *Arabian Journal for Science and Engineering* 45.8 (2020), págs. 6183-6198. DOI: [10.1007/s13369-020-04524-4](https://doi.org/10.1007/s13369-020-04524-4).
- [14] Julien Audibert et al. “USAD: UnSupervised Anomaly Detection on Multivariate Time Series”. En: *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. KDD '20. Virtual Event, CA, USA: Association for Computing Machinery, 2020, págs. 3395-3404. DOI: [10.1145/3394486.3403392](https://doi.org/10.1145/3394486.3403392).
- [15] Abdullah Aydeger, Nico Saputro y Kemal Akkaya. “Cloud-based Deception against Network Reconnaissance Attacks using SDN and NFV”. En: *2020 IEEE 45th Conference on Local Computer Networks (LCN)*. Sydney, NSW, Australia: IEEE, 2020, págs. 279-285. DOI: [10.1109/LCN48667.2020.9314797](https://doi.org/10.1109/LCN48667.2020.9314797).
- [16] Sikha Bagui et al. “Detecting Reconnaissance and Discovery Tactics from the MITRE ATT&CK Framework in Zeek Conn Logs Using Spark’s Machine Learning in the Big Data Framework”. En: *Sensors* 22.20 (2022). DOI: [10.3390/s22207999](https://doi.org/10.3390/s22207999).
- [17] Sikha Bagui et al. “Resampling Imbalanced Network Intrusion Datasets to Identify Rare Attacks”. En: *Future Internet* 15.4 (2023). DOI: [10.3390/fi15040130](https://doi.org/10.3390/fi15040130).
- [18] Sikha S. Bagui et al. “Introducing the UWF-ZeekDataFall22 Dataset to Classify Attack Tactics from Zeek Conn Logs Using Spark’s Machine Learning in a Big Data Framework”. En: *Electronics* 12.24 (2023). DOI: [10.3390/electronics12245039](https://doi.org/10.3390/electronics12245039).
- [19] Sikha S. Bagui et al. “Introducing UWF-ZeekData22: A Comprehensive Network Traffic Dataset Based on the MITRE ATT&CK Framework”. En: *Data* 8.1 (2023). DOI: [10.3390/data8010018](https://doi.org/10.3390/data8010018).
- [20] Sikha S. Bagui et al. “Using a Graph Engine to Visualize the Reconnaissance Tactic of the MITRE ATT&CK Framework from UWF-ZeekData22”. En: *Future Internet* 15.7 (2023). DOI: [10.3390/fi15070236](https://doi.org/10.3390/fi15070236).
- [21] Ángel Casanova Bienzobas y Alfonso Sánchez-Macián. *Threat Trekker: An Approach to Cyber Threat Hunting*. 2023. arXiv: [2310.04197 \[cs.CR\]](https://arxiv.org/abs/2310.04197).
- [22] Ron Bitton et al. “Evaluating the Cybersecurity Risk of Real-world, Machine Learning Production Systems”. En: *ACM Computing Surveys* 55.9 (2023), págs. 1-36. DOI: [10.1145/3559104](https://doi.org/10.1145/3559104).
- [23] Branko Bokan y Joost Santos. “Managing Cybersecurity Risk Using Threat Based Methodology for Evaluation of Cybersecurity Architectures”. En: *2021 Systems and Information Engineering Design Symposium (SIEDS)*. Charlottesville, VA, USA: IEEE, 2021, págs. 1-6. DOI: [10.1109/SIEDS52267.2021.9483736](https://doi.org/10.1109/SIEDS52267.2021.9483736).
- [24] Dibya Jyoti Bora y Anil Kumar Gupta. “Effect of Different Distance Measures on the Performance of K-Means Algorithm: An Experimental Study in Matlab”. En: *International Journal of Computer Science and Information Technologies, (IJCSIT)* 5 (2014), págs. 2501-2506. DOI: <https://doi.org/10.48550/arXiv.1405.7471>.
- [25] Andrew P. Bradley. “The use of the area under the ROC curve in the evaluation of machine learning algorithms”. En: *Pattern Recognition* 30.7 (1997), págs. 1145-1159. DOI: [https://doi.org/10.1016/S0031-3203\(96\)00142-2](https://doi.org/10.1016/S0031-3203(96)00142-2).

- [26] Mariana G. Cains et al. "Defining Cyber Security and Cyber Security Risk within a Multidisciplinary Context using Expert Elicitation". En: *Risk Analysis* 42.8 (2022), págs. 1643-1669. DOI: [10.1111/risa.13687](https://doi.org/10.1111/risa.13687).
- [27] Yanshuai Cao y Luyu Wang. "Automatic Selection of t-SNE Perplexity". En: *CoRR* abs/1708.03229 (2017). arXiv: [1708.03229](https://arxiv.org/abs/1708.03229).
- [28] CAPEC - About CAPEC. Disponible en: <https://capec.mitre.org/about/index.html>. Accedido: 31-mar-2024.
- [29] CAPEC - New to CAPEC? Disponible en: https://capec.mitre.org/about/new_to_capec.html. Accedido: 31-mar-2024.
- [30] CAPEC VIEW: ATT&CK Related Patterns. Disponible en: <https://capec.mitre.org/data/definitions/658.html>. Accedido: 31-mar-2024.
- [31] CCN-CERT BP/30: Aproximación a la Inteligencia Artificial y la ciberseguridad. Informe de Buenas Prácticas. Disponible en: <https://www.ccn-cert.cni.es/es/informes/informes-de-buenas-practicas-bp/7190-ccn-cert-bp-30-aproximacion-a-la-inteligencia-artificial-y-la-ciberseguridad/file.html>. Accedido: 4-abr-2024.
- [32] CCN-CERT IA-35/23: Ciberamenazas y Tendencias. Disponible en: <https://www.ccn-cert.cni.es/es/informes/informes-ccn-cert-publicos/7188-ccn-cert-ia-35-23-ciberamenazas-y-tendencias-edicion-2023/file.html>. Accedido: 5-abr-2024.
- [33] Aleš Černivec et al. "Employing Graphical Risk Models to Facilitate Cyber-Risk Monitoring - the WISER Approach". En: *Graphical Models for Security*. Vol. 10744. Cham, 2018, págs. 127-146. DOI: [10.1007/978-3-319-74860-3_10](https://doi.org/10.1007/978-3-319-74860-3_10).
- [34] Harsh Chaudhary et al. "A review of various challenges in cybersecurity using Artificial Intelligence". En: *2020 3rd International Conference on Intelligent Sustainable Systems (ICISS)*. 2020, págs. 829-836. DOI: [10.1109/ICISS49785.2020.9316003](https://doi.org/10.1109/ICISS49785.2020.9316003).
- [35] Nitesh V. Chawla et al. "SMOTE: Synthetic Minority Over-sampling Technique". En: *Journal of Artificial Intelligence Research* 16 (2002). arXiv:1106.1813 [cs], págs. 321-357. DOI: [10.1613/jair.953](https://doi.org/10.1613/jair.953).
- [36] Chung-Kuan Chen et al. "Building Machine Learning-based Threat Hunting System from Scratch". En: *Digital Threats* 3.3 (2022). DOI: [10.1145/3491260](https://doi.org/10.1145/3491260).
- [37] Instituto Nacional de Ciberseguridad (INCIBE). "Gestión de Riesgos - Una guía de Aproximación para el empresario". Disponible en: https://www.incibe.es/sites/default/files/contenidos/guias/doc/guia_ciberseguridad_gestion_riesgos_metad.pdf. Accedido: 13-feb-2024.
- [38] Andy Coenen y Adam Pearce. "Understanding UMAP". Disponible en: <https://pair-code.github.io/understanding-umap/>. Accedido: 13-feb-2024.
- [39] Luxembourg House of Cybersecurity. "What is MONARC? - MONARC". Disponible en: <https://www.monarc.lu/>. Accedido: 5-feb-2024.
- [40] European Union Agency for Cybersecurity. "Compendium of risk management frameworks with potential interoperability: supplement to the interoperable EU risk management framework report." Publications Office, 2022.
- [41] European Union Agency for Cybersecurity. "Interoperable EU risk management framework: methodology for and assessment of interoperability among risk management frameworks and methodologies." Publications Office, 2022.

- [42] Claudia d'Amato et al. "Inductive reasoning and semantic web search". En: *Proceedings of the 2010 ACM Symposium on Applied Computing*. ACM, 2010, págs. 1446-1447. DOI: [10.1145/1774088.1774397](https://doi.org/10.1145/1774088.1774397).
- [43] Department of Computer Science, University of West Florida: "UWF-ZeekData22 Dataset". Disponible en: <https://datasets.uwf.edu/index.html>. Accedido: 26-feb-2024.
- [44] Gordana Dodig-Crnkovic. "Scientific methods in computer science". En: *Proceedings of the Conference for the Promotion of Research in IT at New Universities and at University Colleges in Sweden, Skövde, Suecia*. 2002, págs. 126-130.
- [45] Club Ebios. "EBIOS: the risk management toolbox". Disponible en: <https://club-ebios.org/site/wp-content/uploads/productions/EBIOS-GenericApproach-2018-09-05-Approved.pdf>. Accedido: 5-feb-2024.
- [46] Gal Engelberg et al. "An Ontology-Driven Approach for Process-Aware Risk Propagation". En: *Proceedings of the 38th ACM/SIGAPP Symposium on Applied Computing*. Tallinn Estonia: ACM, 2023, págs. 1742-1745. DOI: [10.1145/3555776.3577795](https://doi.org/10.1145/3555776.3577795).
- [47] ENISA. "Cramm". Disponible en: https://www.enisa.europa.eu/topics/risk-management/current-risk/risk-management-inventory/rm-ra-methods/m_cramm.html. Accedido: 5-feb-2024.
- [48] ENISA. "EBIOS". Disponible en: https://www.enisa.europa.eu/topics/risk-management/current-risk/risk-management-inventory/rm-ra-methods/m_ebios.html. Accedido: 5-feb-2024.
- [49] Yadigar Erdem y Caner Ozcan. "Fast Data Clustering and Outlier Detection using K-Means Clustering on Apache Spark". En: *International Journal of Advanced Computational Engineering and Networking* 5 (2017), págs. 86-90.
- [50] Fact++ reasoner. Disponible en: <http://owl.cs.manchester.ac.uk/tools/fact/>. Accedido: 30-mar-2024.
- [51] Daniel Jorge Ferreira y Henrique São Mamede. "Predicting Cybersecurity Risk - A Methodology for Assessments". En: *ARIS2 - Advanced Research on Information Systems Security* 2.2 (2022), págs. 50-63. DOI: [10.56394/arис2.v2i2.23](https://doi.org/10.56394/arис2.v2i2.23).
- [52] Mohtadi Ben Fraj. "InDepth: Parameter tuning for Decision Tree". Disponible en: <https://medium.com/@mohtedibf/indepth-parameter-tuning-for-decision-tree-6753118a03c3>. Accedido: 26-mar-2024.
- [53] Ulrik Franke y Joel Brynielsson. "Cyber situational awareness – A systematic review of the literature". En: *Computers & Security* 46 (2014), págs. 18-31. DOI: <https://doi.org/10.1016/j.cose.2014.06.008>.
- [54] Peng Gao et al. "ThreatKG: A Threat Knowledge Graph for Automated Open-Source Cyber Threat Intelligence Gathering and Management". En: *arXiv* (2022). DOI: [10.48550/ARXIV.2212.10388](https://doi.org/10.48550/ARXIV.2212.10388).
- [55] Birte Glimm et al. "HermitT: An OWL 2 Reasoner". En: *Journal of Automated Reasoning* 53.3 (2014), págs. 245-269. DOI: [10.1007/s10817-014-9305-1](https://doi.org/10.1007/s10817-014-9305-1).
- [56] G. Gonzalez-Granadillo et al. "Dynamic risk management response system to handle cyber threats". En: *Future Generation Computer Systems* 83 (2018), págs. 535-552. DOI: [10.1016/j.future.2017.05.043](https://doi.org/10.1016/j.future.2017.05.043).
- [57] Christos Grigoriadis et al. "A Cybersecurity Ontology to Support Risk Information Gathering in Cyber-Physical Systems". En: *Computer Security. ESORICS 2021 In-*

- ternational Workshops*. Vol. 13106. Series Title: Lecture Notes in Computer Science. Cham: Springer International Publishing, 2022, págs. 23-39. DOI: [10.1007/978-3-030-95484-0_2](https://doi.org/10.1007/978-3-030-95484-0_2).
- [58] Nicola Guarino, Daniel Oberle y Steffen Staab. "What Is an Ontology?" En: *Handbook on Ontologies*. Ed. por Steffen Staab y Rudi Studer. Springer Berlin Heidelberg, 2009, págs. 1-17. DOI: [10.1007/978-3-540-92673-3_0](https://doi.org/10.1007/978-3-540-92673-3_0).
- [59] Ministerio de Hacienda y Administraciones Pùblicas. "*MAGERIT – versión 3.0. Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información. Libro I - Método*". 2012.
- [60] Ministerio de Hacienda y Administraciones Pùblicas. "*PAe - MAGERIT v.3 : Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información*". Disponible en: https://administracionelectronica.gob.es/pae_Home/pae_Documentacion/pae_Metodolog/pae_Magerit.html. Accedido: 5-feb-2024.
- [61] Tiancai He y Zhihua Li. "A Model and Method of Information System Security Risk Assessment based on MITRE ATT&CK". En: *2021 2nd International Conference on Electronics, Communications and Information Technology (CECIT)* (2021), págs. 81-86.
- [62] Jinjie Huang et al. "A method for feature selection based on the correlation analysis". En: *Proceedings of 2012 International Conference on Measurement, Information and Control*. Vol. 1. 2012, págs. 529-532. DOI: [10.1109/MIC.2012.6273357](https://doi.org/10.1109/MIC.2012.6273357).
- [63] Samir Khan et al. "Unsupervised anomaly detection in unmanned aerial vehicles". En: *Applied Soft Computing* 83 (2019), pág. 105650. DOI: <https://doi.org/10.1016/j.asoc.2019.105650>.
- [64] Athar Khodabakhsh et al. "Cyber-risk identification for a digital substation". En: *Proceedings of the 15th International Conference on Availability, Reliability and Security*. Virtual Event Ireland: ACM, 2020, págs. 1-7. DOI: [10.1145/3407023.3409227](https://doi.org/10.1145/3407023.3409227).
- [65] Heejung Kim y Hwankuk Kim. "Comparative Experiment on TTP Classification with Class Imbalance Using Oversampling from CTI Dataset". En: *Security and Communication Networks* 2022 (2022). Ed. por Zhe-Li Liu, págs. 1-11. DOI: [10.1155/2022/5021125](https://doi.org/10.1155/2022/5021125).
- [66] Siwar Kriaa y Yahia Chaabane. "SecKG: Leveraging attack detection and prediction using knowledge graphs". En: *2021 12th International Conference on Information and Communication Systems (ICICS)*. 2021, págs. 112-119. DOI: [10.1109/ICICS52457.2021.9464587](https://doi.org/10.1109/ICICS52457.2021.9464587).
- [67] Yosra Lakhdhar y Slim Rekhis. "Machine Learning Based Approach for the Automated Mapping of Discovered Vulnerabilities to Adversarial Tactics". En: *2021 IEEE Security and Privacy Workshops (SPW)*. 2021, págs. 309-317. DOI: [10.1109/SPW53761.2021.00051](https://doi.org/10.1109/SPW53761.2021.00051).
- [68] Xavier Larriva-Novo et al. "An Approach for the Application of a Dynamic Multi-Class Classifier for Network Intrusion Detection Systems". En: *Electronics* 9.11 (2020), pág. 1759. DOI: [10.3390/electronics9111759](https://doi.org/10.3390/electronics9111759).
- [69] Xavier Larriva-Novo et al. "Efficient Distributed Preprocessing Model for Machine Learning-Based Anomaly Detection over Large-Scale Cybersecurity Datasets". En: *Applied Sciences* 10.10 (2020), pág. 3430. DOI: [10.3390/app10103430](https://doi.org/10.3390/app10103430).

- [70] Xavier Larriva-Novo et al. “Leveraging Explainable Artificial Intelligence in Real-Time Cyberattack Identification: Intrusion Detection System Approach”. En: *Applied Sciences* 13.15 (2023), pág. 8587. DOI: [10.3390/app13158587](https://doi.org/10.3390/app13158587).
- [71] Xavier Larriva-Novo et al. “Simulador de APTs realistas avanzados basado en el marco de MITRE ATT&CK”. En: *Actas de las VIII Jornadas Nacionales de Investigación en Ciberseguridad*. Vigo, 2023, págs. 601-608.
- [72] Victor Lavrenko y Charles Sutton. “*IAML: Dimensionality Reduction*”. Disponible en: <http://www.inf.ed.ac.uk/teaching/courses/iaml/2011/slides/pca.pdf>. Accedido: 13-feb-2024.
- [73] Ioanna Lykourentzou et al. “Ontology-based Operational Risk Management”. En: *2011 IEEE 13th Conference on Commerce and Enterprise Computing*. Luxembourg-Kirchberg, Luxembourg: IEEE, 2011, págs. 153-160. DOI: [10.1109/CEC.2011.18](https://doi.org/10.1109/CEC.2011.18).
- [74] Laurens van der Maaten y Geoffrey Hinton. “Visualizing Data using t-SNE”. En: *Journal of Machine Learning Research* 9.86 (2008), págs. 2579-2605.
- [75] Ahmed M. Mahfouz et al. “Toward A Holistic, Efficient, Stacking Ensemble Intrusion Detection System using a Real Cloud-based Dataset”. En: *International Journal of Advanced Computer Science and Applications* 13.9 (2022). DOI: [10.14569/IJACSA.2022.01309110](https://doi.org/10.14569/IJACSA.2022.01309110).
- [76] Trevor Martin. “On the Need for Collaborative Intelligence in Cybersecurity”. En: *Electronics* 11.13 (2022), pág. 2067. DOI: [10.3390/electronics11132067](https://doi.org/10.3390/electronics11132067).
- [77] Vasileios Mavroeidis y Siri Bromander. “Cyber Threat Intelligence Model: An Evaluation of Taxonomies, Sharing Standards, and Ontologies within Cyber Threat Intelligence”. En: *2017 European Intelligence and Security Informatics Conference (EISIC)*. IEEE, 2017, págs. 91-98. DOI: [10.1109/EISIC.2017.20](https://doi.org/10.1109/EISIC.2017.20).
- [78] Leland McInnes, John Healy y James Melville. “*UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction*”. Disponible en: <https://arxiv.org/abs/1802.03426>. Accedido: 13-feb-2024. 2020.
- [79] Otgonpurev Mendsaikhan et al. *Automatic Mapping of Vulnerability Information to Adversary Techniques*. Disponible en: <https://api.semanticscholar.org/CorpusID:229464970>. Accedido: 5-abr-2024. 2020.
- [80] Yazid Merah y Tayeb Kenaza. “Ontology-based Cyber Risk Monitoring Using Cyber Threat Intelligence”. En: *Proceedings of the 16th International Conference on Availability, Reliability and Security*. Vienna Austria: ACM, 2021, págs. 1-8. DOI: [10.1145/3465481.3470024](https://doi.org/10.1145/3465481.3470024).
- [81] Pedro Mercader y Jack Haddad. “Automatic incident detection on freeways based on Bluetooth traffic monitoring”. En: *Accident Analysis & Prevention* 146 (2020), pág. 105703. DOI: <https://doi.org/10.1016/j.aap.2020.105703>.
- [82] Leila Meshkat y Robert L. Miller. “A Systems Approach for Cybersecurity Risk Assessment”. En: *2022 Annual Reliability and Maintainability Symposium (RAMS)*. Tucson, AZ, USA: IEEE, 2022, págs. 1-9. DOI: [10.1109/RAMS51457.2022.9893966](https://doi.org/10.1109/RAMS51457.2022.9893966).
- [83] Sanatan Mishra. “*Unsupervised Learning and Data Clustering*”. Disponible en: <https://towardsdatascience.com/unsupervised-learning-and-data-clustering-eee78b422a>. Accedido: 13-feb-2024.
- [84] MITRE. Disponible en: <https://www.mitre.org/>. Accedido: 31-mar-2024.

- [85] MITRE ATT&CK. Disponible en: <https://attack.mitre.org/>. Accedido: 1-abr-2024.
- [86] Nour Moustafa y Jill Slay. “UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)”. En: *2015 Military Communications and Information Systems Conference (MilCIS)*. Canberra, Australia: IEEE, 2015, págs. 1-6. DOI: [10.1109/MilCIS.2015.7348942](https://doi.org/10.1109/MilCIS.2015.7348942).
- [87] Mohsin Munir et al. “FuseAD: Unsupervised Anomaly Detection in Streaming Sensors Data by Fusing Statistical and Deep Learning Models”. En: *Sensors* 19.11 (2019). DOI: [10.3390/s19112451](https://doi.org/10.3390/s19112451).
- [88] Natalya F. Noy y Deborah L. McGuinness. “Ontology Development 101: A Guide to Creating Your First Ontology”. En: *Knowledge Systems Laboratory* 32 (2001).
- [89] Nombeko Ntingi et al. “Effective Cyber Threat Hunting: Where and how does it fit?” En: *Proceedings of the 21st European Conference on Cyber Warfare and Security*. Vol. 21. 1. 2022, págs. 206-213. DOI: <https://doi.org/10.34190/eccws.21.1.240>.
- [90] Aybars Oruc, Ahmed Amro y Vasileios Gkioulos. “Assessing Cyber Risks of an INS Using the MITRE ATT&CK Framework”. En: *Sensors* 22.22 (2022), pág. 8745. DOI: [10.3390/s22228745](https://doi.org/10.3390/s22228745).
- [91] *OWL Web Ontology Language - Overview*. Disponible en: <https://www.w3.org/TR/owl-features/>. Accedido: 29-mar-2024.
- [92] Julio-Omar Palacio-Niño y Fernando Berzal. “Evaluation Metrics for Unsupervised Learning Algorithms”. En: *CoRR* abs/1905.05667 (2019). arXiv: [1905.05667](https://arxiv.org/abs/1905.05667).
- [93] Panos Panagiotou et al. “Host-based Intrusion Detection Using Signature-based and AI-driven Anomaly Detection Methods”. En: *Information & Security: An International Journal* 50 (2021), págs. 37-48. DOI: <https://doi.org/10.11610/isij.5016>.
- [94] Sihm-Hye Park y Seok-Won Lee. “Threat-driven Risk Assessment for APT Attacks using Risk-Aware Problem Domain Ontology”. En: *2022 IEEE 30th International Requirements Engineering Conference Workshops (REW)*. Melbourne, Australia: IEEE, 2022, págs. 226-231. DOI: [10.1109/REW56159.2022.00050](https://doi.org/10.1109/REW56159.2022.00050).
- [95] Terence Parr. “Course notes for MSDS621 at Univ of San Francisco, Introduction to Machine Learning: msds621/lectures - GitHub”. Disponible en: <https://github.com/parrt/msds621/tree/master/lectures>. Accedido: 26-mar-2024.
- [96] Pathmind. “Machine Learning Algorithms”. Disponible en: <https://wiki.pathmind.com/machine-learning-algorithms>. Accedido: 26-mar-2024.
- [97] Chaitanya Reddy Patlolla. “Understanding the concept of Hierarchical clustering Technique”. Disponible en: <https://towardsdatascience.com/understanding-the-concept-of-hierarchical-clustering-technique-c6e8243758ec>. Accedido: 13-feb-2024.
- [98] Pellet - Semantic Reasoner. Disponible en: <https://www.w3.org/2001/sw/wiki/Pellet>. Accedido: 30-mar-2024.
- [99] Kai Peng, Victor C. M. Leung y Qingjia Huang. “Clustering Approach Based on Mini Batch Kmeans for Intrusion Detection System Over Big Data”. En: *IEEE Access* 6 (2018), págs. 11897-11906. DOI: [10.1109/ACCESS.2018.2810267](https://doi.org/10.1109/ACCESS.2018.2810267).
- [100] Guo Pu et al. “A hybrid unsupervised clustering-based anomaly detection method”. En: *Tsinghua Science and Technology* 26.2 (2021), págs. 146-153. DOI: [10.26599/TST.2019.9010051](https://doi.org/10.26599/TST.2019.9010051).

- [101] J.R.G. Pulido et al. “Ontology languages for the semantic web: A never completely updated review”. En: *Knowledge-Based Systems* 19.7 (2006), págs. 489-497. DOI: [10.1016/j.knosys.2006.04.013](https://doi.org/10.1016/j.knosys.2006.04.013).
- [102] Dorian Pyle. *Data preparation for data mining*. Morgan Kaufmann Publishers, 1999.
- [103] Sreeraj Rajendran et al. “Unsupervised Wireless Spectrum Anomaly Detection With Interpretable Features”. En: *IEEE Transactions on Cognitive Communications and Networking* 5.3 (2019), págs. 637-647. DOI: [10.1109/TCCN.2019.2911524](https://doi.org/10.1109/TCCN.2019.2911524).
- [104] *Repositorio CAR MITRE*. Disponible en: <https://car.mitre.org>. Accedido: 19-mar-2024.
- [105] *Repositorio CISA DECIDER*. Disponible en: <https://github.com/cisagov/Decider/>. Accedido: 10-abr-2024.
- [106] *Resource Description Framework (RDF): Concepts and Abstract Syntax*. Disponible en: <https://www.w3.org/TR/2004/REC-rdf-concepts-20040210/>. Accedido: 29-mar-2024.
- [107] Marshall S. Rich. “Cyberpsychology: A Longitudinal Analysis of Cyber Adversarial Tactics and Techniques”. En: *Analytics* 2.3 (2023), págs. 618-655. DOI: [10.3390/analytics2030035](https://doi.org/10.3390/analytics2030035).
- [108] Víctor Roman. “Aprendizaje No Supervisado en Machine Learning: Agrupación”. Disponible en: <https://medium.com/datos-y-ciencia/aprendizaje-no-supervisado-en-machine-learning-agrupaci%C3%B3n-bb8f25813edc>. Accedido: 13-feb-2024.
- [109] Peter J. Rousseeuw. “Silhouettes: A graphical aid to the interpretation and validation of cluster analysis”. En: *Journal of Computational and Applied Mathematics* 20 (1987), págs. 53-65. DOI: [https://doi.org/10.1016/0377-0427\(87\)90125-7](https://doi.org/10.1016/0377-0427(87)90125-7).
- [110] Shanto Roy et al. *SoK: The MITRE ATT&CK Framework in Research and Practice*. 2023. arXiv: [2304.07411 \[cs.CR\]](https://arxiv.org/abs/2304.07411).
- [111] Carmen Sánchez-Zas et al. “Design and Evaluation of Unsupervised Machine Learning Models for Anomaly Detection in Streaming Cybersecurity Logs”. En: *Mathematics* 10.21 (2022). DOI: [10.3390/math10214043](https://doi.org/10.3390/math10214043).
- [112] Carmen Sánchez-Zas et al. “Sistema de conciencia cibersituacional y gestióndinámica de riesgos basado en ontologías”. En: *Actas de las VII Jornadas Nacionales de Investigación en Ciberseguridad*. Bilbao, 2022, págs. 91-98.
- [113] Carmen Sánchez-Zas et al. “Desarrollo de una ontología para modelar una metodología interoperable de gestión dinámica de riesgos”. En: *Actas de las VIII Jornadas Nacionales de Investigación en Ciberseguridad*. Vigo, 2023, págs. 339-346.
- [114] Carmen Sánchez-Zas et al. “Ontology-based approach to real-time risk management and cyber-situational awareness”. En: *Future Generation Computer Systems* 141 (2023), págs. 462-472. DOI: <https://doi.org/10.1016/j.future.2022.12.006>.
- [115] Prasad Saripalli y Ben Walters. “QUIRC: A Quantitative Impact and Risk Assessment Framework for Cloud Security”. En: *2010 IEEE 3rd International Conference on Cloud Computing*. Miami, FL, USA: IEEE, 2010, págs. 280-288. DOI: [10.1109/CLOUD.2010.22](https://doi.org/10.1109/CLOUD.2010.22).
- [116] *scikit-learn 1.4.1 - DecisionTreeClassifier*. Disponible en: <https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html>. Accedido: 26-mar-2024.

- [117] *scikit-learn 1.4.1 - GradientBoostingClassifier*. Disponible en: <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html>. Accedido: 26-mar-2024.
- [118] *scikit-learn 1.4.1 - GridSearchCV*. Disponible en: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html. Accedido: 9-abr-2024.
- [119] *scikit-learn 1.4.1 - LabelEncoder*. Disponible en: <https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.LabelEncoder.html>. Accedido: 28-mar-2024.
- [120] *scikit-learn 1.4.1 - RandomForestClassifier*. Disponible en: <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html>. Accedido: 26-mar-2024.
- [121] ANSSI [Agence Nationale de la Sécurité des Systèmes d'information]. “*E BIOS-Risk Manager*”. Disponible en: https://www.ssi.gouv.fr/uploads/2019/11/ansi-guide-ebios_risk_manager-en-v1.0.pdf. Accedido: 5-feb-2024.
- [122] Rawan Al-Shaer, Jonathan M. Spring y Eliana Christou. “Learning the Associations of MITRE ATT&CK Adversarial Techniques”. En: *CoRR* abs/2005.01654 (2020). arXiv: [2005.01654](https://arxiv.org/abs/2005.01654).
- [123] Avi Shaked y Oded Margalit. “OnToRisk – a formal ontology approach to automate cyber security risk identification”. En: *2022 17th Annual System of Systems Engineering Conference (SOSE)*. Rochester, NY, USA: IEEE, 2022, págs. 74-79. DOI: [10.1109/SOSE55472.2022.9812653](https://doi.org/10.1109/SOSE55472.2022.9812653).
- [124] Avi Shaked y Oded Margalit. “Sustainable Risk Identification Using Formal Ontologies”. En: *Algorithms* 15.9 (2022), pág. 316. DOI: [10.3390/a15090316](https://doi.org/10.3390/a15090316).
- [125] Pukkit Sharma. “*The Ultimate Guide to K-Means Clustering: Definition, Methods and Applications*”. Disponible en: <https://www.analyticsvidhya.com/blog/2019/08/comprehensive-guide-k-means-clustering/>. Accedido: 13-feb-2024.
- [126] Yashovardhan Sharma, Simon Birnbach e Ivan Martinovic. “RADAR: A TTP-based Extensible, Explainable, and Effective System for Network Traffic Analysis and Malware Detection”. En: *Proceedings of the 2023 European Interdisciplinary Cybersecurity Conference*. EICC '23. New York, NY, USA: Association for Computing Machinery, 2023, págs. 159-166. DOI: [10.1145/3590777.3590804](https://doi.org/10.1145/3590777.3590804).
- [127] Yashovardhan Sharma et al. “To TTP or not to TTP?: Exploiting TTPs to Improve ML-based Malware Detection”. En: *2023 IEEE International Conference on Cyber Security and Resilience (CSR)*. 2023, págs. 8-15. DOI: [10.1109/CSR57506.2023.10225000](https://doi.org/10.1109/CSR57506.2023.10225000).
- [128] Aishwarya Singh. “*Build Better and Accurate Clusters with Gaussian Mixture Models*”. Disponible en: <https://www.analyticsvidhya.com/blog/2019/10/gaussian-mixture-models-clustering/>. Accedido: 13-feb-2024.
- [129] *SPIN - SPARQL Syntax*. Disponible en: <https://www.w3.org/submissions/spin-sparql/>. Accedido: 30-mar-2024.
- [130] *SPIN: SPARQL Inferencing Notation*. Disponible en: <https://spinrdf.org/>. Accedido: 30-mar-2024.
- [131] International Organization for Standardization. *Risk management — Guidelines (ISO 31000:2018)*. 2018.

- [132] International Organization for Standardization. *Information security, cybersecurity and privacy protection — Information security management systems — Requirements (ISO/IEC 27001:2022)*. 2022.
- [133] SWRL: A Semantic Web Rule Language Combining OWL and RuleML. Disponible en: <https://www.w3.org/submissions/SWRL/>. Accedido: 29-mar-2024.
- [134] Romilla Syed. “Cybersecurity vulnerability management: A conceptual ontology and cyber intelligence alert system”. En: *Information & Management* 57.6 (2020), pág. 103334. DOI: [10.1016/j.im.2020.103334](https://doi.org/10.1016/j.im.2020.103334).
- [135] Zareen Syed et al. “UCO: A Unified Cybersecurity Ontology”. En: *Proceedings of the AAAI Workshop on Artificial Intelligence for Cyber Security* (2016), 8 pages. DOI: [10.13016/M2862BG1V](https://doi.org/10.13016/M2862BG1V).
- [136] Joshua B. Tenenbaum, Vin de Silva y John C. Langford. “A Global Geometric Framework for Nonlinear Dimensionality Reduction”. En: *Science* 290.5500 (2000), págs. 2319-2323. DOI: [10.1126/science.290.5500.2319](https://doi.org/10.1126/science.290.5500.2319).
- [137] Vladimir Vasilyev et al. “Cybersecurity Risk Assessment Based on Cognitive Attack Vector Modeling with CVSS Score”. En: *2021 International Conference on Information Technology and Nanotechnology (ITNT)*. Samara, Russian Federation: IEEE, 2021, págs. 1-6. DOI: [10.1109/ITNT52450.2021.9649191](https://doi.org/10.1109/ITNT52450.2021.9649191).
- [138] Mario Vega-Barbas et al. “AFOROS: A Low-Cost Wi-Fi-Based Monitoring System for Estimating Occupancy of Public Spaces”. En: *Sensors* 21.11 (2021). DOI: [10.3390/s21113863](https://doi.org/10.3390/s21113863).
- [139] Jorge R. Vergara y Pablo A. Estévez. “A review of feature selection methods based on mutual information”. En: *Neural Computing and Applications* 24.1 (2014), págs. 175-186. DOI: [10.1007/s00521-013-1368-0](https://doi.org/10.1007/s00521-013-1368-0).
- [140] E. Vicente, A. Mateos y A. Jiménez-Martín. “Risk analysis in information systems: A fuzzification of the MAGERIT methodology”. En: *Knowledge-Based Systems* 66 (2014), págs. 1-12. DOI: [10.1016/j.knosys.2014.02.018](https://doi.org/10.1016/j.knosys.2014.02.018).
- [141] Vivek Agrawal. “Towards the Ontology of ISO/IEC 27005:2011 Risk Management Standard”. En: *Tenth International Symposium on Human Aspects of Information Security & Assurance (HAISA 2016)*. Frankfurt, Germany, 2016, págs. 101-111.
- [142] Wojciech Widel, Preetam Mukherjee y Mathias Ekstedt. “Security Countermeasures Selection Using the Meta Attack Language and Probabilistic Attack Graphs”. En: *IEEE Access* 10 (2022), págs. 89645-89662. DOI: [10.1109/ACCESS.2022.3200601](https://doi.org/10.1109/ACCESS.2022.3200601).
- [143] Ika Arthalia Wulandari et al. “Ontologies for Decision Support System: The Study of Focus and Techniques”. En: *2018 10th International Conference on Information Technology and Electrical Engineering (ICITEE)*. Kuta: IEEE, 2018, págs. 609-614. DOI: [10.1109/ICITEED.2018.8534947](https://doi.org/10.1109/ICITEED.2018.8534947).
- [144] Soner Yıldırım. *Gradient Boosted Decision Trees-Explained - Towards Data Science*. Disponible en: <https://towardsdatascience.com/gradient-boosted-decision-trees-explained-9259bd8205af>. Accedido: 26-mar-2024.
- [145] Chunhui Yuan y Haitao Yang. “Research on K-Value Selection Method of K-Means Clustering Algorithm”. En: *J (MDPI)* 2.2 (2019), págs. 226-235. DOI: [10.3390/j2020016](https://doi.org/10.3390/j2020016).
- [146] Zeek. Disponible en: <https://zeek.org>. Accedido: 26-feb-2024.

- [147] Leila Zemmouchi-Ghomari. “Ontology and Machine Learning: A Two-Way Street to Improved Knowledge Representation and Algorithm Accuracy”. En: *Proceedings of International Conference on Paradigms of Communication, Computing and Data Analytics*. Springer Nature Singapore, 2023, págs. 181-189.
- [148] Tao Zhang et al. “Comprehensive IoT SIM Card Anomaly Detection Algorithm Based on Big Data”. En: *2019 IEEE International Conferences on Ubiquitous Computing & Communications (IUCC) and Data Science and Computational Intelligence (DSCI) and Smart Computing, Networking and Services (SmartCNS)*. Shenyang, China: IEEE, 2019, págs. 602-606. DOI: [10.1109/IUCC/DSCI/SmartCNS.2019.00126](https://doi.org/10.1109/IUCC/DSCI/SmartCNS.2019.00126).

Apéndice A

Estructura de los catálogos del entorno de conciencia cibersituacional

En este apéndice se utilizan las siguientes abreviaturas:

- Muy Bajo (*Very Low*, VL)
- Bajo (*Low*, L)
- Medio (*Medium*, M)
- Alto (*High*, H)
- Muy Alto (*Very High*, VH)
- Confidencialidad (*Confidentiality*, C)
- Integridad (*Integrity*, I)
- Disponibilidad (*Availability*, A)

Los catálogos que se cargan en el sistema tienen la siguiente estructura. Si los atributos tienen un conjunto de valores limitado, se indicará entre paréntesis.

Información del sistema: Recoge datos relativos al sistema que se evalúa.

{“**id**”: *identificador del sistema*,
“**risk_acceptance_level**”: (*VL, L, M, H, VH*),
“**original_methodology**”: (*none, EBIOS, MAGERIT, MONARC, ITSRM, CRAMM*)}.

Activos: Lista de activos primarios, los activos secundarios relacionados y su valoración en las dimensiones de seguridad confidencialidad, integridad y disponibilidad en una escala del 0 al 10.

```
{“id”: PrimaryAssetID,  
“name”: PrimaryAssetName,  
“type”: TypeOfPrimaryAsset,  
“supporting_assets”:  
[{“id”: SupportingAssetID,  
“name”: SupportingAssetName,  
“type”: TypeOfSupportingAsset,  
“confidentiality”:(0-10),  
“integrity”:(0-10),  
“availability”:(0-10)}]}.
```

Vulnerabilidad: Información sobre las vulnerabilidades asociadas a los activos, y parámetros para identificarlas y evaluar su severidad.

```
{“CVE”: CVE_value,  
“description”: información,  
“type”: typeOfVulnerability,  
“CWE”: [listOfCWE],  
“CPE”: [listOfCPE],  
“CVSS”: (VL, L, M, H, VH),  
“affects”: [listOfSupportingAssetID]}.
```

Amenazas anteriores: Si existen amenazas previas que se deban tener en cuenta, se incluyen especificando su categorización, origen y sobre qué dimensiones de los activos afectan y en qué medida, estimada en valores del 0 al 1.

```
{“id”: identificador,  
“cat_general”: ThreatType,  
“cat_specific”: ThreatSubType,  
“security_dimension”: [(C,I,A)],  
“security_dimension_val” : [listOfImpactValuesOnSecurityDimension (0-1)],  
“origin”: ThreatOrigin,  
“supporting_assets”: [listOfSupportingAssetsIDs]}.
```

Escenarios de riesgo: Definen las posibles situaciones a las que el sistema debe hacer frente (qué activo se ataca y cómo) y qué amenazas genera.

```
{“id”: identificador,  
“primary_asset”: [listOfPrimaryAssetsIDs],  
“supporting_assets”: [listOfSupportingAssetsIDs],  
“attack_pattern”: [listOfCAPECsInvolved],  
“vulnerabilities”: [listOfRelatedVulnerabilities],  
“threat”: [listOfGeneratedThreats (Formato de catálogo de amenaza)],  
“impact”: impactValue,  
“probability”: probabilityValue}.
```

Incidentes de Seguridad: Los incidentes identificados mediante un IDS y, en caso de estar caracterizados, las técnicas y patrones identificados.

```
{“id”: identifier,  
“date”: detectionDate,  
“description”: otherInformation,  
“TTPs”: [listOfIdentifiedTTPs],  
“affected_assets”: [listOfAffectedAssetsIDs],  
“attack_patterns”: [listOfCAPECs]}.  
“
```

Contramedidas: Lista de respuestas frente a los incidentes que hay disponibles en el sistema. Los distintos parámetros permiten compararlas y elegir la óptima para cada situación. Las palabras clave describen la mitigación de manera que permita identificar a qué técnica o amenaza hacen frente.

```
{“id”: identificador,  
“name”: nombre,  
“keywords”: [listOfKeywordsDescribingAction],  
“type”: typeOfCountermeasure,  
“deployment_cost”: costEuros,  
“deployment_effort”: (0-10),  
“impact”: (0-10),  
“installation_complexity”: (0-10),  
“operation_complexity”: (0-10),  
“time_to_be_up_and_running”: timeInMinutes,  
“effectiveness”: (0-10),  
“mitigation_factor”: (0-1),  
“protected_assets”: [listOfAssetsSubTypes]}.  
“
```