Universidad
Rey Juan Carlos

TESIS DOCTORAL

Seguridad Adaptativa: mecanismos y dominios de aplicación

Autor:
Miguel Calvo Matalobos

Directora:
Marta Beltrán Pardo

Programa de Doctorado en Tecnologías de la Información y las Comunicaciones

Escuela Internacional de Doctorado

2023

Universidad
Rey Juan Carlos

TESIS DOCTORAL

Seguridad Adaptativa: mecanismos y dominios de aplicación

Autor:
Miguel Calvo Matalobos

Directora:
Marta Beltrán Pardo

Programa de Doctorado en Tecnologías de la Información y las Comunicaciones

Escuela Internacional de Doctorado

2023

Abstract

The objective of this doctoral thesis is to propose a new adaptive security model based on risk. This model can work in complex, heterogeneous, and changing environments in which different assets coexist. It is based on modifying various security controls and countermeasures to increase or decrease protection or detection levels. In this way, they can adapt themselves to the risk being run at any given time and according to the context in which the monitored assets operate.

To validate the effectiveness of this proposal, two real-use cases have been conducted. Firstly, the model has been implemented in a Cloud Computing environment. In this case, the context and environment of a Web Application Firewall have been monitored to increase or relax the protection level, in relation to the risk of suffering specific attacks. In the second case, it has been applied to a server which includes a solution for security threat detection, adapting according to past malware incidents to prevent them from occurring again.

The results of the use cases have shown that the proposed model can effectively adapt different security controls to the context and risk, no matter how changing they are. Additionally, it improves the security against threats in many assets, requiring only one adaptation capability —the model— to adapt numerous countermeasures.

Resumen

El objetivo de esta tesis doctoral es proponer un nuevo modelo de seguridad adaptativa basado en el riesgo. Este modelo puede funcionar en entornos complejos, heterogéneos y cambiantes en los que coexisten diferentes activos. Se basa en la modificación de diversos controles y contramedidas de seguridad para aumentar o disminuir los niveles de protección o detección. De este modo, pueden adaptarse al riesgo que se corre en cada momento y en función del contexto en el que operen los activos monitorizados.

Para validar la efectividad de esta propuesta, se han llevado a cabo dos casos de uso reales. En primer lugar, se ha implementado el modelo en un entorno *Cloud Computing*. En este caso, se ha monitorizado el contexto y el entorno de un *Web Application Firewall* para aumentar o relajar el nivel de protección, en relación con el riesgo de sufrir ataques específicos. En el segundo caso, se ha aplicado a un servidor que incluye una solución para la detección de amenazas de seguridad, adaptándose en función de incidentes de *malware* pasados para evitar que vuelvan a producirse.

Los resultados de los casos de uso han demostrado que el modelo propuesto puede adaptar, de manera efectiva, diferentes controles de seguridad al contexto y al riesgo, sin importar lo cambiantes que sean. Además, mejora la seguridad contra amenazas en muchos activos, requiriendo solo una capacidad de adaptación —el modelo— para adaptar numerosas contramedidas.

Agradecimientos

Quiero expresar mi más sincero agradecimiento a mi directora de tesis, Marta Beltrán, por el incondicional apoyo que me ha brindado a lo largo de estos años. No solo ha sido una mentora excepcional, sino también mi guía y amiga, y estaré eternamente agradecido por todo lo que ha hecho por mí.

Asimismo, deseo dar las gracias a mi familia, especialmente a mis padres y hermana, por estar siempre a mi lado, creer en mí y ser mi principal fuente de motivación. Gracias a ellos, a la educación que me han brindado, sus sacrificios y sabios consejos, he logrado superar los obstáculos y llegar hasta aquí, convirtiéndome en la persona que soy hoy.

También quiero manifestar mi profundo agradecimiento a mis amigos y amigas, quienes han sido un pilar fundamental en mi vida y una constante fuente de alegría y buenos momentos. En especial, aquellos que han estado a mi lado en tiempos difíciles, aun cuando han pasado largas temporadas sin vernos. Los que, con su apoyo y una simple conversación, logran que las dificultades se desvanezcan.

Finalmente, deseo expresar mi gratitud hacia todas las personas que han formado parte de mi vida en algún momento, incluso aquellas que solo pasaron de largo. Su contribución, aunque aparentemente modesta o eventualmente contraproducente, la considero valiosa y ha dejado una huella imborrable en mi biografía.

Índice general

Índice de tablas

Índice de figuras

Acrónimos

API Application Programming Interface.

APT Advanced Persistent Threat.

BBDD Bases de Datos.

BYOD Bring Your Own Device.

CPU Central Processing Unit.

CU Caso de Uso.

CVE Common Vulnerabilities and Exposures.

CVSS Common Vulnerability Scoring System.

DDoS Distributed Denial of Service.

DFSS Design For Six Sigma.

DL Deep Learning.

DMAIC Define-Measure-Analyze-Improve-Control.

GPS Global Positioning System.

HTTP HyperText Transfer Protocol.

IA Inteligencia Artificial.

ICS Industrial Control Systems.

IDPS Intrusion Detection and Prevention Systems.

IDS Intrusion Detection System.

IIoT Industrial Internet of Things.

IoA Indicator of Attack.

IoC Indicator of Compromise.

IoT Internet of Things.

IP Internet Protocol.

IT Information Technology.

JSON JavaScript Object Notation.

KRI Key Risk Indicator.

LSTM Long Short-Term Memory.

MAC Media Access Control.

MAPE-K Monitor-Analyze-Plan-Execute over a shared Knowledge.

MDE Microsoft Defender for Endpoint.

ML Machine Learning.

MTBF Mean Time Between Failures.

NFV Network Function Virtualization.

OODA Observe-Orient-Decide-Act.

PDCA Plan-Do-Check-Act.

RAM Random Access Memory.

RGPD Reglamento General de Protección de Datos.

RiAS Risk-based Adaptive Security.

SA Seguridad Adaptativa.

SAS Self-Adaptive Systems.

SDN Software Defined Networking.

SIEM Security Information and Event Management.

SOAP Simple Object Access Protocol.

SQL Structured Query Language.

SSH Secure SHell.

SVM Support Vector Machine.

TPM Trusted Platform Module.

TTP Tactic, Technique and Procedure.

URI Uniform Resource Identifier.

URL Uniform Resource Locator.

VPN Virtual Private Network.

WAF Web Application Firewall.

WinRM Windows Remote Management.

Capítulo 1

Introducción y objetivos

En los últimos años, se ha observado un aumento en la sofisticación y la frecuencia de los ataques informáticos, lo que ha provocado que, en determinadas ocasiones, los mecanismos de protección existentes resulten obsoletos e insuficientes para hacer frente a las nuevas amenazas. Estas, en muchos casos, son dinámicas y autoadaptables al entorno en el que se propagan, pudiéndose encontrar entre ellas ataques como «NotPetya», «WannaCry» o «Emotet» [2–4]. Afortunadamente, también ha habido un gran progreso en los controles, protocolos, productos y técnicas para la defensa y protección de los activos informáticos, especialmente frente a los ataques más evolucionados [5–7].

El avance de paradigmas como *Cloud*, *Mobile*, *Fog*, *Edge* e IoT (*Internet of Things*), junto con la evolución en los modelos de desarrollo de software, las infraestructuras de red y los nuevos dominios de aplicación —como la Industria 4.0, las ciudades inteligentes (*Smart Cities*), 5G, robótica y la salud electrónica (e-Salud)—, también han hecho proliferar los ataques y despertar el apetito de los ciberdelincuentes. En muchos casos, se trata de tecnologías poco maduras, en las que se utilizan recursos limitados y, en

ocasiones, encontrándose de forma física al alcance de casi cualquier usuario.

Tradicionalmente, los avances en ciberseguridad se han centrado en lo estático, es decir, en la configuración o parametrización manual o automática de los mecanismos de seguridad en la fase de diseño o implementación de la contramedida. Sin embargo, los nuevos paradigmas mencionados, el cambio de las infraestructuras informáticas y los sofisticados ataques emergentes, han obligado a plantearse una nueva evolución en las capacidades de seguridad para que estas se vuelvan dinámicas o adaptativas.

La Seguridad Adaptativa implica la capacidad de los mecanismos de seguridad de reconfigurarse y adaptarse —si es necesario, en tiempo real— a los cambios en el entorno de los activos protegidos, en función de los riesgos o con las necesidades cambiantes de la organización; en contraposición a las propuestas más tradicionales, que implementan un conjunto fijo de medidas de seguridad. En los últimos años, diversas investigaciones han abordado la Seguridad Adaptativa, proponiendo controles que se aplican o no según la situación, cambios de configuración que se modifican en función del contexto o adicción/reducción de elementos de seguridad según el riesgo que se está corriendo en un determinado momento, entre otras posibilidades. Este enfoque también puede involucrar tecnologías avanzadas como la Inteligencia Artificial o el aprendizaje automático, por ejemplo, para monitorizar y analizar el comportamiento de las amenazas o el contexto y adaptar los controles de seguridad en consecuencia.

La Seguridad Adaptativa permite a las organizaciones detectar y responder rápidamente a las amenazas emergentes o a los cambios inesperados, reduciendo así la probabilidad de que se materialice un incidente de seguridad o minimizando su impacto si se produce. También puede ayudar a las diferentes instituciones a optimizar los recursos destinados a la seguridad, asignándolos

adecuadamente a las áreas de mayor riesgo en cada momento. A pesar de ello, y aunque representa una evolución de los enfoques tradicionales, ofreciendo una protección más efectiva y precisa para los activos, la Seguridad Adaptativa sigue siendo un concepto ambiguo que carece de una definición clara.

Es posible afirmar que los sistemas de Seguridad Adaptativa son altamente ventajosos en términos de protección frente a amenazas sofisticadas, en entornos cambiantes y cuando se manejan diversidad de dispositivos —con características variadas— conviviendo en ellos. Sin embargo, pese a las investigaciones llevadas a cabo en este ámbito, dicha evolución aún no ha alcanzado un nivel de madurez suficiente. En la actualidad, es común encontrarse con controles de seguridad que carecen de la capacidad necesaria para adaptarse a situaciones cambiantes, mientras que los pocos que sí poseen tal competencia suelen incorporarla de manera específica e integrada con el propio control, como se puede apreciar en la figura 1.1.

En este contexto, la complejidad para satisfacer todas las necesidades de las organizaciones, junto con la falta de escalabilidad y el elevado esfuerzo de implementación, hacen que la relación coste-beneficio de utilizar el conjunto de soluciones de Seguridad Adaptativa necesarias no resulte factible para la mayoría de las instituciones. Por esta razón, su manejo o incluso la exploración de este tipo de herramientas suele fracasar. Así pues, es necesario contar con sistemas o mecanismos que permitan la adaptación simultánea de las protecciones de seguridad o contramedidas, sin necesidad de desplegar una solución para cada activo; además, esta debe ser escalable y orientarse hacia la facilidad de uso y despliegue.

Es importante tener en cuenta también que distintas organizaciones, con diferentes características y tamaños, pueden requerir soluciones de Seguridad

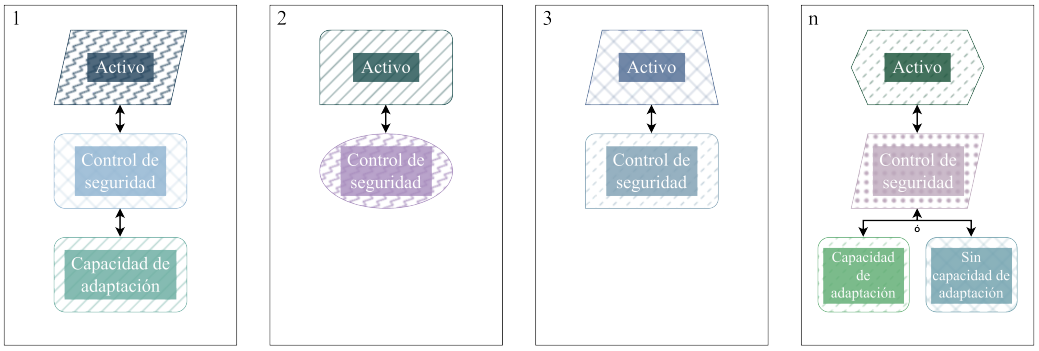

Figura 1.1: Estado actual de las soluciones de Seguridad Adaptativa.

Adaptativa. Las pequeñas empresas, con recursos económicos y humanos limitados, igualmente necesitan mejorar su seguridad y proteger sus activos frente a amenazas, por lo que estas herramientas deben ser accesibles y fáciles de implementar para ellas. Es necesario considerar la diversidad de las instituciones y su capacidad para adoptar este tipo de soluciones, debiendo ser estas flexibles, eficaces, sencillas y, cuando sea posible, reutilizables.

1.1. Hipótesis de partida

Teniendo en cuenta los desafíos que plantea la adaptación en entornos variados y en constante cambio, la hipótesis de partida de esta tesis es la siguiente:

Es posible proponer un modelo de Seguridad Adaptativa basado en el riesgo capaz de hacer frente a las dificultades de adaptación de los controles de seguridad en dominios cambiantes y heterogéneos, sin requerir diferentes mecanismos para proteger todos los activos que pueden convivir en una misma arquitectura o entorno. Además, este modelo puede permitir la reutilización y mejora constantes, y ser aplicable tanto en organizaciones de gran tamaño que necesitan asegurar numerosos activos, como en otras con recursos limitados.

La hipótesis descrita se resume de manera gráfica en la figura 1.2, en contraste con la situación actual mostrada en la figura 1.1. De esta forma, se busca dar

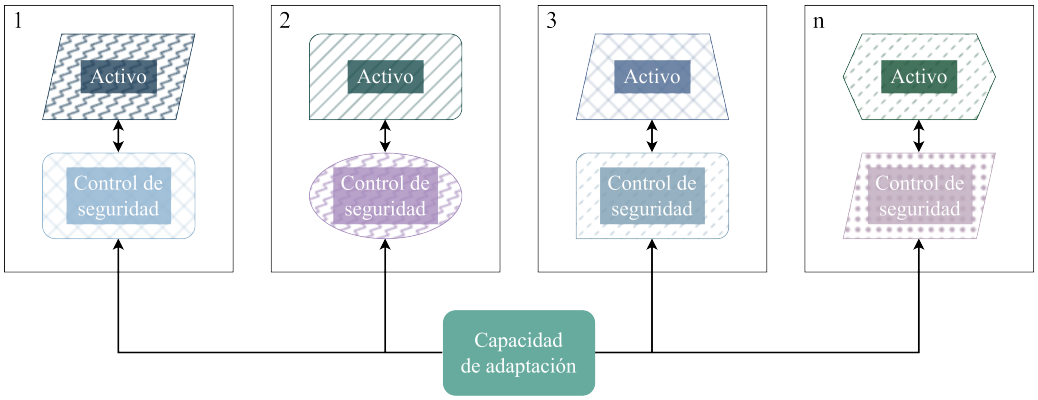

Figura 1.2: Resumen gráfico de la hipótesis de partida.

cabida a los diversos dominios, controles de seguridad o activos que, en la actualidad, carecen de la capacidad de adaptación necesaria para mantener el nivel de protección acorde con el contexto en el que operan o al riesgo en un determinado momento. Asimismo, se consideran aquellos dispositivos que sí disponen de tal capacidad, pero que se encuentran estrechamente ligados al control de seguridad y no pueden utilizarse en otros ámbitos.

1.2. Objetivos

Los objetivos generales de la presente tesis doctoral se encuentran determinados por la hipótesis de partida (≪1.1. Hipótesis de partida≫) y pueden resumirse en los siguientes puntos:

- Diseñar un modelo de Seguridad Adaptativa basado en el riesgo que permita la adecuación de los controles de seguridad a los cambios y heterogeneidades de los entornos en los que opere. Este modelo proporcionará un marco común para la mejora de la seguridad de diversos activos en una misma arquitectura o entorno, sin la necesidad de implementar diferentes capacidades de adaptación para cada uno de ellos.

- Aplicar el modelo sobre distintos controles de seguridad y en múltiples dominios de aplicación y organizaciones, con el objetivo de demostrar su viabilidad y efectividad en términos de mejora constante, eficiencia en el consumo de recursos y reutilización. Se buscará evidenciar su aplicabilidad tanto en instituciones de gran tamaño que necesitan proteger numerosos activos como en otras con recursos más limitados.

Con el propósito de alcanzar los objetivos generales establecidos, se han definido objetivos específicos, que se detallan a continuación:

1. Diseñar un modelo de Seguridad Adaptativa basado en el riesgo:
    - Identificar las deficiencias y limitaciones actuales en el ámbito de la Seguridad Adaptativa, considerando los trabajos previos en el estado del arte.
    - Proponer un modelo de Seguridad Adaptativa basado en el riesgo que cumpla con los siguientes requisitos: (1) se pueda emplear en organizaciones de diversos tamaños y capacidades; (2) se enfoque en la reutilización y facilidad de uso; (3) permita la adaptación simultánea de diferentes controles de seguridad o activos en distintos dominios; (4) no requiera grandes capacidades de cómputo.
    - Definir el modelo propuesto de manera que: (1) se facilite su gestión, pudiendo ser operado por diferentes roles o administradores encargados de tareas específicas; (2) esté formado por capas bien diferenciadas que agrupen los distintos mecanismos necesarios para su funcionamiento, facilitándose así la distribución de cada una de ellas en diversos emplazamientos (por ejemplo, todo desplegado en un único servidor —centralizado— o cada una de las capas en uno diferente —distribuido—); (3) englobe la posibilidad de

ejecutarse tanto de manera local como remota (añadiendo con ello una mayor versatilidad y flexibilidad para las organizaciones); (4) permita la adaptación de diferentes controles de seguridad de forma simultánea, sin importar sus características.

2. Aplicar el modelo sobre diversos controles de seguridad y en diferentes dominios de aplicación y organizaciones:

   - Implementar un primer prototipo que ejecute el modelo de Seguridad Adaptativa basado en el riesgo propuesto, de tal forma que cumpla con los requisitos y especificaciones determinadas durante su diseño.
   - Desplegar o utilizar entornos realistas o reales donde el modelo pueda ser aplicado para mejorar la seguridad de una organización, un conjunto de activos o un activo en particular. En este sentido, principalmente se valorará la adaptación de capacidades de protección, pero también se explorará la posibilidad de incluir las de detección.
   - Integrar el modelo de Seguridad Adaptativa en la arquitectura o arquitecturas existentes o desplegadas para su evaluación, realizando, si es necesario, los complementos o modificaciones pertinentes.
   - Comprender el contexto operativo del activo, conjunto de activos u organización sobre el cual se validará el modelo para configurarlo según las necesidades específicas.
   - Verificar y medir los resultados obtenidos en términos de consumo de recursos, tiempos y cualquier otro indicador que pueda ser útil para determinar la validez y rendimiento del modelo presentado en la tesis.

1.3. Metodología

A continuación se describe la metodología propuesta para validar la hipótesis de partida («1.1. Hipótesis de partida») y lograr los objetivos establecidos («1.2. Objetivos») en la presente tesis doctoral:

- Investigación y análisis exhaustivo de los aspectos más relevantes relacionados con el estado del arte en materia de Seguridad Adaptativa y sus dominios de aplicación, incluyendo las capacidades, dificultades encontradas y otros atributos o características que ayuden a comprender, de manera completa, el tema principal de la tesis.

- Determinación de las categorías existentes en materia de Seguridad Adaptativa, con el objetivo de catalogar las diferentes investigaciones del estado del arte, prestando atención a distintos aspectos de estas soluciones, como el desencadenante, el mecanismo de decisión o la propia adaptación.

- Selección de los requisitos y necesidades específicos para el nuevo modelo de Seguridad Adaptativa, teniendo en cuenta las dificultades encontradas durante la investigación y el análisis del estado del arte.

- Propuesta detallada del modelo de Seguridad Adaptativa, especificando exhaustivamente los elementos y componentes que lo conforman, su funcionamiento y las diferentes fases por las que puede pasar. Además, se identificarán, si fuera necesario, los roles o gestores encargados de las tareas de administración requeridas por el modelo.

- Implementación de un prototipo que ejecute el modelo de Seguridad Adaptativa, teniendo en cuenta tanto los requisitos y necesidades definidos previamente, como la propuesta detallada. Además, se proporcio-

narán códigos o ejemplos de los elementos o componentes que requieran una mayor clarificación.

- Validación del modelo mediante casos de uso realistas para certificar tanto su correcto funcionamiento como su utilidad. Estos casos de uso se seleccionarán cuidadosamente, teniendo en cuenta las necesidades de seguridad de diferentes organizaciones.

- Evaluación del rendimiento del modelo propuesto a través de distintas mediciones y experimentos.

1.4. Estructura del documento

Tras la introducción realizada en el presente capítulo, el resto de la tesis doctoral se estructura de la siguiente manera:

- En el **capítulo 2**, se realiza un análisis exhaustivo del estado del arte en materia de Seguridad Adaptativa. Para ello, se proporciona contexto sobre el tema tratado y se proponen diferentes categorizaciones de los trabajos de investigación relacionados, centrándose en el desencadenante, el mecanismo de decisión y la adaptación. Asimismo, se detallan las lógicas de adaptación más relevantes y se analizan con mayor grado de detalle los diferentes trabajos previos del estado del arte sobre Seguridad Adaptativa, exponiendo sus características concretas.

- El **capítulo 3** presenta el modelo propuesto en esta tesis, donde se justifica la necesidad de la solución. Después, se define la arquitectura subyacente y se especifican los supuestos de partida. Asimismo, se describe el flujo de adaptación seguido por el modelo, incluyendo las explicaciones necesarias con respecto a los diferentes componentes y

capas que lo forman, así como las fases por las que pasa durante su funcionamiento.

- El **capítulo 4** detalla los dos casos de uso utilizados para validar el modelo de Seguridad Adaptativa. En primer lugar, se describe la arquitectura, el prototipo y los resultados experimentales de la utilización del modelo para adaptar una capacidad de protección. Posteriormente, en el segundo caso de uso, se exponen los datos correspondientes a la adaptación de una capacidad de detección.

- Por último, el **capítulo 5** completa esta tesis doctoral mostrando las diferentes conclusiones obtenidas tras su realización. Además, se incluyen posibles líneas de trabajo futuro relacionadas con la investigación realizada.

Capítulo 2

Estado del arte

En el presente capítulo se expone el estado del arte en materia de Seguridad Adaptativa para la protección de activos informáticos, así como aquellos conceptos y descripciones relevantes o necesarios para la comprensión de la investigación desarrollada en esta tesis.

En primer lugar, se realiza una pequeña introducción exponiendo la importancia de la Seguridad Adaptativa y por qué se ha evolucionado hacia ella en los últimos años, así como la definición de esta y los conceptos clave necesarios para entender las siguientes secciones. Después, se proponen diferentes catalogaciones de la Seguridad Adaptativa: atendiendo al desencadenante de la adaptación, al mecanismo de decisión o a la adaptación propiamente dicha. También se analizan las distintas lógicas de adaptación, destacando las más utilizadas tanto en mecanismos y propuestas de Seguridad Adaptativa, como, de forma más genérica, aquellas empleadas en otros ámbitos de la ciberseguridad. Por último, se lleva a cabo un amplio análisis de los diferentes trabajos previos en la materia, remarcando sus características más relevantes y comparándolos entre ellos.

2.1. Contexto

Investigadores y fabricantes se han visto obligados a plantear una evolución en las capacidades y herramientas para la mejora de la seguridad debido a diversos factores, lo que ha resultado en un mayor dinamismo. El empuje viene dado, en primer lugar, por el gran avance y la aparición de nuevas tecnologías y paradigmas como la computación en la nube (o *Cloud Computing*) [8], la computación de niebla (o *Fog Computing*), la computación de borde (o *Edge Computing*) [9], los dispositivos móviles (o *Mobile*), el IoT (*Internet of Things* o Internet de las cosas), el IIoT (*Industrial Internet of Things* o Internet de las cosas industrial) [10, 11], la IA (Inteligencia Artificial), el *Big Data* [12], el DL (*Deep Learning* o aprendizaje profundo) o el ML (*Machine Learning* o aprendizaje automático). Por otro lado, también ha influido la transformación en las infraestructuras de red, cada vez más definidas por software (mediante *Software Defined Networking* —SDN— y *Network Function Virtualization* —NFV—). Además, este cambio ha sido impulsado por el auge de nuevos dominios de aplicación y sus requisitos en términos de calidad de servicio y experiencia, como, por ejemplo, la Industria 4.0 [13], las ciudades inteligentes [14, 15], la salud electrónica, el 5G o la robótica. Otra razón importante de esta variación es la heterogeneidad de los entornos, así como la proliferación de nuevas amenazas y ataques, siendo estos cada vez más dirigidos, cambiantes y avanzados, en los que se invierte gran cantidad de recursos para llevarlos a cabo [16–22].

Este dinamismo en las capacidades y herramientas para el incremento de la seguridad implica que dichos elementos estén diseñados para reconfigurarse y adaptarse —en ocasiones en tiempo real— al contexto operativo, acorde con las amenazas actuales o conforme al riesgo tolerado en un determinado momento. En este sentido, diferentes investigaciones han abordado la Segu-

ridad Dinámica o *Dynamic Security* en los últimos años [23], proponiendo herramientas o controles que pueden aplicarse en función de la situación en un momento concreto (realizando cambios de configuración dependientes del entorno, como en [24]) o tomando unas u otras decisiones según el riesgo que se está corriendo (ofreciendo, por ejemplo, unos mecanismos de autenticación u otros [25]). A pesar de lo anteriormente expuesto, la Seguridad Dinámica es un concepto aún bastante ambiguo e indefinido; de hecho, esta idea también puede ser identificada mediante otros nombres como, por ejemplo, Seguridad Adaptativa (SA) o *Adaptive Security*.

La Seguridad Dinámica o Seguridad Adaptativa es un modelo de seguridad con la capacidad de analizar comportamientos, características, estados y eventos tanto dentro como fuera del activo que debe protegerse. Estas observaciones del entorno o el contexto se utilizan para mejorar la seguridad de determinados recursos o activos frente a las amenazas antes de que se produzcan o cuando se están materializando. Con ello permiten, normalmente de forma automática, realizar diferentes acciones como, por ejemplo, aumentar o disminuir la protección, repararse, optimizarse o reconfigurarse. En la mayoría de las soluciones de SA existentes, la eficacia de sus mecanismos o el momento de aplicación de estos, mejora a medida que pasa el tiempo, ya que se nutren de los sucesos o situaciones que ya han ocurrido en el pasado (o bien en el propio entorno que protegen o en otros de características similares) [26, 27].

Cabe destacar que esta idea —la Seguridad Adaptativa— tiene el objetivo de frenar amenazas activas, así como reducir los posibles vectores de ataque. Para ello, engloba mecanismos capaces de: (1) predecir, por ejemplo, pronosticando un ataque o una posible amenaza; (2) prevenir, aplicando cambios que mejoren la seguridad o disminuyan la superficie de exposición; (3) detectar, descubriendo vulnerabilidades, observando un incremento del riesgo o averiguando incidentes de seguridad existentes para actuar en consecuencia;

o (4) responder, llevando a cabo modificaciones en el diseño del activo o conjunto de activos o incorporando nuevas herramientas de seguridad cuando los ataques se hayan perpetrado o cuando estén haciéndolo [28].

Un concepto muy extendido al hablar de dinamismo o adaptación es el de Sistemas Adaptativos, Sistemas Auto-Adaptativos o SAS (*Self-Adaptive Systems*). Estas soluciones son las empleadas para aplicar la Seguridad Adaptativa, por ello, tienen la capacidad de modificar su composición, arquitectura o comportamiento (o la de otro activo) en respuesta a su estado o la percepción del contexto operativo. Este contexto operativo u operacional se puede definir como la combinación de condiciones o circunstancias bajo las que un activo o conjunto de activos se espera que opere. Hoy en día, existen diferentes definiciones sobre el concepto de Sistemas Adaptativos, aunque las más extendidas son las que comprenden las herramientas, mecanismos o propuestas que tienen la capacidad de conocer su entorno y adaptarse según los cambios que se produzcan y de acuerdo con unos objetivos prefijados o deducidos [29–31].

Tal y como se presenta en [32] y [33], y orientado hacia el ámbito de la Seguridad Adaptativa, para conocer los requisitos de un Sistema Adaptativo pueden plantearse las siguientes cuestiones: (1) dónde se debe llevar a cabo el cambio o la adaptación; (2) cuándo se debe aplicar la adaptación; (3) qué cambios es necesario aplicar; y (4) por qué se requiere dicho cambio.

En las siguientes secciones, se presentan y analizan distintos trabajos de Seguridad Adaptativa, clasificándolos en tres categorizaciones, atendiendo a diferentes aspectos. La dificultad de esta catalogación radica en que, aunque las propuestas y soluciones actuales están ampliamente diferenciadas por el modo y el motivo por el que se toman las decisiones y avaladas por los estudios e investigaciones más relevantes en relación con ellas, a menudo

aparecen juntas, son difícilmente diferenciables o se presentan sin atender a un estándar o unos parámetros fundamentados, decidiéndose su categoría según los intereses o las percepciones de los investigadores e incluso catalogando como adaptativas o auto-adaptativas algunas propuestas que, por su naturaleza, no deberían considerarse así [34]. La figura 2.1 muestra, de forma concisa y para una mejor comprensión, las categorizaciones que se proponen en esta tesis junto a las categorías que las conforman.

La primera categorización (SA atendiendo al desencadenante de la adaptación), explicada en detalle en próximas secciones, tiene en cuenta la forma en la que se recopila la información necesaria para aplicar la adaptación, existiendo en ella dos categorías: (1) la Seguridad Adaptativa sensible al contexto (cuyo desencadenante se calcula a partir de información extraída del contexto operativo) y (2) la Seguridad Adaptativa basada en riesgos (que tiene en cuenta diferentes riesgos calculados para adaptarse).

La segunda clasificación (SA atendiendo al mecanismo de decisión), también presentada en detalle a continuación, se centra en el mecanismo o la tecnología empleada para tomar la decisión de adaptación; en este caso, los posibles tipos son tres: (1) Seguridad Adaptativa predefinida (cuando la decisión se basa en la elección o las pautas previamente definidas por un operador humano), (2) Seguridad Adaptativa inteligente (cuando la decisión de adaptación es totalmente autónoma, determinada por el propio sistema) y (3) Seguridad Adaptativa consciente (si la decisión se toma de forma semi-autónoma, combinando elecciones humanas con las del propio sistema).

En la tercera categorización (SA atendiendo a la adaptación), se enfatiza en la acción que se lleva a cabo cuando el sistema de Seguridad Adaptativa ha tomado la decisión, pudiendo ser estas adaptaciones de diversa índole:

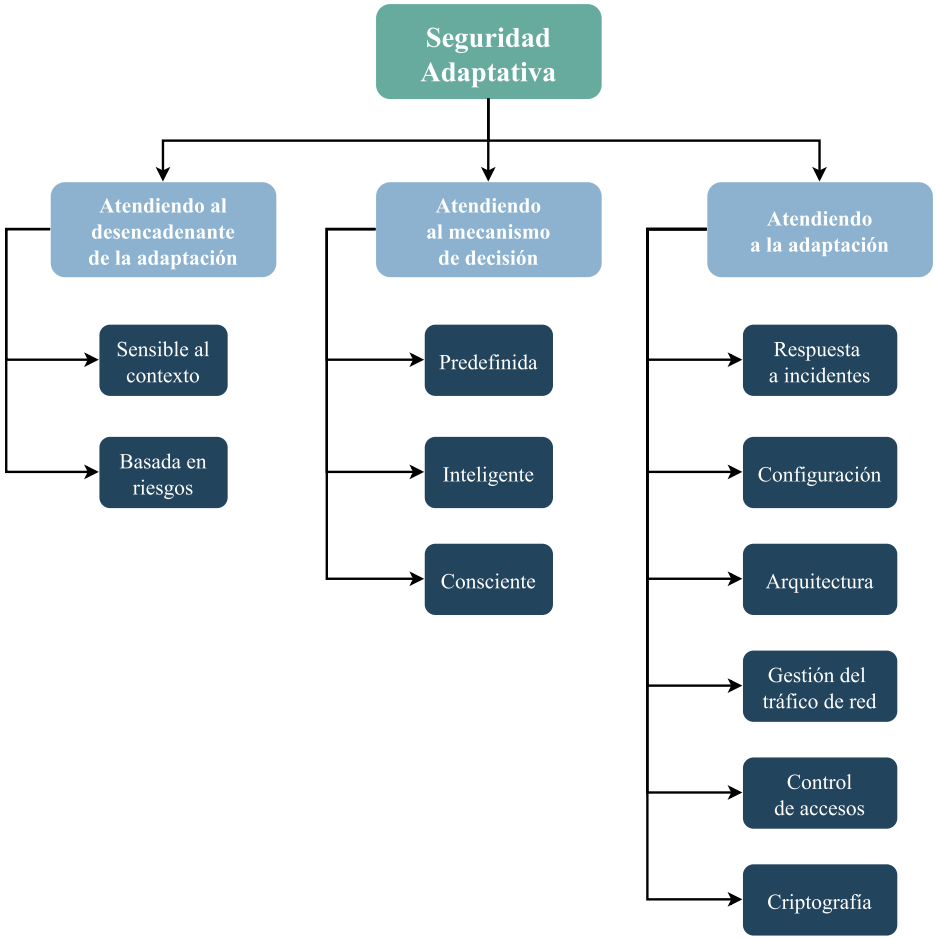

Figura 2.1: Catalogaciones y categorías de la Seguridad Adaptativa.

(1) Respuesta a incidentes (si la adaptación consiste en informar o aconsejar sobre un incidente), (2) Configuración (si la modificación que se aplica es sobre la configuración o la forma de trabajar de los activos o el mecanismo de seguridad), (3) Arquitectura (cuando la adaptación se realiza cambiando aspectos relacionados con la arquitectura del sistema o los activos), (4) Gestión del tráfico de red (en los casos en los que se decide sobre diferentes aspectos de la red o el tráfico que circula por ella), (5) Control de accesos (si la adaptación tiene que ver con la gestión de identidades y accesos) y (6) Criptografía (en situaciones donde se ajustan aspectos criptográficos).

2.2. SA atendiendo al desencadenante de la adaptación

En esta primera categorización de la Seguridad Adaptativa se ha decidido desglosar los trabajos, investigaciones y sistemas dinámicos atendiendo a la información que se requiere para, tras ser recopilada y procesada, llevar a cabo la adaptación. Dicho de otra manera, las categorías aquí descritas se estudian teniendo en cuenta las métricas o mediciones que sirven a la lógica para tomar una decisión antes de realizar la adaptación propiamente dicha.

De esta forma, tras un análisis exhaustivo de las diferentes propuestas estudiadas, se ha determinado que estos datos, que sirven como desencadenante de la adaptación, pueden ser principalmente de dos tipos. El primero, atendiendo al contexto (Seguridad Adaptativa sensible al contexto), recopilando información del entorno o el contexto operativo del activo o activos que deben protegerse (o de otros entornos/contextos que pueden afectar directa o indirectamente a su seguridad); en esta categoría, las métricas y mediciones recogidas no sufren ninguna modificación o tratamiento y se analizan tal y como se reciben, siendo ellas mismas o un conjunto de ellas y sus valores el desencadenante para aplicar la adaptación, en ocasiones cuando superan o no alcanzan ciertos umbrales y otras simplemente por el hecho de recibirse. En la segunda categoría se engloban aquellos trabajos que se apoyan en los riesgos para la transformación (Seguridad Adaptativa basada en riesgos), procesando y tratando las métricas recibidas (que en la mayoría de los casos se corresponden con datos de contexto) para medir el riesgo; de esta forma, se consigue conocer el nivel de riesgo que está corriendo un activo, conjunto de activos o incluso la organización al completo en un momento determinado y, así, poder actuar en consecuencia.

2.2.1. Seguridad Adaptativa sensible al contexto

La primera categoría de la Seguridad Adaptativa atendiendo al desencadenante de la adaptación se corresponde con la Seguridad Adaptativa sensible al contexto o *Context-aware Security*. Se trata de un enfoque basado en el conocimiento del entorno y engloba aquellas soluciones, herramientas y propuestas capaces de recopilar información de este, pero también de eventos o actuaciones que permiten valorar o analizar si se debe llevar a cabo una adaptación de la arquitectura o el comportamiento de un activo. Toda esta información, reunida y procesada de forma dinámica, puede ofrecer una mayor seguridad, más robusta y precisa que las técnicas estáticas habituales, empleándose en diferentes áreas de aplicación [35, 36].

En la SA sensible al contexto, la información del entorno recopilada puede ser del propio activo que debe protegerse, pero también de otros activos o fuentes que se consideran de valor para determinar la necesidad de cambio. Además, es posible tratar o combinar la información —tal y como se verá en la siguiente sección, SA atendiendo al mecanismo de decisión— antes de evaluar si se debe realizar una modificación o no. Para ello se utilizan, por ejemplo, datos como el estado de la batería de un teléfono [37], información del tráfico de red [38], el identificador de un dispositivo [39] y un largo etcétera de posibles variables que se cogen como referencia para tomar decisiones de seguridad. Sin embargo, es necesario mencionar que estas mediciones no siguen ningún patrón o estándar, siendo heterogéneas y concretas, adaptadas al problema que se pretende resolver, e incluso no especificándose en gran parte de los trabajos analizados.

Por otro lado, la adaptación sirve para incrementar o disminuir los niveles de seguridad y evitar o paliar, de esta forma, ataques que se están produciendo

o que pueden llegar a producirse, pero también se han encontrado soluciones que no llevan a cabo ninguna adaptación, por ejemplo, teniendo como resultado el envío de alertas; estas propuestas se han tenido en cuenta por su cercanía, en los demás componentes, al resto de soluciones de Seguridad Adaptativa sensible al contexto encontradas, ya que analizan el entorno y lo valoran, aunque sin llegar a realizar ninguna acción o siendo la única el envío de una alerta como en [40] y [41] o el muestreo de información, con [42] como referencia. Algunos ejemplos de adaptación de esta categoría son la renovación de claves de cifrado [43], la decisión de acceso al dispositivo [44] o la autoadaptación de políticas de control de acceso [45]; no obstante, estos trabajos son catalogados y ampliados en la sección «2.4. SA atendiendo a la adaptación».

El concepto de Seguridad Adaptativa sensible al contexto aparece, en la mayoría de los casos, aplicado para la gestión de identidades y accesos ([37, 43–60]). También puede encontrarse un gran número de propuestas en escenarios de detección de incidentes como amenazas, anomalías, intrusiones o ataques ([38, 40–42, 61–74]). Sin embargo, estos no son los únicos entornos en los que se emplea, también pueden encontrarse herramientas relacionadas con la definición de la arquitectura y la configuración de la red ([24, 39, 75–81]), con la distribución de tareas ([82]) e incluso con el cifrado ([43, 44, 51, 57, 60]).

Los detalles de las soluciones enumeradas en esta sección se amplían en «2.6. Comparativa de trabajos previos».

2.2.2. Seguridad Adaptativa basada en riesgos

En segundo lugar, atendiendo al desencadenante de la adaptación, se encuentra la Seguridad Adaptativa basada en riesgos o *Risk-based Security*. Esta

categoría incluye aquellos mecanismos, herramientas o propuestas capaces de calcular el riesgo que corren los activos, un conjunto de ellos o la organización para, tras comprobar si se encuentra dentro de los umbrales tolerados o pre-establecidos, ejecutar la adaptación o llevar a cabo la acción que corresponda. En este sentido, y a diferencia de la categoría Seguridad Adaptativa sensible al contexto, la información recopilada (que puede ser de diversa índole y depende de la solución) siempre se trata para extraer, a partir de ella, un nivel de riesgo.

Este paradigma logra identificar los diferentes riesgos de cada uno de los activos de IT (*Information Technology*) de una organización, dando prioridad al coste de mitigación (en términos económicos, de personal, pérdida de reputación, etc.), lo que significa reducir esos riesgos a un nivel aceptable para un momento concreto [83]. El método expuesto brinda objetivos de seguridad prácticos y realistas, ofreciendo y permitiendo a las organizaciones encontrar siempre un equilibrio entre la inversión y la seguridad [84, 85].

Para alcanzar una Seguridad Adaptativa basada en el riesgo eficaz, es necesario monitorizar y evaluar constantemente los riesgos, así como planificarlos cuidadosamente desde el principio. También es imprescindible identificar los activos y las amenazas que pueden surgir en ellos, registrar las vulnerabilidades existentes y crear los diferentes perfiles de riesgo de cada activo o grupo de activos, asignándole una puntuación (en función de los controles y mecanismos de seguridad disponibles para su protección) [86]. Es necesario además acordar cómo se tratará este riesgo cuando surja (declarando siempre los motivos que llevaron a la decisión). El principal desafío en estas asignaciones suele ser calcular el riesgo y traducir la puntuación obtenida en una decisión coherente y útil para la organización [87].

Este tipo de soluciones suele apoyarse en ciertas mediciones o métricas que,

tras procesarse empleando técnicas muy dispares, determinan si se lleva a cabo la adaptación o, por el contrario, no es necesario aplicar ningún cambio. En todos los estudios analizados, estas métricas han resultado ser muy específicas y orientadas a la solución concreta que se expone; sin existir un catálogo o un marco que permita la creación o elección de estas, dependiendo del riesgo que se quiere medir o para qué se necesitan estas mediciones. Algunos ejemplos de información recopilada para su evaluación es la reputación de un sujeto [88], la localización [89] o el *fingerprint* del navegador [90].

No existe una fórmula común para el cálculo del riesgo; cada propuesta o trabajo determina su propia regla. Generalmente, suelen centrarse en el uso de la fórmula de facto para el cálculo del riesgo de forma cuantitativa, la cual indica que el riesgo («R») se puede calcular como la probabilidad («P») de que ocurra un incidente de seguridad por el impacto («I») que este tendrá en la organización o el activo ($R = P * I$).

Al igual que ocurre en la categoría de Seguridad Adaptativa sensible al contexto, tras el análisis del riesgo y la determinación de la adaptación, esta sirve para aumentar o disminuir el nivel de seguridad, evitando o paliando con ello los posibles ataques que se están produciendo o pueden producirse. La gran variedad de acciones y adaptaciones de esta categoría va desde los cambios en la configuración de dispositivos y la arquitectura ([56, 75, 76, 91]) a la decisión de acceso al equipo o el recurso ([44, 52, 58, 60, 92–95]), pasando por el envío de alertas de seguridad ([40, 69]) o la determinación de permitir la acción solicitada o no ([96]). Estas acciones o adaptaciones se analizan en detalle y se categorizan en la sección «2.4. SA atendiendo a la adaptación».

La aplicación más recurrente en esta categoría es la gestión de identidades y accesos ([25, 88–90, 92–106]). Además, se encuentran algunas propuestas para la detección de incidentes como amenazas o intrusiones ([107–109]). Otro

ámbito de la SA basada en riesgos, aunque menos explorado, es la definición de la arquitectura o la configuración de red ([110]).

Los trabajos e investigaciones citadas en esta sección se profundizan en «2.6. Comparativa de trabajos previos», donde se especifican sus características más relevantes.

2.3. SA atendiendo al mecanismo de decisión

A diferencia de la primera catalogación (SA atendiendo al desencadenante de la adaptación), que observa las métricas o mediciones que inician el proceso de adaptación, en esta segunda clasificación, se exploran las diversas investigaciones, trabajos y sistemas, observando y considerando el modo o la forma en la que se decide si se debe llevar a cabo esa adaptación o no. Es decir, se tiene en cuenta quién (o qué) y cómo se determina si el sistema debe aplicar algún cambio.

Así, tras examinar los diferentes trabajos en la materia, se llega a la conclusión de que pueden darse tres categorías bien diferenciadas atendiendo a este mecanismo de decisión. La primera clase que puede encontrarse es la Seguridad Adaptativa predefinida; esta categoría engloba aquellas soluciones que requieren de la intervención humana previa para definir, por ejemplo, ciertas reglas, políticas o configuraciones que deben seguirse a la hora de decidir si se debe adaptar o no. En segundo lugar, puede hablarse de la Seguridad Adaptativa inteligente; en ella, las decisiones de adaptación son determinadas por un algoritmo (normalmente previamente entrenado, como es el caso de las soluciones apoyadas en el aprendizaje automático, el aprendizaje profundo o la inteligencia artificial), sin intervención humana directa a la hora de determinar si algo debe hacerse. Por último, se encuentra la categoría Seguridad

Adaptativa consciente; en este caso se incorporan las propuestas, que, a pesar de tener cierto grado de inteligencia o decisión autónoma —por parte de un algoritmo—, también requieren de la intervención humana previa, por ejemplo, las herramientas que emplean lógica difusa o aquellas que combinan aprendizaje automático, aprendizaje profundo, IA, etc. con políticas, reglas o configuraciones manuales.

2.3.1. Seguridad Adaptativa predefinida

Centrándose en el mecanismo de decisión, el primer enfoque que puede encontrarse es la Seguridad Adaptativa predefinida. Esta tiene que ver con aquellas soluciones, herramientas e investigaciones que emplean configuraciones, reglas o políticas preestablecidas para indicar, de acuerdo con las métricas y mediciones recibidas, si se debe llevar a cabo alguna acción y cuál será esta. Para llegar a estos términos y que los sistemas funcionen correctamente, es necesaria la intervención y los conocimientos de los operadores humanos.

Hace algún tiempo, los analistas de seguridad realizaban un examen manual de todos los datos e información que pudiera ser útil para reforzar la seguridad de la organización o de determinados activos. Aunque la Seguridad Adaptativa predefinida ha supuesto un gran avance para la protección, sigue requiriendo de esta intervención humana para definir tanto las métricas y mediciones relevantes (como ocurre en cualquier categoría de Seguridad Adaptativa), como el «qué» ocurrirá cuando esas métricas sean de un determinado tipo, superen ciertos umbrales o el riesgo calculado a partir de ellas no esté entre los límites que el operador haya considerado previamente.

Los trabajos contenidos en esta categoría, una vez recibidas las mediciones correspondientes, aumentan o disminuyen la fortaleza de sus mecanismos de

seguridad atendiendo a las políticas y reglas ([24, 37, 44–46, 48, 50–52, 56–61, 64, 70, 71, 75–82, 88, 91, 95, 97–99, 101–106, 109–111]), al contenido de bases de datos ([39, 46, 54, 105]) o a otras configuraciones o algoritmos concretos ([42, 43, 56, 62, 76, 88–90, 96–98, 101–103, 107, 108]) que los operadores o administradores han elaborado con valores predeterminados. Estas decisiones preestablecidas suelen ser permanentes, por lo que, ante los mismos valores de las métricas o tras el cálculo del riesgo, siempre efectúan idénticas acciones. Su facilidad de uso y de configuración, así como la de desarrollo, la ha convertido en la categoría de SA atendiendo al mecanismo de decisión más extendida.

La acción que se realiza tras la toma de la decisión no difiere en absoluto de otras categorías de SA atendiendo al mecanismo de decisión. De hecho, al existir más trabajos relacionados, la variedad es aún mayor que en sus categorías hermanas. En este caso, la más destacada es la decisión de acceso al recurso o dispositivo ([44, 48, 52, 54, 57–60, 88, 90, 95, 97–99, 101, 102, 104–106, 111]), pero también es posible encontrar investigadores que se han centrado en determinar qué hacer o qué no con el tráfico de red ([39, 77–79, 82, 89, 103, 109]) o en renovar la clave de cifrado ([43]), por poner solo algunos ejemplos. Estas adaptaciones son analizadas y catalogadas con mayor grado de detalle en la sección «2.4. SA atendiendo a la adaptación».

Poniendo la vista en el ámbito de aplicación, destaca la enorme cantidad de trabajos relacionados con la gestión de identidades y accesos ([37, 43–46, 48, 50–52, 54, 56–60, 88–90, 95–99, 101–106, 111]). Sin embargo, también es reseñable su uso para redefinir la arquitectura o la configuración de la red ([24, 39, 75–81, 109, 110]) o para la detección de incidentes como amenazas, anomalías, intrusiones o filtraciones de información ([42, 61, 62, 64, 70, 71, 107, 108]). Además, se encuentran mecanismos orientados al cifrado ([43, 44, 51, 57, 60, 106]) o a la distribución de tareas ([82]).

En la sección «2.6 Comparativa de trabajos previos» se analizan las investigaciones aquí expuestas con un mayor grado de detalle.

2.3.2. Seguridad Adaptativa inteligente

Por otro lado, la segunda categoría que atiende al mecanismo de decisión se corresponde con la Seguridad Adaptativa inteligente, en inglés *Intelligent Adaptive Security*. En esta categoría se engloban aquellas propuestas que emplean técnicas y herramientas novedosas como el *Big Data*, *Analytics*, la inteligencia artificial, el aprendizaje profundo, el aprendizaje automático o la gestión de información y eventos de seguridad (*Security Information and Event Management* o SIEM) para detectar anomalías, valores atípicos o desviaciones de los comportamientos estándar y actuar en consecuencia.

La Seguridad Adaptativa inteligente recopila, estandariza y analiza los datos generados por redes, aplicaciones, bases de datos (BBDD), registros y otros elementos de la infraestructura de IT en tiempo real, sin necesidad de la intervención humana —salvo, como es lógico, para entrenar los diferentes modelos—. Esta información, una vez recopilada, se evalúa y procesa (a través de procedimientos y mecanismos como la IA, el ML o el reconocimiento de patrones) para traducir los datos a un formato legible por humanos que respalde la toma de decisiones informada. Esta técnica logra una mejora en la seguridad de una organización y ayuda a los profesionales a su cargo a ser más rápidos y proactivos.

En este sentido, se ha observado que la mayoría de los ejemplos de toma de decisiones inteligente delegan la adaptación en técnicas de aprendizaje automático (como ocurre en [38, 40, 49, 55, 66, 67, 72, 73]). Otros, sin embargo, lo hacen en mecanismos de aprendizaje profundo (ejemplos de ello

son [41, 65]). También existen soluciones que usan Redes Neuronales ([53]) u otras técnicas relacionadas con la inteligencia artificial ([68]). La forma concreta del uso de estos recursos queda fuera del alcance de esta tesis, por lo que no ha sido analizada en detalle.

Tras la decisión y como ocurre en cualquiera de las categorías de Seguridad Adaptativa, la adaptación permite incrementar o disminuir la seguridad del activo o conjunto de activos que se están protegiendo. El catálogo de adaptaciones posibles se puede encontrar en la sección «2.4. SA atendiendo a la adaptación», donde se estudian en profundidad. Sin embargo, es importante mencionar la diversidad de acciones que las soluciones de esta categoría desempeñan. Por ejemplo, [67] y [68], llevan a cabo una autoadaptación del modelo a los cambios en el entorno; en [49] y [55] se decide si se accede o no al recurso; [53] es capaz de cerrar o mantener la sesión iniciada; [40], [65] y [73] envían alertas de seguridad; y, por su parte, [38], [66] y [72] muestran información de anomalías, intrusiones y ataques.

Con respecto al ámbito de aplicación, las investigaciones previas están enfocadas, casi en su totalidad, a detectar incidentes de seguridad ([38, 40, 41, 65–68, 72, 73]). Aunque, en oposición a esto, [49], [53] y [55] son capaces de aplicar mecanismos inteligentes para mejorar la gestión de identidades y accesos.

Puede encontrarse más información de los ejemplos aquí mencionados con respecto a los detalles y sus características en «2.6. Comparativa de trabajos previos».

2.3.3. Seguridad Adaptativa consciente

Por último, la tercera categoría de la catalogación SA atendiendo al mecanismo de decisión, es la Seguridad Adaptativa consciente. En esta se encuentran los estudios y propuestas en una posición intermedia entre la Seguridad Adaptativa predefinida y la Seguridad Adaptativa inteligente, recayendo parte de la decisión en algoritmos o tecnologías inteligentes (como, por ejemplo, cuando se emplea lógica difusa), pero también siendo necesaria la intervención previa o la preconfiguración por parte de un operador humano.

Al combinarse ambas propuestas para la toma de decisiones (la inteligente y la predefinida), tras la recepción de las métricas y mediciones necesarias, estos datos, normalmente, se analizan o modifican empleando técnicas como las descritas en la sección ≪2.3.2. Seguridad Adaptativa inteligente≫ (aprendizaje automático, aprendizaje profundo, reconocimiento de patrones, redes Bayesianas, etc.). Después, el resultado de este análisis se compara con los valores predefinidos en reglas, políticas, bases de datos o configuraciones redactadas por los operadores en las que se habrá indicado lo que debe hacerse en cada caso (cuál es la adaptación que debe aplicar).

Aunque los trabajos de esta categoría son escasos, la combinación de opciones para la toma de decisiones no lo es. Por ejemplo, [63] y [69] emplean aprendizaje automático, políticas y reglas para determinar la adaptación; [100] se apoya en lógica difusa y en reglas; [25] utiliza clasificación Bayesiana mediante aprendizaje automático y niveles de riesgo predefinidos; [47] decide a partir de los niveles de confiabilidad aportados por el operador y el cálculo probabilístico mediante comparaciones por pares y tasa de error; [112], por su parte, se ayuda de redes Bayesianas multicapa y una matriz de incidencia predefinida; [92] mezcla las políticas con la lógica difusa, el

riesgo de las diferentes acciones permitidas y la criticidad de los datos a los que puede accederse para tomar la decisión; [93] analiza el riesgo mediante redes Bayesianas, tomándose la decisión a partir de información predefinida para el control de acceso; [94] emplea lógica difusa para el cálculo del riesgo que, combinada con políticas e información sobre la criticidad de los datos a los que se puede acceder y el riesgo de las acciones permitidas, es capaz de tomar una decisión; y [74] se encarga de definir qué hacer a partir de reglas y agrupamiento difuso.

Una vez se ha determinado la adaptación, se aplica la acción que corresponde en cada una de las propuestas. Aunque las diferentes posibilidades se detallan y catalogan en la sección «2.4. SA atendiendo a la adaptación» y no guardan relación alguna con el desencadenante («2.2. SA atendiendo al desencadenante de la adaptación») o el mecanismo de decisión («2.3. SA atendiendo al mecanismo de decisión»), se ha considerado relevante destacar las utilizadas por estas herramientas. Por ejemplo, [25] y [47] eligen el mecanismo de autenticación; en [92], [93] y [94] se decide si se puede acceder o no al recurso; [100] determina el acceso al dispositivo; [63] realiza cambios en la configuración o despliega más recursos; [69] envía alertas y reportes de seguridad; y [74] y [112], por su parte, reconfiguran la red y el propio dispositivo respectivamente.

En este caso, y debido a las escasas investigaciones que encajan en esta categoría, el ámbito de aplicación se reduce notablemente. Únicamente se observa esta propuesta enfocada a la gestión de identidades y accesos ([25, 47, 92–94, 100]) y a la detección de incidentes como anomalías o intrusiones ([63, 69, 74, 112]).

Puede encontrarse un análisis en profundidad, con más información sobre los detalles y características de los diferentes ejemplos mencionados en esta sección, en «2.6 Comparativa de trabajos previos».

2.4. SA atendiendo a la adaptación

En esta última catalogación se considera lo que se realiza una vez que se ha determinado la necesidad de adaptación. En otras palabras, se organizan los diferentes trabajos teniendo en cuenta o atendiendo a la adaptación en sí, el cambio en el mecanismo de seguridad o el activo (o conjunto de activos) que se están protegiendo.

Tras el análisis de los diferentes trabajos relacionados con la Seguridad Adaptativa, se llega a la conclusión de que esta categorización es muy diferente a las restantes, ya que no es posible concretar tanto y hacer categorías tan amplias como para que la mayoría de los trabajos recaigan en ellas. Una nueva propuesta podría hacer necesaria la creación de otra categoría; las adaptaciones posibles son casi infinitas. A pesar de ello, se ha tratado de reducir el número de opciones atendiendo a la adaptación que realizan los mecanismos de Seguridad Adaptativa, cogiendo sus aspectos más universales y dándoles nombres genéricos.

Así, estas categorías son: Respuesta a incidentes, cuando la adaptación está relacionada con informar, alertar o responder ante un incidente de seguridad; Configuración, si la modificación consiste en cambiar ciertos parámetros o valores en la configuración de los mecanismos de seguridad o los activos para que operen de una manera diferente; Arquitectura, en los casos en los que se adapta la arquitectura, ampliando o disminuyendo recursos, por ejemplo; Gestión del tráfico de red, cuando la adaptación está asociada a la toma de ciertas decisiones sobre el flujo de los datos a través de la red; Control de accesos, con aquellos trabajos que determinan qué hacer de cara a la identificación y la autorización; y Criptografía, donde se engloban las propuestas que emplean diferentes mecanismos criptográficos o decisiones

sobre estos para adaptar.

2.4.1. Respuesta a incidentes

Centrando la atención en la adaptación, la primera categoría que puede encontrarse es la Seguridad Adaptativa para la respuesta a incidentes. Aquí se incluyen todas las propuestas capaces de respaldar la adaptación o la necesidad de aplicar una acción que requiera del envío de alertas o de informar a los usuarios de incidentes como amenazas, riesgos, *malware*, etc. detectados o pronosticados. Esta necesidad de adecuación se habrá determinado basándose en el desencadenante de la adaptación (véase la sección «2.2. SA atendiendo al desencadenante de la adaptación») y el mecanismo de decisión (estudiado en «2.3. SA atendiendo al mecanismo de decisión»), pudiendo ser ambos de cualquiera de las opciones descritas en la catalogación correspondiente, sin ningún tipo de restricción.

Si bien la categoría «Respuesta a incidentes» no es la más extensa, es necesario hacer hincapié en ella por ser diferente al resto. A pesar de que algunos de los trabajos aquí incluidos no atienden a la definición exacta de Seguridad Adaptativa, se ha considerado necesario incluir en este análisis aquellas investigaciones que, aunque no realizan ninguna adaptación propiamente dicha, se acercan a este modelo. Puede decirse que son solo una parte de los mecanismos de Seguridad Adaptativa; los dos primeros pasos (la recepción de métricas y mediciones, así como el análisis de estas) son idénticos a lo que se busca en la SA, pero las soluciones de este tipo, sin embargo, no llegan a adaptar nada o simplemente envían una alerta a usuarios o administradores. Por tanto, tras recibir las mediciones pertinentes —y, en caso de ser necesario, calcular el riesgo que se está corriendo—, se analizan estos valores y se toma la decisión de actuar como corresponda (basándose en especificaciones

predefinidas o con la ayuda de mecanismos con ciertas capacidades especiales o inteligencia propia), igual que en cualquier solución de SA, aunque en este caso estando la acción que debe realizarse alineada con la respuesta a la detección de incidentes.

Dentro de esta categoría únicamente se encuentran soluciones del ámbito de aplicación de los mecanismos de detección de incidentes. Un ejemplo de ello son aquellas propuestas que avisan a los usuarios o administradores de que algo inusual está ocurriendo en su sistema o su entorno mediante el envío de alertas ([40, 41, 69, 73]). Otras soluciones muestran información relevante de incidentes —como amenazas, vulnerabilidades, riesgos, ataques, etc.— mediante paneles o reportes ([38, 40, 42, 59, 62, 65, 66, 69, 72]). Además, también existen herramientas de este tipo capaces de generar planes de respuesta a incidentes detectados ([108]). Es posible encontrar más detalle de estos trabajos en la sección «2.6. Comparativa de trabajos previos».

2.4.2. Configuración

El siguiente enfoque destacable de esta categorización es la Seguridad Adaptativa para la adaptación de la configuración. Aquí, las soluciones propuestas tratan de realizar diferentes configuraciones en respuesta a la modificación que el mecanismo de decisión ha determinado (pueden observarse las categorías en «2.3. SA atendiendo al mecanismo de decisión») y con base en el desencadenante que corresponda (véase la catalogación «2.2. SA atendiendo al desencadenante de la adaptación»); estos —desencadenantes y mecanismos de decisión—, pueden ser de cualquier tipo, no estando ligados al elemento de adaptación.

En la categoría aquí descrita se engloban aquellas propuestas que modifican

el propio mecanismo o los ficheros de configuración de este, sin cambiar la arquitectura. La adaptación se obtiene alterando la configuración del activo —o activos— que deben protegerse o del control de seguridad, ajustando diferentes parámetros o especificando valores distintos para los componentes que ya se encuentran desplegados; también se tienen en cuenta los trabajos que utilizan el control de seguridad de otra forma, cambiando un protocolo o sus propias políticas, reglas o procedimientos, por poner solo algunos ejemplos. De esta forma, el control de seguridad, el activo o activos, funcionarán de un modo diferente a como lo hacían antes de la adaptación.

Un claro ejemplo de SA para la adaptación de la configuración son aquellas herramientas capaces de reajustar sus modelos de IA, ML, etc. para que funcionen con otros valores o se adapten a nuevos entornos ([38, 40, 66–68]). Otras soluciones se encargan de modificar sus propias políticas de seguridad o sus reglas ([69, 70, 75, 76, 80]). También existen investigaciones comprendidas en esta categoría que desactivan servicios o funciones de los activos ([56, 112]). Los ejemplos expuestos se describen con mayor detalle en la sección «2.6. Comparativa de trabajos previos».

2.4.3. Arquitectura

En la SA para la adaptación de la arquitectura, tal y como ocurre en el resto de las categorías de SA atendiendo a la adaptación, se tiene en cuenta la modificación o la acción que se realiza gracias al desencadenante (véase esta categorización en «2.2. SA atendiendo al desencadenante de la adaptación») y el posterior análisis de esta información con el mecanismo de decisión que corresponda (estudiado en «2.3. SA atendiendo al mecanismo de decisión»), pudiendo ser estos de cualquiera de los tipos presentados en el resto de las clasificaciones.

La categoría aquí descrita engloba todos los trabajos que, en su adaptación, ejecutan cualquier cambio en la arquitectura del propio mecanismo de seguridad, del entorno o del activo o activos que deben protegerse. La modificación se basa en alteraciones estructurales como la eliminación o desactivación de componentes, servicios o herramientas. También tienen cabida las soluciones que despliegan nuevos elementos y aquellas que interactúan de forma diferente con los ya existentes.

Un ejemplo de modificación de arquitectura puede ser la reconfiguración de la estructura de la red, como ocurre en [24] o en [64]. También se tienen en cuenta las soluciones que eligen si se debe desplegar una VPN (*Virtual Private Network*) para transmitir cierta información a través de ella [39]. Otra posibilidad es la de adaptar, optimizar o ampliar recursos de la propia red, como en [63]; o la de apagar o encender contenedores software [61]. Los trabajos aquí enumerados se detallan en la sección «2.6. Comparativa de trabajos previos».

2.4.4. Gestión del tráfico de red

Para esta categoría de Seguridad Adaptativa, denominada SA de gestión del tráfico de red, una vez se ha determinado que es necesaria la adaptación y habiéndose tenido en cuenta para ello tanto el desencadenante («2.2. SA atendiendo al desencadenante de la adaptación») como el mecanismo de decisión («2.3. SA atendiendo al mecanismo de decisión»), se actúa eligiendo cuándo o cómo se debe intercambiar la información a través de la red. Dentro de esta clase de SA, igual que ocurre en las anteriores, desencadenante y mecanismo de decisión pueden ser de cualquiera de las categorías estudiadas, no existiendo ninguna restricción al respecto.

En este sentido, aquí se tienen en cuenta aquellas investigaciones que, para su adaptación, toman una decisión importante acerca de la gestión del tráfico de red, mejorando así la seguridad de un activo o un conjunto de activos. Por ejemplo, sobre la ruta que debe utilizar la información para su transcurso a través de la red. También engloba los trabajos que deciden si estos datos se transmiten, o, por el contrario, no deben hacerlo; o aquellos que especifican si se debe partir o enviar solo un fragmento de esta información.

Ejemplos de investigaciones que se ajustan a esta clase de SA son [77] o [79], cuya adaptación consiste en elegir qué hacer con el tráfico (permitirlo, denegarlo, enrutarlo, etc.). Se pueden considerar también soluciones que seleccionan la ruta óptima para el flujo de datos, enrutándolos por el mejor camino disponible, como en [39], en [78] y en [109]. En [82] la adaptación consiste en seleccionar el nodo *Cloud* que se utiliza en cada momento. Por su parte, [107], además de enrutar los datos, los filtra; y [50] fragmenta la información antes de distribuirla. Puede encontrarse un análisis más detallado acerca de estos trabajos en la sección «2.6. Comparativa de trabajos previos».

2.4.5. Control de accesos

Dentro de la categoría de SA para adaptar el control de accesos, se tienen en cuenta aquellas soluciones que actúan para determinar lo que hacer con los usuarios, activos o servicios, de tal forma que controlan quién o cómo puede acceder a qué. En este sentido, la acción que debe realizarse es independiente del mecanismo de decisión («2.3. SA atendiendo al mecanismo de decisión») y del desencadenante de la adaptación («2.2. SA atendiendo al desencadenante de la adaptación»), los cuales habrán determinado previamente si se debe ejecutar la modificación o no.

Dentro de esta clasificación se tienen en cuenta todas las soluciones que, de una forma u otra, son capaces de gestionar y tomar una decisión con respecto a la identidad o a los accesos de los usuarios o activos que tratan de recuperar información o de acceder a determinados recursos o dispositivos; pero también aquellas que se encargan de otorgar unos privilegios u otros atendiendo a las decisiones previamente tomadas; o las que proponen diferentes mecanismos de autenticación de acuerdo con la información recopilada o el riesgo que se está corriendo en cada momento.

Para ejemplificar esta categoría, algunas soluciones sugieren la decisión de acceso al dispositivo ([44, 52, 88, 100, 101, 104, 107]) o al recurso ([48, 49, 54, 55, 57–60, 90, 92–95, 97–99, 102, 105, 106, 111]). Otros trabajos se centran en ofrecer diferentes mecanismos de autenticación de acuerdo con la decisión tomada ([25, 47, 51, 59, 89, 111]). La elección de privilegios y permisos también es una adaptación muy extendida dentro de esta categoría ([37, 48, 50, 57, 103]). Además, existen trabajos cuya modificación consiste, únicamente, en mantener o no la sesión de un usuario iniciada ([53]); en decidir si se debe realizar o no cierta acción ([56, 96]); o en modificar las políticas de control de acceso ([45]). Puede encontrarse un mayor detalle de los trabajos mencionados en la sección «2.6. Comparativa de trabajos previos».

2.4.6. Criptografía

La última categoría propuesta dentro de la catalogación SA atendiendo a la adaptación es la SA para adaptar la criptografía. Tras la recepción del desencadenante de la adaptación (véase la sección «2.2. SA atendiendo al desencadenante de la adaptación») y el posterior análisis gracias al mecanismo de decisión (expuesto en «2.3. SA atendiendo al mecanismo de decisión»),

las soluciones de SA de esta clase llevan a cabo ciertas acciones relacionadas con la criptografía.

Para esta categoría se consideran aquellas investigaciones que, empleando diferentes técnicas y procedimientos criptográficos, tratan de proteger la información y las comunicaciones como respuesta a la decisión de adaptación, fortaleciendo, de esta forma, la seguridad de los activos que están protegiendo —ya sean estos datos o elementos de la infraestructura— y manteniéndolos fuera de la vista de aquellos individuos, servicios o activos a los que no está dirigida la información. De este modo, se elige si es necesario el cifrado o no, con qué mecanismo, si se requiere la renovación de las claves de cifrado, etc. En algunos de los casos mencionados, es posible utilizar el concepto de cripto-agilidad [113] adaptativa para referirse a la capacidad de los sistemas criptográficos de adaptarse y actualizar sus mecanismos de cifrado ante la aparición de cambios, o incluso en previsión de estos.

En este sentido, el primer ejemplo que puede encontrarse es el de elección del mecanismo de cifrado, como se presenta en [50] o en [51]. Otras propuestas emplean la criptografía para cifrar los mensajes o firmarlos, siendo un ejemplo de ello [106]. Por su parte, en [43] se aplica la renovación de la clave de cifrado cuando cambian ciertos valores del entorno. También se observan mecanismos cuya adaptación consiste en el cifrado de las comunicaciones, como en [91]. Estas investigaciones se muestran con mayor detalle en la sección «2.6. Comparativa de trabajos previos».

2.5. Lógicas de adaptación

A lo largo de los años y en diferentes ámbitos —no solo en el de la informática, o más concretamente el de la ciberseguridad—, se ha comprobado que

algunos procesos requieren la observación del entorno para ser más eficientes y adaptarse a él. Por ello y para hacer frente a los heterogéneos desafíos que se plantean, se proponen las técnicas o lógicas de adaptación (también llamados ciclos de retroalimentación o de mejora). Algunos ejemplos de estas propuestas son [114], [115], [116] o [117].

Aunque existen variaciones de estas lógicas de adaptación [118], de forma general, todas guardan una estructura similar, estando compuestas por cuatro fases bien diferenciadas. La primera de ellas se encarga de recopilar información contextual de diferentes fuentes (por ejemplo, con la observación del medio, utilizando sensores, gracias a mediciones concretas, etc.); la segunda, es la encargada de analizar los datos recopilados en la anterior fase para poder determinar cómo ha cambiado la situación (si lo ha hecho); el siguiente paso es decidir si debe llevarse a cabo alguna modificación o si es necesario actuar en consecuencia a lo observado; por último, si fuera necesario, la lógica de adaptación se encarga de actuar de acuerdo con la decisión tomada [119, 120]. Cabe destacar que las fases descritas pueden ser parte de un único sistema, siendo así la lógica de adaptación centralizada o, por el contrario, ser una lógica de adaptación descentralizada, formada por diferentes elementos aislados pero coordinados entre sí para realizar las acciones pertinentes cuando estas sean necesarias [121].

Los diferentes mecanismos de Seguridad Adaptativa, normalmente, siguen estas lógicas de adaptación, siendo utilizadas como base de las soluciones. Sin embargo, se puede observar en los trabajos analizados en secciones anteriores que, en muchos casos, estas propuestas no se apoyan en ninguno de los modelos existentes para implementarlas o, al menos, no lo hacen siendo conscientes de ello. Los estándares o patrones para aplicar lógicas de adaptación son muy variados y, aunque no están estrechamente relacionados con la ciberseguridad en particular y con el mundo de la informática en

general y se crearon para cubrir otras necesidades muy diferentes, en algunos casos se han adaptado o modificado consiguiendo satisfacer nuevos requisitos. En este sentido, las lógicas de adaptación más extendidas en el ámbito de la informática y la ciberseguridad son PDCA (*Plan-Do-Check-Act*), como en [122]; OODA (*Observe-Orient-Decide-Act*), siendo [123] un ejemplo de ello; y MAPE-K (*Monitor-Analyze-Plan-Execute over a shared Knowledge*), tal y como se muestra en [124].

2.5.1. *Plan-Do-Check-Act*

El ciclo *Plan-Do-Check-Act* (PDCA) o Ciclo de Deming [125, 126] es un método iterativo para el control y la mejora continua tanto de procesos como de productos. Esta técnica se diseñó con la idea de que las empresas u organizaciones pudieran mejorar y ser más competitivas (hacer procesos óptimos y con una mayor calidad, reducir costes y precios, etc.).

De forma general, se puede afirmar que PDCA es un ciclo de mejora continua que, mediante iteraciones, permite la optimización de un objetivo. Consiste en evaluar y aplicar diferentes acciones para corregir errores y solucionar variaciones o problemas; tras ello, el proceso vuelve a iniciarse, ya que es cíclico.

Las fases que forman PDCA son *Plan* o Planificar, *Do* o Hacer, *Check* o Verificar (que con el paso del tiempo cambió el nombre a *Study*) y *Act* o Actuar (véase la figura 2.2) [127]:

- En *Plan*, la primera fase, es donde se lleva a cabo la planificación. Se fijan los objetivos o resultados deseados y se eligen los procesos y medidas a aplicar para llegar a ellos.

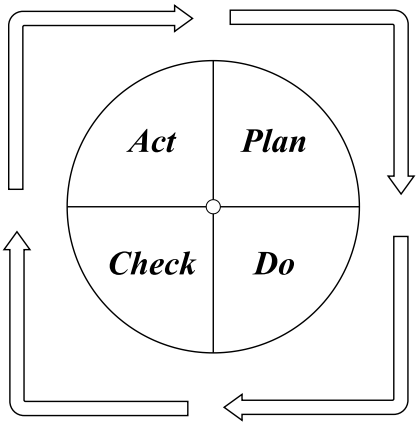

Figura 2.2: Ciclo *Plan-Do-Check-Act* o ciclo de Deming.

- *Do*, en segundo lugar, es la encargada de aplicar y ejecutar todas esas medidas y procesos planificados en el elemento anterior.
- En *Check/Study* se comprueban tanto los resultados como los procesos y medidas aplicadas, monitorizándolos y verificando que los objetivos fijados en *Plan* han sido alcanzados.
- *Act*, la cuarta fase, utiliza el resultado de *Check/Study* y *Do* para revisar si es necesario aplicar cambios en alguno de los procesos del ciclo (por existir algún error, no haberse alcanzado los objetivos, etc.).

A pesar de que en sus orígenes esta técnica tenía una plena orientación a la mejora de la productividad, los costes, la competitividad, etc.; con el paso del tiempo se le ha dado otra perspectiva y se ha aplicado a ámbitos muy diferentes, siendo la seguridad informática uno de ellos. Por poner solo algunos ejemplos, en [128] puede encontrarse un enfoque que permite evaluar la seguridad en entornos dinámicos y cambiantes de IoT; para ello, se emplean tecnologías como *Elasticsearch Stack Solution* [129], basándose en el ciclo PDCA. En la línea del anterior, [130] se centra —empleando PDCA— en proporcionar un plan estratégico para la administración y prevención de riesgos en entornos *Cloud*, teniendo en cuenta diversos estándares y normas. Otro ejemplo del uso de esta técnica, en este caso más acorde con la temática

de esta tesis, es [78], donde se presenta una solución de Seguridad Adaptativa para la gestión, de forma dinámica, de redes SDN/NFV y sus recursos, atendiendo a las anomalías detectadas en el entorno.

2.5.2. *Observe-Orient-Decide-Act*

El ciclo *Observe-Orient-Decide-Act* (OODA) o ciclo de Boyd [131] es un modelo reiterativo de toma de decisiones formado por procesos que interactúan con el entorno. En sus inicios, tenía un contexto marcial y describía una técnica para adquirir, procesar y actuar sobre la información del enemigo; ha sido utilizado por diferentes instituciones militares y ha influido en distintas doctrinas bélicas.

Con este ciclo, se consigue obtener un nuevo escenario de partida en cada iteración, gracias a la observación de la situación, la posterior orientación con los datos obtenidos en ese reconocimiento, la toma de decisiones y la actuación en consecuencia.

Las cuatro fases que forman este modelo son *Observe* u Observación, *Orient* u Orientar, *Decide* o Decidir y *Act* o Actuar (véase la figura 2.3) [127]:

- *Observe* es el proceso de identificar el problema o la amenaza y comprenderlo, así como el entorno en el que se está produciendo, las circunstancias que se están dando en ese momento, etc.
- En *Orient*, la segunda fase, tiene cabida la reflexión sobre lo que se ha observado, así como algunas consideraciones de lo que se podría hacer o no en las siguientes etapas.
- *Decide*, en tercer lugar, es la fase en la que se elige la que se considera mejor solución o actuación, basándose tanto en los datos observados

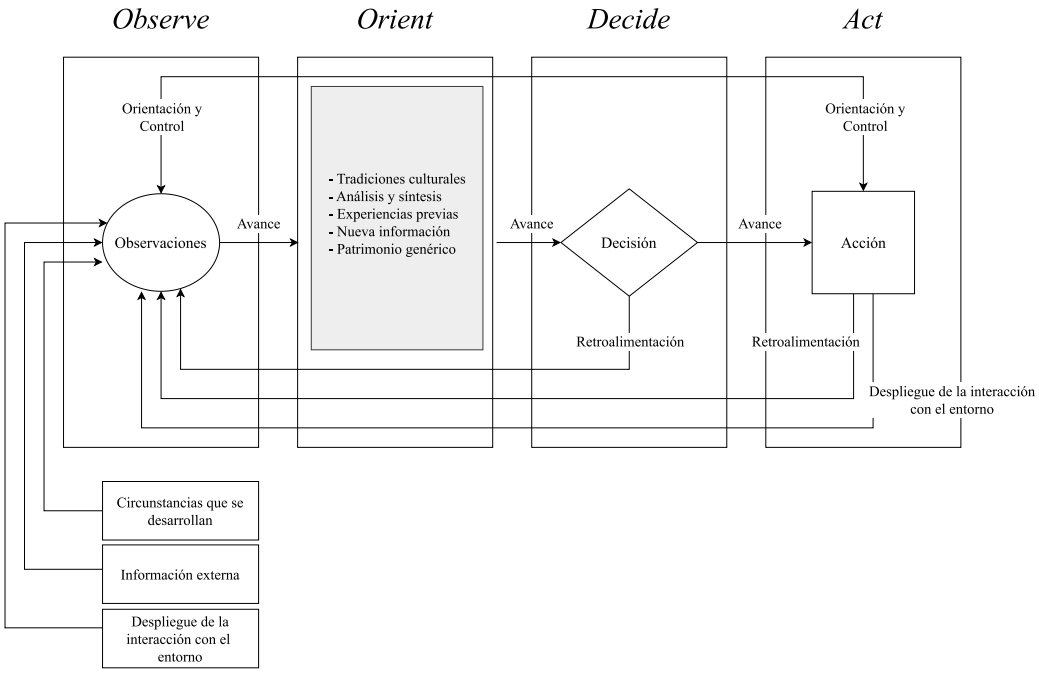

Figura 2.3: Ciclo *Observe-Orient-Decide-Act* o ciclo de Boyd.

como en el análisis realizado en la etapa de orientación. La decisión tomada sirve de retroalimentación para *Observe* en futuras iteraciones del ciclo.

- Por último, en *Act*, donde se aplica la decisión, se lleva a cabo la acción pertinente para ello. Esta acción o actuación alimenta también a la fase *Observe* de cara a futuras iteraciones.

A pesar de su idea inicial, esta técnica ha evolucionado y se ha adaptado para dar cabida a infinidad de soluciones fuera del ámbito militar. En el dominio de la ciberseguridad existen estudios como [132], donde se expone un método para la detección temprana de APTs (*Advanced Persistent Threats* o Amenazas Persistentes Avanzadas). Otro ejemplo es [133], que aprovecha OODA para la evaluación de riesgos y la respuesta a incidentes de ciberseguridad en la industria sanitaria. En la línea del tema tratado en esta tesis, encontramos [79], que propone un mecanismo —con OODA como base— para la adaptación de redes corporativas (por ejemplo, modificando la lista de control de acceso

de un *firewall*), tratando de minimizar el impacto o paliar ciertas amenazas (como cuando se produce un ataque de denegación de servicio —DDoS o *Distributed Denial of Service*).

2.5.3. *Monitor-Analyze-Plan-Execute over a shared Knowledge*

El ciclo *Monitor-Analyze-Plan-Execute over a shared Knowledge* (MAPE-K) [134], es un modelo de control con retroalimentación que fue creado para llevar a cabo autoadaptación en diferentes sistemas, siendo este el más genérico de los analizados.

Con su propuesta, MAPE-K es capaz de monitorizar tanto el contexto operativo como el entorno para, tras el análisis de la información recopilada, planificar los cambios o modificaciones que se van a realizar y, por último, aplicar o ejecutar dichas adaptaciones. Al igual que los anteriores, se trata de un modelo reiterativo, por lo que estas acciones se repiten tras finalizar.

Este ciclo se compone de una secuencia de cuatro cálculos: *Monitor* o Monitorizar (≪M≫), *Analyse* o Analizar (≪A≫), *Plan* o Planificar (≪P≫) y *Execute* o Ejecutar (≪E≫). Todos los componentes que forman MAPE-K pueden estar descentralizados, pero siempre coordinados entre sí. La figura 2.4 muestra la estructura de este ciclo:

- ≪M≫ o *Monitor* se encarga de la monitorización del contexto operativo y del entorno, recopilando los datos e información necesaria y utilizando para ello, normalmente, sensores o sondas.
- ≪A≫ o *Analyse* recibe y analiza los datos recopilados por ≪M≫ en busca de cambios significativos o para detectar determinados patrones.

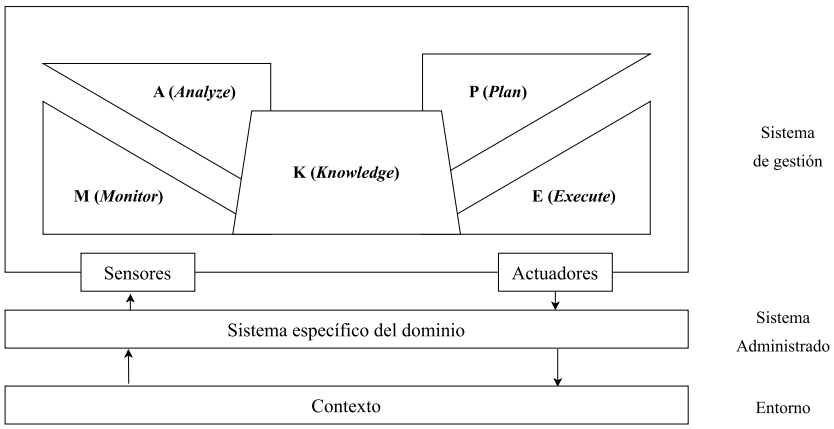

Figura 2.4: Ciclo *Monitor-Analyze-Plan-Execute over a shared Knowledge* (Reproducido de [1]).

- ≪P≫ o *Plan*, después, planifica la adaptación que se debe realizar para lograr los objetivos planteados inicialmente. Para este cálculo, suelen emplearse reglas y políticas, de forma que las decisiones se tomarán acorde con el análisis realizado por ≪A≫ y con el resultado de evaluar las políticas en ≪P≫, aplicándose las adaptaciones que corresponda mediante ≪E≫.

- ≪E≫ o *Execute*, en cuarto lugar, realiza las adaptaciones indicadas por ≪P≫, controlando que la ejecución de estas se haga de forma correcta.

- Por último, también existe un repositorio de conocimiento (≪K≫), que se comparte entre componentes y sirve como base de datos para almacenar información del sistema que se está administrando, del contexto, las metas de adaptación, etc.

Este ciclo, que fue concebido con la idea de aplicarse en sistemas autogestionados, es el más influyente y utilizado en informática. Un claro ejemplo de esta distribución es [135], donde se utiliza MAPE-K con una arquitectura orientada a servicios en un *middleware* reflexivo basado en computación autónoma. Por otro lado, [136] aplica MAPE-K a la robótica de servicios para, entre otras, mejorar la seguridad. Otros ejemplos de este tipo pueden

encontrarse en [137], un estudio con diferentes enfoques de ciberseguridad en el que se detallan también aquellos que aprovechan MAPE-K. Con relación a lo dispuesto en esta tesis, destaca [110], un *framework* apoyado en MAPE-K para la adaptación de diferentes controles de seguridad basándose en el riesgo calculado; [68], un sistema de detección de intrusiones que lleva a cabo el auto-reentreno del modelo de DL —con MAPE-K como base—; [76] y [80], que presentan sendos mecanismos de Seguridad Adaptativa para reconfigurar las reglas de un *firewall* y un WAF (*Web Application Firewall*) respectivamente, acorde con las mediciones del entorno en el que operan y con MAPE-K como lógica de adaptación; y [56], que propone una herramienta para aplicar cambios en la forma en la que se gestionan las identidades en entornos *Cloud*, con las amenazas detectadas como desencadenante y MAPE-K como soporte.

Además de estos ciclos de mejora (MAPE-K, OODA y PDCA), existen otros muchos que, por su naturaleza, han tenido una menor aplicación en el ámbito informático, en el de la autoadaptación o en el de la ciberseguridad. Entre ellos, cabe mencionar algunos como DFSS (*Design For Six Sigma*) [138] y DMAIC (*Define-Measure-Analyze-Improve-Control*) [139].

A pesar de la patente utilidad de estas lógicas de adaptación, en ciberseguridad, y más concretamente en Seguridad Adaptativa, han tenido aún muy poca acogida. La mayor parte de los trabajos analizados (véase la sección «2.6. Comparativa de trabajos previos»), atienden a lógicas de adaptación propias o utilizan alguna de las existentes —o muy similares a ellas—, pero sin ser conscientes de su uso o sin mencionarlo en sus investigaciones y propuestas. En otras disciplinas más clásicas y maduras, como por ejemplo en sistemas distribuidos ([140, 141]) o sistemas autónomos ([142, 143]), estas lógicas de adaptación son ampliamente utilizadas y aplicadas en una gran cantidad de proyectos.

2.6. Comparativa de trabajos previos

A pesar de que los trabajos, investigaciones y soluciones aquí expuestas se han mencionado y catalogado en secciones anteriores, se ha considerado necesario un análisis más exhaustivo de cada uno de ellos, mejorándose así la comprensión, lectura y comparación de sus funcionalidades, así como el motivo de las clasificaciones. Este estudio también es útil para favorecer la posterior reflexión de sus fortalezas y debilidades y llegar a la conclusión de las carencias que es necesario abordar en el ámbito de la Seguridad Adaptativa.

Los resultados de la investigación se exponen en la tabla 2.1, ordenada y diferenciada por dominios de aplicación: IoT, *Cloud Computing* o *Cloud*, Móvil, redes corporativas, SDN, Aplicaciones web, Sistemas de Control Industrial (ICS o *Industrial Control Systems*), BYOD (*Bring Your Own Device*), 5G, contenedores software, vehículos autónomos, *Fog Computing* o *Fog*, *Edge Computing* o *Edge*, sistemas distribuidos autónomos, sistemas de detección de intrusos (IDSs o *Intrusion Detection Systems*), *middleware* de mensajería autónomos y sistemas software.

En la tabla pueden encontrarse todos los detalles de cada trabajo comparado, incluyendo: (1) el trabajo en cuestión —en la columna «Ref.»—; (2) su dominio de aplicación —columna «Dominio»—; (3) su categoría de SA atendiendo al desencadenante de la adaptación —en la columna «C. Desenc.»—; (4) su clase de SA atendiendo al mecanismo de decisión —columna «C. Decisión»—; (5) el tipo de SA atendiendo a la adaptación —columna «C. Adapt.»—; (6) su ámbito de aplicación —columna «Aplicación»—; (7) las métricas o mediciones utilizadas como desencadenante —columna «Métricas y mediciones»—; (8) la técnica a partir de la cual toma la decisión de adaptación —columna «T. Decisión»—; y (9) la lógica de adaptación utilizada

—columna «L. Adaptación»—.

Como puede observarse gracias a este análisis en profundidad, las soluciones de Seguridad Adaptativa existentes cubren un amplio espectro de dominios de aplicación, siendo el más extendido el de IoT; sin embargo, no se han hallado soluciones genéricas o con la capacidad de incluirse en diferentes dominios, permitiendo así a los administradores manejar una única herramienta para cubrir todas las posibles necesidades de una organización. En este sentido, en el ámbito de aplicación también se observa la problemática mencionada, ya que, a pesar de que las soluciones son heterogéneas, estas se centran en resolver problemas muy concretos, siendo necesario desplegar más de una si se quieren solventar varios de ellos. Las categorías de adaptación también son variadas y poco generalistas, de nuevo sin existir ninguna propuesta capaz de aplicar adaptaciones para todas —o muchas— de las necesidades de las organizaciones.

Es importante destacar además la falta de información sobre métricas, mediciones y datos recopilados para actuar de desencadenantes de la adaptación. En la mayoría de las propuestas, estas mediciones no se especifican o lo hacen de forma muy ambigua y con poco detalle, dejando la elección en manos de los administradores o del lector; esto dificulta la comprensión y el despliegue de las soluciones, haciéndolas difíciles y costosas para las organizaciones, tanto en términos económicos como de recursos humanos. También resalta la baja utilización de lógicas de adaptación como PDCA, OODA o MAPE-K o, al menos, el desconocimiento sobre su uso, ya que en ocasiones se ha observado que, sin mencionarse, la estructura utilizada es similar a la de alguno de estos ciclos.

Para finalizar este análisis, debe señalarse que, según la categorización atendiendo al desencadenante de la adaptación, la SA sensible al contexto es, con

diferencia, la más explorada; esto es comprensible, ya que llevar a cabo un cálculo del riesgo coherente, útil y realista, es una tarea ardua y que requiere grandes conocimientos tanto de la organización como de la infraestructura que se quiere proteger. Por otro lado, atendiendo al mecanismo de decisión, las propuestas de las categorías inteligente y consciente son muy similares en cantidad, habiendo una clara diferencia entre estas y la predefinida; esto ocurre debido a que la última es más fácil de implementar y desplegar, sin necesidad de grandes capacidades o recursos computacionales (como sí ocurre con aquellas que emplean, por ejemplo, IA) o conocimiento de técnicas estadísticas (tales como los necesarios para usar redes Bayesianas).

Tabla 2.1: Resumen de trabajos previos en Seguridad Adaptativa.

Ref.	**Dominio**	**C. Desenc.**	**C. Decisión**	**C. Adapt.**	**Aplicación**	**Métricas y mediciones**	**T. Decisión**	**L. Adaptación**
[62]	IoT	Sensible al contexto	Predefinida	Respuesta a incidentes	Detección de incidentes	*Fingerprints* de señales de sensores físicos obtenidos directamente del sensor	Algoritmos propios	No definido formalmente
[59]	IoT	Sensible al contexto	Predefinida	Control de accesos/ Respuesta a incidentes	Gestión de identidades y accesos	No definido formalmente	Políticas y Reglas	No definido formalmente
[52]	IoT (Hogar inteligente)	Sensible al contexto	Predefinida	Control de accesos	Gestión de identidades y accesos	Patrones de uso de los dispositivos, perfiles de usuarios y calendario	Políticas y Reglas	No definido formalmente
[44]	IoT	Sensible al contexto	Predefinida	Control de accesos	Gestión de identidades y accesos/ Cifrado	Ubicación, hora, situación de emergencia, situación normal, tipo de datos, etc.	Políticas y Reglas	No definido formalmente
[57]	IoT	Sensible al contexto	Predefinida	Control de accesos	Gestión de identidades y accesos/ Cifrado	No definido formalmente	Políticas y Reglas	No definido formalmente

Continúa en la siguiente página.

Tabla 2.1 – Continuación de la página anterior.

Ref.	Dominio	C. Desenc.	C. Decisión	C. Adapt.	Aplicación	Métricas y mediciones	T. Decisión	L. Adaptación
[51]	IoT	Sensible al contexto	Predefinida	Control de accesos/ Criptografía	Gestión de identidades y accesos/ Cifrado	No definido formalmente	Políticas y Reglas	No definido formalmente
[48]	IoT (Espacios inteligentes y e-Salud)	Sensible al contexto	Predefinida	Control de accesos	Gestión de identidades y accesos	No definido formalmente	Políticas y Reglas, *Web Ontology Language*	No definido formalmente
[24]	IoT	Sensible al contexto	Predefinida	Arquitectura	Detección de incidentes	No definido formalmente	Políticas y Reglas	No definido formalmente
[39]	IoT (Ciudades inteligentes)	Sensible al contexto	Predefinida	Arquitectura / Gestión del tráfico de red	Definición de la arquitectura/ Configuración de la red	Identificador del dispositivo IoT	BBDD de criticidad de los datos y de las rutas más seguras	No definido formalmente

Continúa en la siguiente página.

Tabla 2.1 – Continuación de la página anterior.

Ref.	Dominio	C. Desenc.	C. Decisión	C. Adapt.	Aplicación	Métricas y mediciones	T. Decisión	L. Adaptación
[43]	IoT (Red eléctrica inteligente)	Sensible al contexto	Predefinida	Criptografía	Gestión de identidades y accesos/ Cifrado	Estimación de estado (potencia activa, potencia reactiva, magnitud y fase del voltaje del bus, etc.)	Cambios en las mediciones de potencia	No definido formalmente
[46]	IoT	Sensible al contexto	Predefinida	No definido formalmente	Gestión de identidades y accesos	No definido formalmente	Políticas y Reglas, BBDD con identificadores de dispositivos	No definido formalmente
[75]	IoT	Sensible al contexto	Predefinida	Configuración	Detección de incidentes	No definido formalmente	Políticas y Reglas	No definido formalmente
[41]	IoT (Red eléctrica inteligente)	Sensible al contexto	Inteligente	Respuesta a incidentes	Detección de incidentes	No definido formalmente	DL	No definido formalmente
[73]	IoT (Hogar inteligente)	Sensible al contexto	Inteligente	Respuesta a incidentes	Detección de incidentes	Estados lógicos de sensores y dispositivos; Funciones, localización y comandos enviados; código de las aplicaciones	ML (Cadena de Márkov)	No definido formalmente

Continúa en la siguiente página.

Tabla 2.1 – Continuación de la página anterior.

Ref.	Dominio	C. Desenc.	C. Decisión	C. Adapt.	Aplicación	Métricas y mediciones	T. Decisión	L. Adaptación
[38]	IoT (Sistemas de Transporte Marítimo)	Sensible al contexto	Inteligente	Respuesta a incidentes/ Configuración	Detección de incidentes	Información del tráfico de red (IP/puerto origen y destino, servicio, protocolo, etc.)	ML	No definido formalmente
[67]	IoT (Hogar inteligente)	Sensible al contexto	Inteligente	Configuración	Detección de incidentes	No definido formalmente	ML	No definido formalmente
[91]	IoT (e-Salud)	Basada en riesgos	Predefinida	Criptografía	No definido formalmente	No definido formalmente	Políticas y Reglas	No definido formalmente
[107]	IoT	Basada en riesgos	Predefinida	Gestión del tráfico de red/ Control de accesos/ etc.	Detección de incidentes	No definido formalmente	Políticas y Reglas, firmas de incidentes	No definido formalmente
[98]	IoT	Basada en riesgos	Predefinida	Control de accesos	Gestión de identidades y accesos	Fecha/hora, localización, red, aplicaciones, antivirus, etc.	Políticas y Reglas, niveles de riesgo de atributos	No definido formalmente

Continúa en la siguiente página.

Tabla 2.1 – Continuación de la página anterior.

Ref.	Dominio	C. Desenc.	C. Decisión	C. Adapt.	Aplicación	Métricas y mediciones	T. Decisión	L. Adaptación
[96]	IoT (Hogar inteligente)	Basada en riesgos	Predefinida	Control de accesos	Gestión de identidades y accesos	No definido formalmente	Matriz de riesgos/acciones permitidas	No definido formalmente
[100]	IoT y *Fog*	Basada en riesgos	Consciente	Control de accesos	Gestión de identidades y accesos	Sensor o dispositivo, cliente, acción a realizar, fecha/hora, tipo de red, información del dispositivo, localización, etc.	Políticas y Reglas, lógica difusa	No definido formalmente
[92]	IoT	Basada en riesgos	Consciente	Control de accesos	Gestión de identidades y accesos	Atributos de usuario (hora, ubicación, sensibilidad de los datos a los que accede, gravedad de las acciones que realizará, historial de riesgos del usuario, etc.)	Políticas y Reglas, lógica difusa, nivel de criticidad y riesgo de datos y acciones	No definido formalmente

Continúa en la siguiente página.

Tabla 2.1 – Continuación de la página anterior.

Ref.	Dominio	C. Desenc.	C. Decisión	C. Adapt.	Aplicación	Métricas y mediciones	T. Decisión	L. Adaptación
[94]	IoT	Basada en riesgos	Consciente	Control de accesos	Gestión de identidades y accesos	Atributos de usuario (hora, ubicación, sensibilidad de los datos a los que accede, gravedad de las acciones que realizará, historial de riesgos del usuario, etc.)	Políticas y Reglas, lógica difusa, nivel de criticidad y riesgo de datos y acciones	No definido formalmente
[25]	IoT (Hogar inteligente y e-Salud)	Basada en riesgos	Consciente	Control de accesos	Gestión de identidades y accesos	Patrones de comportamiento del usuario y características del canal del dispositivo utilizado	Niveles de riesgo, ML (Clasificación Bayesiana)	No definido formalmente
[69]	IoT	Sensible al contexto	Consciente	Respuesta a incidentes/ Configuración	Detección de incidentes	Información de tráfico de red (con Snort [144]), información de tráfico WiFi (con Kismet [145]), información de vulnerabilidades (con OpenVAS [146])	Políticas y Reglas, ML	No definido formalmente

Continúa en la siguiente página.

Tabla 2.1 – Continuación de la página anterior.

Ref.	Dominio	C. Desenc.	C. Decisión	C. Adapt.	Aplicación	Métricas y mediciones	T. Decisión	L. Adaptación
[80]	*Cloud*	Sensible al contexto	Predefinida	Configuración	Definición de la arquitectura/ Configuración de la red	No definido formalmente	Políticas y Reglas	MAPE-K
[82]	*Cloud*	Sensible al contexto	Predefinida	Gestión del tráfico de red	Distribución de tareas	Proximidad del dispositivo, disponibilidad del *Cloud*, calidad del servicio, latencia, etc.	Políticas y Reglas	No definido formalmente
[111]	*Cloud*	Sensible al contexto	Predefinida	Control de accesos	Gestión de identidades y accesos	Dispositivo utilizado, hora, lugar, huella digital del usuario (biometría, cara, voz, firma, iris, etc.), historial de autenticación	Políticas y Reglas	No definido formalmente
[56]	*Cloud* (OpenStack [147])	Sensible al contexto	Predefinida	Configuración/ Control de accesos	Gestión de identidades y accesos	No definido formalmente	Políticas y Reglas, acciones y respuestas asociadas, impactos, etc.	MAPE-K

Continúa en la siguiente página.

Tabla 2.1 – Continuación de la página anterior.

Ref.	Dominio	C. Desenc.	C. Decisión	C. Adapt.	Aplicación	Métricas y mediciones	T. Decisión	L. Adaptación
[60]	*Cloud* (*e-Salud*)	Sensible al contexto	Predefinida	Control de accesos	Gestión de identidades y accesos/ Cifrado	Identidad del usuario, rol, patrones de acceso, tipo de conexión, etc.	Políticas y Reglas	No definido formalmente
[54]	*Cloud* y Móvil	Sensible al contexto	Predefinida	Control de accesos	Gestión de identidades y accesos	Coordenadas GPS, hora/fecha, procesos, tareas, tipo y versión del sistema operativo, tipo de conexión, aplicaciones	BBDD de acciones y comportamientos permitidos	No definido formalmente
[50]	*Cloud* (*Platform as a Service*)	Sensible al contexto	Predefinida	Control de accesos/ Criptografía/ Gestión del tráfico de red	Gestión de identidades y accesos	Localización, fecha/hora, información de conexión, motivos del acceso, secuencia de acciones, etc.	Políticas y Reglas	No definido formalmente
[65]	*Cloud*	Sensible al contexto	Inteligente	Respuesta a incidentes	Detección de incidentes	Información del tráfico de red (IP/puerto origen y destino, duración, servicio, protocolo, etc.)	DL	No definido formalmente

Continúa en la siguiente página.

Tabla 2.1 – Continuación de la página anterior.

Ref.	Dominio	C. Desenc.	C. Decisión	C. Adapt.	Aplicación	Métricas y mediciones	T. Decisión	L. Adaptación
[101]	*Cloud*	Basada en riesgos	Predefinida	Control de accesos	Gestión de identidades y accesos	No definido formalmente	Políticas y Reglas, niveles de riesgo, necesidad operacional	No definido formalmente
[97]	*Cloud*	Basada en riesgos	Predefinida	Control de accesos	Gestión de identidades y accesos	No definido formalmente	Políticas y Reglas, niveles de riesgo	No definido formalmente
[102]	*Cloud*	Basada en riesgos	Predefinida	Control de accesos	Gestión de identidades y accesos	No definido formalmente	Políticas y Reglas, niveles de riesgo	No definido formalmente
[105]	*Cloud*	Basada en riesgos	Predefinida	Control de accesos	Gestión de identidades y accesos	No definido formalmente	Políticas y Reglas, BBDD con niveles de riesgo	No definido formalmente

Continúa en la siguiente página.

Tabla 2.1 – Continuación de la página anterior.

Ref.	Dominio	C. Desenc.	C. Decisión	C. Adapt.	Aplicación	Métricas y mediciones	T. Decisión	L. Adaptación
[95]	*Cloud*	Basada en riesgos	Predefinida	Control de accesos	Gestión de identidades y accesos	Características del solicitante y de los componentes IT, factores situacionales y del entorno, características de la información solicitada, etc.	Políticas y Reglas	No definido formalmente
[58]	Móvil (Android)	Sensible al contexto	Predefinida	Control de accesos	Gestión de identidades y accesos	No definido formalmente	Políticas y Reglas	No definido formalmente
[37]	Móvil (Android)	Sensible al contexto	Predefinida	Control de accesos	Gestión de identidades y accesos	Estado del teléfono y la batería, fecha/hora, tipo de almacenamiento, información de sensores, etc.	Políticas y Reglas	No definido formalmente
[72]	Móvil (Android)	Sensible al contexto	Inteligente	Respuesta a incidentes	Detección de incidentes	Información de sensores y de su uso durante acciones cotidianas	ML (Cadena de Márkov, Clasificador bayesiano y Árbol Modelo Logístico)	No definido formalmente

Continúa en la siguiente página.

Tabla 2.1 – Continuación de la página anterior.

Ref.	Dominio	C. Desenc.	C. Decisión	C. Adapt.	Aplicación	Métricas y mediciones	T. Decisión	L. Adaptación
[53]	Móvil (Android)	Sensible al contexto	Inteligente	Control de accesos	Gestión de identidades y accesos	Mensajes de texto y llamadas entrantes y salientes; historial del navegador, del WiFi y de aplicaciones; estado de la pantalla	Redes neuronales	No definido formalmente
[49]	Móvil	Sensible al contexto	Inteligente	Control de accesos	Gestión de identidades y accesos	Geolocalización, fecha/hora, actividad, identidad, etc.	ML (SVM o *Support vector machine*)	No definido formalmente
[47]	Móvil	Sensible al contexto	Consciente	Control de accesos	Gestión de identidades y accesos	Dispositivo, fecha/hora, medios, condiciones ambientales (luz, ruido, movimiento), histórico, motivo, etc.	niveles de confiabilidad, comparaciones por pares y tasa de error	No definido formalmente

Continúa en la siguiente página.

Tabla 2.1 – Continuación de la página anterior.

Ref.	Dominio	C. Desenc.	C. Decisión	C. Adapt.	Aplicación	Métricas y mediciones	T. Decisión	L. Adaptación
[104]	Móvil	Basada en riesgos	Predefinida	Control de accesos	Gestión de identidades y accesos	Información del usuario (ubicación, fecha/hora, cercanía al *token* de proximidad del usuario, etc.) y del dispositivo (IP, reputación de la red, etc.)	Políticas y Reglas	No definido formalmente
[42]	Redes corporativas	Sensible al contexto	Predefinida	Respuesta a incidentes	Detección de incidentes	No definido formalmente	Algoritmos propios	No definido formalmente
[76]	Redes corporativas	Sensible al contexto	Predefinida	Configuración	Detección de incidentes	CVE —*Common Vulnerabilities and Exposures*—, CVSS —*Common Vulnerability Scoring System*—, información de servidores (sistema operativo, IP, servicios, versiones, puertos, etc.)	Políticas y Reglas, inventario de vulnerabilidades	MAPE-K
[79]	Redes corporativas	Sensible al contexto	Predefinida	Gestión del tráfico de red	Detección de incidentes	No definido formalmente	Políticas y Reglas	OODA

Continúa en la siguiente página.

Tabla 2.1 – Continuación de la página anterior.

Ref.	Dominio	C. Desenc.	C. Decisión	C. Adapt.	Aplicación	Métricas y mediciones	T. Decisión	L. Adaptación
[68]	Redes corporativas	Sensible al contexto	Inteligente	Configuración	Detección de incidentes	Características de la red (topología, *hosts* disponibles, servicios en ejecución, puertos, sistemas operativos, vulnerabilidades potenciales, etc.) y tráfico (IP/puerto origen y destino, servicio, protocolo, etc.)	IA, DL	MAPE-K
[40]	Redes corporativas	Sensible al contexto	Inteligente	Respuesta a incidentes/ Configuración	Detección de incidentes	Información del tráfico de red (IP/puerto origen y destino, protocolo, tamaño y duración de las peticiones, etc. —con *NetFlow* [148]—)	ML	No definido formalmente
[66]	Redes corporativas	Sensible al contexto	Inteligente	Respuesta a incidentes/ Configuración	Detección de incidentes	No definido formalmente	ML	No definido formalmente

Continúa en la siguiente página.

Tabla 2.1 – Continuación de la página anterior.

Ref.	Dominio	C. Desenc.	C. Decisión	C. Adapt.	Aplicación	Métricas y mediciones	T. Decisión	L. Adaptación
[74]	Redes corporativas	Sensible al contexto	Consciente	No definido formalmente	Detección de incidentes	Carga útil media, n.º total de bytes y duración por conexión; protocolo de transmisión; tamaño de la sesión; geolocalización; tiempo de respuesta del servidor; longitud del nombre de dominio, etc.	Políticas y Reglas, agrupamiento difuso	No definido formalmente
[78]	SDN/NFV	Sensible al contexto	Predefinida	Gestión del tráfico de red	Detección de incidentes	No definido formalmente	Políticas y Reglas	PDCA
[109]	SDN	Basada en riesgos	Predefinida	Gestión del tráfico de red	Detección de incidentes	No definido formalmente	Políticas y Reglas, inventario de vulnerabilidades	No definido formalmente

Continúa en la siguiente página.

Tabla 2.1 – Continuación de la página anterior.

Ref.	Dominio	C. Desenc.	C. Decisión	C. Adapt.	Aplicación	Métricas y mediciones	T. Decisión	L. Adaptación
[77]	SDN	Sensible al contexto	Predefinida	Gestión del tráfico de red	Definición de la arquitectura/ Configuración de la red	Atributos de flujo (identificador, secuencia de paquetes, perfil de seguridad, etc.), atributos de dominio (identidades origen y destino, direcciones de subred, puertas de enlace, etc.), atributos de los conmutadores (identificadores, etiquetas de seguridad, etc.), atributos del *host* (IP y MAC origen y destino, etc.), etc.	Políticas y Reglas	No definido formalmente
[70]	Aplicaciones Web (Mensajes SOAP)	Sensible al contexto	Predefinida	Configuración	Detección de incidentes	Contenido de los mensajes SOAP (*Simple Object Access Protocol*)	Políticas y Reglas	No definido formalmente
[90]	Aplicaciones Web	Basada en riesgos	Predefinida	Control de accesos	Gestión de identidades y accesos	*Fingerprint* del navegador y geolocalización	Algoritmos propios	No definido formalmente

Continúa en la siguiente página.

Tabla 2.1 – Continuación de la página anterior.

Ref.	Dominio	C. Desenc.	C. Decisión	C. Adapt.	Aplicación	Métricas y mediciones	T. Decisión	L. Adaptación
[89]	Aplicaciones Web	Basada en riesgos	Predefinida	Control de accesos	Gestión de identidades y accesos	Localización, periodo de tiempo, servicio al que desea acceder, dispositivo, etc.	Niveles de riesgo asociados a los mecanismos de autenticación	No definido formalmente
[103]	Aplicaciones Web	Basada en riesgos	Predefinida	Control de accesos	Gestión de identidades y accesos	No definido formalmente	Políticas y Reglas, niveles de riesgo, valor de los recursos, etc.	No definido formalmente
[64]	ICS	Sensible al contexto	Predefinida	Arquitectura	Detección de incidentes	No definido formalmente	Políticas y Reglas	No definido formalmente

Continúa en la siguiente página.

Tabla 2.1 – Continuación de la página anterior.

Ref.	Dominio	C. Desenc.	C. Decisión	C. Adapt.	Aplicación	Métricas y mediciones	T. Decisión	L. Adaptación
[108]	ICS	Basada en riesgos	Predefinida	Respuesta a incidentes	Detección de incidentes	Identificador y tipo del dispositivo; vulnerabilidades asociadas o CVE; etc.	Políticas y Reglas, inventario de vulnerabilidades, matriz de accesibilidad, acciones de mitigación autorizadas	No definido formalmente
[112]	ICS	Basada en riesgos	Consciente	Configuración	Detección de incidentes	No definido formalmente	Red Bayesiana multicapa, matriz de incidencia	No definido formalmente

Continúa en la siguiente página.

Tabla 2.1 – Continuación de la página anterior.

Ref.	Dominio	C. Desenc.	C. Decisión	C. Adapt.	Aplicación	Métricas y mediciones	T. Decisión	L. Adaptación
[55]	BYOD	Sensible al contexto	Inteligente	Control de accesos	Gestión de identidades y accesos	Identificadores de dispositivos y sus usuarios, de área y de celda; hora y día de la semana; actividad; dirección de la actividad; duración; plan mensual; proveedor del dispositivo; grupo y afiliación del usuario; etc.	ML (Redes neuronales artificiales, árboles de decisión)	No definido formalmente
[93]	BYOD	Basada en riesgos	Consciente	Control de accesos	Gestión de identidades y accesos	No definido formalmente	Redes Bayesianas, granularidad para la configuración del control de acceso	No definido formalmente

Continúa en la siguiente página.

Tabla 2.1 – Continuación de la página anterior.

Ref.	Dominio	C. Desenc.	C. Decisión	C. Adapt.	Aplicación	Métricas y mediciones	T. Decisión	L. Adaptación
[63]	5G	Sensible al contexto	Consciente	Arquitectura	Detección de incidentes	Información de flujos (volumen de datos de entrada y salida, IP/puertos de origen y destino, n.º de flujos entrantes y salientes, etc.)	Políticas y Reglas, ML	No definido formalmente
[61]	Contenedores Software	Sensible al contexto	Predefinida	Arquitectura	Detección de incidentes	No definido formalmente	Políticas y Reglas	No definido formalmente
[45]	Vehículos autónomos	Sensible al contexto	Predefinida	Control de accesos	Gestión de identidades y accesos	No definido formalmente	Políticas y Reglas	No definido formalmente

Continúa en la siguiente página.

Tabla 2.1 – Continuación de la página anterior.

Ref.	Dominio	C. Desenc.	C. Decisión	C. Adapt.	Aplicación	Métricas y mediciones	T. Decisión	L. Adaptación
[88]	*Fog*	Basada en riesgos	Predefinida	Control de accesos	Gestión de identidades y accesos	Reputación del sujeto (histórico de peticiones, acceso al sistema y accesos maliciosos), contexto del sujeto (preferencia de nodo, permisos, etc.), entorno de configuración (hora, dirección, estado de la red, etc.), estado del objeto (tamaño, comportamiento, etc.), etc.	Políticas y Reglas, niveles de riesgo	No definido formalmente
[81]	*Edge* (Ciudades inteligentes)	Sensible al contexto	Predefinida	No definido formalmente	Definición de la arquitectura/ Configuración de la red	No definido formalmente	Políticas y Reglas	No definido formalmente
[110]	Sistemas distribuidos autónomos	Basada en riesgos	Predefinida	No definido formalmente	Definición de la arquitectura/ Configuración de la red	No definido formalmente	Políticas	MAPE-K

Continúa en la siguiente página.

Tabla 2.1 – Continuación de la página anterior.

Ref.	Dominio	C. Desenc.	C. Decisión	C. Adapt.	Aplicación	Métricas y mediciones	T. Decisión	L. Adaptación
[99]	IDS	Basada en riesgos	Predefinida	Control de accesos	Gestión de identidades y accesos	No definido formalmente	Políticas y Reglas	No definido formalmente
[106]	*Middleware* de mensajería	Basada en riesgos	Predefinida	Control de accesos/ Criptografía	Gestión de identidades y accesos/ Cifrado	No definido formalmente	Políticas y Reglas	No definido formalmente
[71]	Sistemas Software	Sensible al contexto	Predefinida	No definido formalmente	Detección de incidentes	Partes del código fuente de la aplicación	Políticas y Reglas, firmas de vulnerabilidades, lista de acciones predefinida	No definido formalmente

Capítulo 3

Un nuevo modelo para la Seguridad Adaptativa basado en el riesgo

Tras el análisis del estado del arte en la materia tratada en esta tesis (véase el capítulo «2. Estado del arte»), se han detectado, entre otras limitaciones, notables deficiencias en relación con la información disponible sobre las métricas relevantes y su utilidad, la dificultad en la determinación y el cálculo de riesgos, la aplicación limitada de ciclos o lógicas de adaptación, la restricción de las soluciones a dominios y activos específicos, así como la preferencia generalizada por mecanismos de decisión preestablecidos.

Como resultado, se propone un nuevo modelo de Seguridad Adaptativa basado en el riesgo. Así mismo, se diseña y especifica la arquitectura subyacente de este, definiendo también los roles que deben implicarse y sus respectivas responsabilidades. Por último, se fija el flujo de adaptación seleccionado en la propuesta, detallando los pasos y elecciones necesarias para su correcto funcionamiento, y exponiendo las posibilidades de decisión que se pretenden

con el nuevo sistema.

3.1. Justificación de la necesidad de un nuevo modelo

La Seguridad Adaptativa busca ofrecer la protección de diferentes activos, observando su contexto o el riesgo que se está corriendo en un momento concreto, dejando a un lado la rigidez de los controles de seguridad tradicionales, así como la complejidad y el coste que podría suponer su adaptación manual en entornos cambiantes y heterogéneos.

Sin embargo, y aunque es patente su necesidad y utilidad, investigaciones anteriores no han propuesto un modelo o metodología genéricos que permita aplicarla en cualquier —o casi cualquier— dominio. Las propuestas introducidas en el capítulo 2, se centran en el diseño y el desarrollo de diferentes enfoques de Seguridad Adaptativa específicos, para desplegarse en entornos y llevar a cabo adaptaciones muy concretas. En este sentido, se encuentran investigaciones enfocadas en IoT, *Cloud Computing*, móvil, redes corporativas, SDN o aplicaciones web, por mencionar solo algunos ejemplos; las adaptaciones, por su parte, van desde el control de accesos a la gestión del tráfico de red, pasando por la respuesta a incidentes o la modificación de la arquitectura. Esta falta de un modelo o metodología de amplio espectro hace difícil y costoso fusionar las soluciones existentes para obtener enfoques adaptativos que cubran todas las necesidades de una organización, reutilizarlos con diferentes controles de seguridad o ampliar su utilización cuando así se requiera.

Por otro lado, cabe recalcar que para la correcta protección de las organiza-

ciones, es importante encontrar un equilibrio entre seguridad, rendimiento e inversión. En este sentido, es fundamental definir el nivel de seguridad adecuado, considerando tanto sus objetivos, los recursos disponibles (económicos y humanos) y los riesgos potenciales a los que se expone. Para calcular la inversión necesaria en materia de seguridad, no solo se deben tener en cuenta los activos para la protección, sino también aspectos relevantes como el mantenimiento y actualización de estos elementos, el cumplimiento normativo, los costes de formación y los posibles impactos en la reputación que pueden darse si se materializa una amenaza. Una inversión excesiva o la utilización de más controles de los necesarios —o su uso con niveles de paranoia altos— puede disminuir el rendimiento de los controles de seguridad o los activos protegidos por estos, debido a su sobrecarga o a la falta de flexibilidad en la infraestructura; por ello, es necesario establecer una estrategia de inversión equilibrada acorde con el riesgo. Del mismo modo, una escasa inversión puede resultar en una protección inadecuada que dé lugar a la materialización de amenazas, con sus correspondientes inconvenientes. Es necesario un modelo que permita garantizar la seguridad proporcional al riesgo que se está corriendo en cada momento, evitando que los controles de seguridad se mantengan fijos desde su despliegue o que requieran su modificación por parte de operadores humanos.

Teniendo en cuenta lo expuesto, y con el fin de optimizar y mejorar los controles existentes de Seguridad Adaptativa, en esta tesis se propone agregar la capacidad de adaptación desde fuera del control encargado de proteger el activo o conjunto de activos (véase la figura 3.1). Este planteamiento propicia su uso en todo tipo de dominios y facilita la aplicación de la diversidad de adaptaciones de seguridad requeridas, sin importar el control con el que se está tratando o el activo que se quiere proteger. Específicamente, la adaptación, en el nuevo enfoque que se propone, se basa en el riesgo, manteniéndolo en todo momento en niveles tolerables, lo que permite tomar

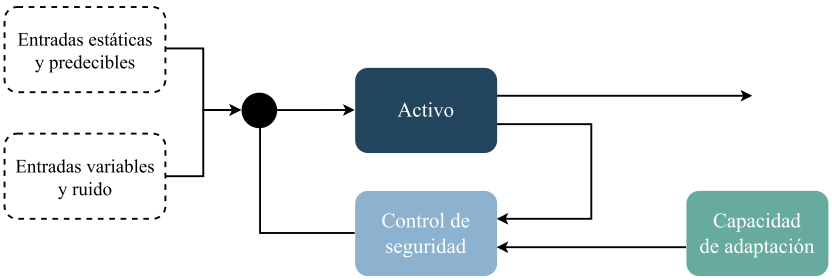

Figura 3.1: Descripción a alto nivel de posibles casos de uso del modelo propuesto.

medidas adecuadas cuando se sobrepasan los umbrales preestablecidos. Este planteamiento implica importantes desafíos de investigación; las siguientes preguntas resumen los más importantes:

- Para dotar de capacidad de adaptación, desde el exterior, a todo tipo de controles de seguridad, ¿qué tipo de arquitectura y flujo de decisiones se puede utilizar?
- ¿Cómo puede facilitarse la gestión y despliegue de este tipo de arquitecturas utilizando, además, unos recursos computacionales limitados?
- ¿Cómo se puede cuantificar el riesgo para orientar las decisiones de adaptación?
- ¿Qué atributos del contexto de un activo se deben medir para saber si la adaptación del control de seguridad es necesaria considerando criterios basados en riesgo a través de indicadores KRIs (*Key Risk Indicators*), IoCs (*Indicators of Compromise*), IoAs (*Indicators of Attack*) o puntuaciones de riesgo?
- ¿Qué tipo de adaptaciones son posibles?
- ¿Cómo se puede decidir qué adaptación es mejor en cada caso?
- ¿Cómo se pueden expresar, de forma genérica (independientemente del control), las adaptaciones permitidas?

- ¿Cómo se puede manifestar la necesidad de adaptación de un control dado?
- ¿Cómo es posible aplicar, de forma simultánea, diferentes adaptaciones en distintos controles de seguridad?
- ¿De qué manera se puede hacer efectiva una adaptación después de haberse tomado la decisión de llevarla a cabo?
- ¿Es posible la reutilización de componentes?

En relación a las categorías de Seguridad Adaptativa detalladas en el capítulo 2, el modelo propuesto encaja en la Seguridad Adaptativa basada en riesgos —cuando se atiende al desencadenante de la adaptación—, en la Seguridad Adaptativa predefinida —al considerar el mecanismo de decisión— y en ninguna de las descritas en la categorización que responde a la adaptación, por tratarse de un modelo genérico capaz de hacer frente a cualquier —o casi cualquier— adaptación.

Las siguientes secciones de este capítulo presentan el nuevo modelo propuesto en esta tesis —al que se ha denominado RiAS (por las siglas de ***Ri****sk-based* ***A****daptive* ***S****ecurity*)—, cuyo objetivo es responder a las cuestiones planteadas.

3.2. Arquitectura subyacente y supuestos

El modelo propuesto, RiAS, se basa, de forma genérica, en un agente de confianza capaz de adaptar los controles y contramedidas de seguridad en función de la evolución del riesgo (véase la figura 3.1). Este se calcula a partir de diferentes eventos, mediciones y atributos del contexto operativo del activo —o activos— que deben protegerse, de la propia organización u otros

aspectos que pueden ser relevantes para su estimación. Además, el agente puede ejecutarse internamente —desplegándose y siendo administrado por la propia organización— o externamente —consumiéndose como servicio—. La decisión de implementar este tipo de estrategia se basa, principalmente, en la más que evidente dirección que está tomando el mundo tecnológico, hacia una desvinculación de los servicios y dispositivos de red físicos, ganando una especial relevancia el modelo «como servicio» o «*as a Service*» y, en general, el *Cloud* [149, 150]. Gracias a esto, se abre la posibilidad a que, tanto las organizaciones que apuestan por un entorno *Cloud*, como aquellas más conservadoras, tengan la posibilidad de utilizar la solución propuesta en esta tesis. La desvinculación de los controles de seguridad de la capacidad de adaptación permite asegurar, de manera dinámica, una variedad de activos heterogéneos en diferentes dominios de aplicación, sin necesidad de contar con un mecanismo de Seguridad Adaptativa para cada uno de ellos.

Por otro lado, tal como se muestra en la figura 3.2, se propone que la arquitectura que soporta RiAS conste de tres capas, que pueden implementarse de forma centralizada (por ejemplo, todas desplegadas en un mismo servidor), distribuida (cada una de ellas en un nodo diferente) o híbrida (algunas de forma centralizada y otras distribuida). La disposición de estas capas dependerá, en gran medida, de los recursos que deben ser protegidos, la envergadura del entorno que es necesario supervisar y, por supuesto, los medios de los que se disponga. Con esto, se pretende facilitar su uso en organizaciones pequeñas o con limitados recursos económicos o cuando existan pocos activos que requieran protección, pero también en proyectos sin restricción financiera o que precisen proteger un gran número de activos; pudiéndose elegir la disposición que más se ajuste a las necesidades e intereses de la propia organización. Estas capas, cuyo objetivo específico se presenta en la sección «3.3.2. Pasos en línea», son: «Medición», «Decisión» y «Adaptación»; y, junto a los pasos

sin conexión necesarios para el despliegue de la propuesta (véase la sección «3.3.1. Pasos fuera de línea»), proporcionan la funcionalidad completa de este nuevo modelo para la Seguridad Adaptativa basado en el riesgo.

Como ha quedado demostrado en el capítulo anterior («2. Estado del arte»), existen diferentes ciclos o lógicas de adaptación con los que se ha experimentado en el pasado en relación con la Seguridad Adaptativa (PDCA, OODA, MAPE-K), siendo MAPE-K el más utilizado —creado desde sus inicios con la idea de llevar a cabo la autoadaptación—. El volumen de investigaciones y usos que se le ha dado a esta lógica de adaptación, además, lo convierte en prácticamente un estándar. MAPE-K ofrece respuestas a todas las preguntas que plantea la creación de RiAS y, por ende, al utilizarlo, facilita su diseño y permite destinar más recursos a otros elementos o componentes de los que no existe tanta evolución o conocimiento al respecto. Por ello, haciendo referencia a la figura 3.2, la arquitectura del modelo presentado puede observarse semejante a la estructura de este ciclo, en el que se basa. En ella, se puede encontrar el cálculo «M» o *Monitor*, el cálculo «A» o *Analyse* y «K» o *Knowledge* de MAPE-K, todos ellos correspondientes a la capa «Medición» de RiAS. Por otro lado, «P» (*Plan*) está contenido en la capa «Decisión»; y «E» o *Execute* se asemeja a la capa «Adaptación» de la propuesta.

RiAS, cuyo flujo de adaptación se presenta en la próxima sección («3.3. Flujo de adaptación»), puede interactuar con tres tipos diferentes de actores (Decisores, Administradores de políticas y Propietarios del control), encargados de tomar ciertas decisiones o llevar a cabo determinadas actividades y proponer soluciones, según corresponda. Las investigaciones anteriores con respecto a la Seguridad Adaptativa se centran en exponer la solución, sin detenerse ni determinar quién debe hacer qué dentro de sus esquemas. Sin embargo, la propuesta planteada también ofrece esta información y es esto por lo que,

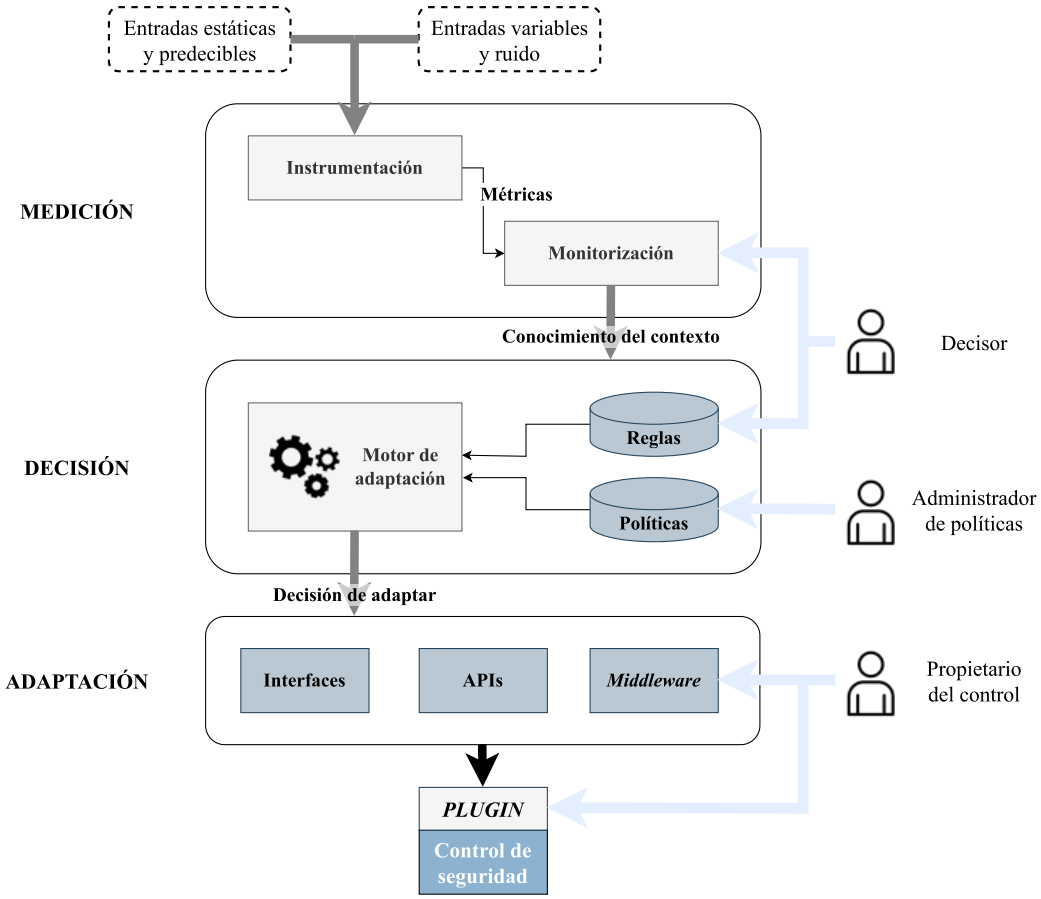

Figura 3.2: Arquitectura de tres capas de RiAS.

además de lo descrito en párrafos anteriores, se ha tomado como base la arquitectura de tres capas que se observa en la figura 3.2. De esta forma, cada uno de los actores —o personal a su cargo—, son los responsables del manejo, preparación y configuración de las capas que forman la solución. Además, al existir tres capas bien diferenciadas horizontalmente y, a su vez, un rol encargado de cada una de estas, la comunicación entre estos es fluida y estandarizada (no siendo necesario relacionarse todos con todos), a pesar de que los objetivos de cada uno de los roles sean diversos. A continuación, se exponen las competencias definidas para ellos dentro del ámbito de RiAS:

1. **Decisor**. Este rol es el responsable de decidir cómo se captura el contexto; incluyendo la recopilación de información y eventos, así como la creación de métricas y disparadores que posteriormente emplean

tanto las políticas como las reglas. También se encarga del desarrollo de los instrumentos (sensores y sondas) necesarios para ello (véase la sección «3.3.1.1. Instrumentos»). Además, su papel incluye la definición de las reglas, las cuales determinan cómo, cuándo y dónde se debe realizar la adaptación de los controles de seguridad y la aplicación de las contramedidas («3.3.1.2. Reglas»). Su comunicación con otros roles se hace por medio de las reglas, las métricas y los disparadores, siendo en las primeras donde es necesario indicar los manejadores a utilizar cuando se requiera aplicar una adaptación, puestos a su disposición por el Propietario del control. A su vez, las reglas y las métricas son empleadas en la capa «Decisión» por parte del Administrador de políticas, concretamente al redactar las políticas; los disparadores se emplean dentro de las reglas (las de tipo «controladas por eventos»), por lo que él mismo es responsable de su utilización.

2. **Administrador de políticas**. Se trata del encargado de definir las políticas que gestionan las adaptaciones (véase la sección «3.3.1.3. Políticas»). Por tanto, su función comprende la traducción, a un lenguaje estándar, de la respuesta a una pregunta crucial: ¿Por qué adaptar un control de seguridad específico? Su trabajo recae en la capa «Decisión» de RiAS. La comunicación con la capa «Medición» se hace a través del uso de las métricas y las reglas proporcionadas por el Decisor, que deben incluirse en las políticas que redacte.

3. **Propietario del control**. Este rol debe integrar la arquitectura propuesta con controles específicos y contramedidas, de tal forma que se ejecuten las adaptaciones decididas. Su cometido es proporcionar los manejadores necesarios para realizar los cambios que se han de llevar a cabo en el control de seguridad, el sistema o el activo que requiere protección, así como aquellos útiles para la comunicación entre

estos y el modelo («3.3.1.4. Manejadores»). El Propietario del control es la persona responsable de la capa «Adaptación». Por su parte, la comunicación con el Decisor se realiza mediante la oferta de los elementos mencionados, los manejadores (interfaces, APIs —Application Programming Interfaces—, *plugin*, *middleware*, etc.), que utilizará en las diferentes reglas.

Las tres capas que componen la arquitectura propuesta para RiAS brindan los recursos necesarios a los actores para poder gestionar los procesos de adaptación (Decisor), gobernar las decisiones (Administrador de políticas) y ajustar los controles de seguridad (Propietario del control). También incluyen los componentes esenciales para seguir el flujo de adaptación descrito en las siguientes secciones, tal y como se presenta en la figura 3.3.

3.3. Flujo de adaptación

El flujo de adaptación de RiAS se ha considerado en dos etapas claramente diferenciadas. La primera engloba todas las acciones que deben realizarse fuera de línea, la parte de la arquitectura previa a su puesta en marcha y que necesita de los roles presentados en secciones anteriores para tomar ciertas decisiones y realizar los preparativos y ajustes necesarios para el correcto funcionamiento, cuando RiAS se está calibrando; en ella, se definen los componentes que la forman, así como el modo de utilizarlos (instrumentos, reglas, políticas y manejadores). La segunda etapa consiste en aquellas acciones y procesos que se ejecutan una vez que el modelo está en funcionamiento, los pasos en línea, que abarca la medición (recuperación de toda la información contextual y los eventos, así como la transformación de estos en métricas y disparadores), la decisión (evaluación de los valores recopilados en la capa de medición

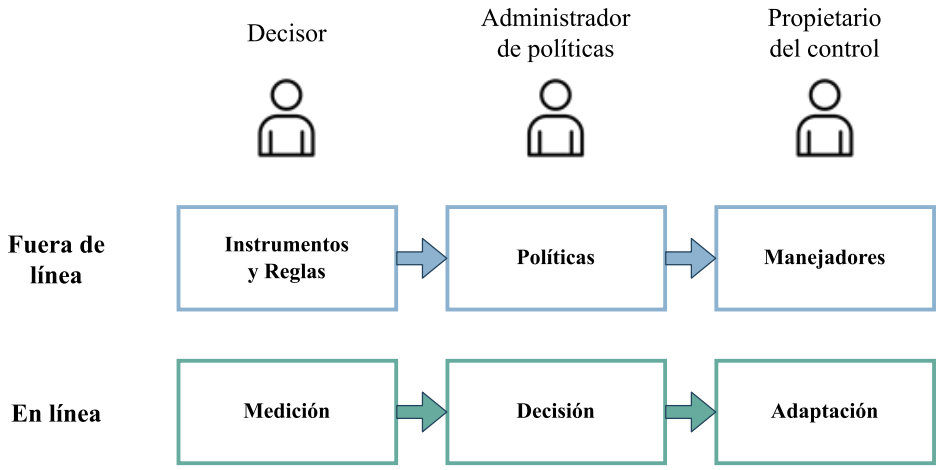

Figura 3.3: Flujo para realizar la adaptación basada en riesgos de los controles de seguridad en RiAS.

para determinar si se deben tomar medidas) y, por último, la adaptación (encargada de aplicar los cambios considerados).

La figura 3.3 expone la relación directa entre los diferentes roles descritos en anteriores secciones y los pasos fuera de línea y en línea que se definen en esta. Como puede observarse, el Decisor es responsable de la creación de los instrumentos para la medición y de las reglas —en relación con los pasos fuera de línea—, siendo también el rol a cargo de la capa de «Medición» —durante los pasos en línea—; por su parte, el Administrador de políticas se encarga de la redacción de las políticas —fuera de línea— y la supervisión de la segunda capa, la de «Decisión» —en línea—; por último, el Propietario del control gestiona los manejadores necesarios para las adaptaciones —fuera de línea—, siendo además la autoridad responsable de la capa de «Adaptación» —en línea—.

3.3.1. Pasos fuera de línea

Para un correcto funcionamiento de RiAS, en primer lugar, es necesario definir las políticas y las reglas de adaptación, así como desarrollar los sensores,

sondas, disparadores, interfaces, APIs, *middleware*, *plugin*, etc. requeridos tanto para la recopilación de información y eventos relacionados con el contexto, como aquellos útiles para aplicar las adaptaciones de los diferentes controles de seguridad. Este proceso se lleva a cabo fuera de línea, es decir, se realiza previo a la puesta en marcha del modelo o durante su funcionamiento, pero sin requerir una interrupción de este; en él, están involucrados todos los roles declarados en secciones anteriores (Decisor, Administrador de políticas y Propietario del control).

Realizar la adaptación de forma eficaz requiere fijar ciertos parámetros de toma de decisiones, así como determinar cuándo se deben aplicar las adaptaciones. Para cubrir esta necesidad, se ha optado por la utilización de políticas y reglas. Otra posibilidad igualmente válida hubiera sido el uso de ficheros de configuración o la utilización de plantillas, por poner algunos ejemplos. Sin embargo, las reglas y políticas permiten una enorme atomización de cada una de las decisiones, posibilitando su modificación de forma independiente y, gracias a la arquitectura del modelo, sin necesidad de parar o reiniciar el sistema cuando alguna de ellas requiera actualizarse.

Además, este conjunto de normas y directrices (políticas y reglas) permiten una división clara entre cómo llevar a cabo una adaptación (reglas) y por qué motivo se debe hacer (políticas), siendo esto lo óptimo para la arquitectura propuesta y los roles que interactúan con ella. De esta forma, el Decisor —quien redacta las reglas—, se encarga de la toma de decisiones, ofreciéndoselas después al Administrador de políticas, que las utiliza cuándo y cómo cree conveniente. A su vez, el Decisor puede usar, en las reglas, todos los mecanismos que el Propietario del control pone a su disposición para aplicar las adaptaciones que más se ajustan a las necesidades de cada caso concreto. Así, todos los roles mantienen un contacto y un intercambio de información fluidos, siendo únicamente necesario adecuarse al estándar de comunicación

que hayan acordado entre ellos.

Otro de los beneficios del uso de políticas y reglas es la sencillez y escalabilidad que aportan al modelo. Esto permite a organizaciones grandes, con un amplio volumen de estos elementos, repartir su gestión entre diferentes recursos humanos (Decisores o Administradores de políticas), no siendo necesario que un único individuo tenga que encargarse de todas y cada una de ellas, lo cual convertiría en imposible, o poco realista, la gestión del sistema y, por ende, su utilidad, dejando de ser una herramienta escalable. También es beneficioso para pequeñas organizaciones, que, gracias a la facilidad de uso y despliegue, no requieren una gran inversión de tiempo y recursos en desarrollar las habilidades necesarias para su manejo.

Aparte de la ya comentada utilidad de políticas y reglas con respecto a su atomicidad y a la posibilidad de realizar cambios sin involucrar a otros elementos o roles, la elección de estos componentes para la definición de RiAS también ha sido impulsada por la enorme conexión que existe entre el ciclo MAPE-K y su empleo para la toma de decisiones. De igual manera, la mayoría de las investigaciones relacionadas con la Seguridad Adaptativa predefinida (véase la sección «2.3.1. Seguridad Adaptativa predefinida») atienden a estos elementos, habiéndose plasmado en ellas su gran utilidad y sencillez para los roles encargados de su gestión.

Por otro lado, se busca impulsar tanto la reutilización como la posibilidad de que se forme una comunidad en la que diferentes roles, de distintas organizaciones, compartan sus conocimientos y desarrollos. Es por ello por lo que se ha determinado el uso de instrumentos y manejadores como sensores, sondas, *plugin*, APIs, interfaces, *middleware*, etc. tanto para la recopilación de información y eventos relacionados con el contexto —instrumentos—, como para aplicar la adaptación en los diferentes controles de seguridad —maneja-

dores—. Además, la compartición propicia el crecimiento de RiAS, mejorando considerablemente los tiempos y costes de despliegue e incrementando sus posibilidades. Estos métodos se encuentran enormemente extendidos en otro tipo de soluciones, como, por ejemplo, sistemas de monitorización y alertas o mecanismos de recopilación de métricas y estadísticas, como ocurre en Nagios [151], Fluentd [152] o Telegraf [153].

La decisión que se plantea también facilita el trabajo de los desarrolladores encargados de la creación de los instrumentos y manejadores, siendo únicamente necesario para ellos seguir las pautas predefinidas y acordadas en relación con las llamadas a estas herramientas y los datos o atributos que deben ser capaces de recibir o enviar (códigos de respuesta, estructura de la información, etc.) tras haberse ejecutado o cuando se ejecuten. Al no existir limitaciones con respecto a estos componentes, los desarrolladores pueden elegir el lenguaje de programación que más conozcan o el que se adapte al sistema operativo, activo, software o recurso a proteger o del que se requiera extraer las mediciones. De igual forma, el protocolo utilizado (HTTP —*Hypertext Transfer Protocol*—, SSH —*Secure Shell*—, *Bluetooth*, etc.) es elección de los desarrolladores y estará marcado por los activos o los controles de seguridad, no por RiAS.

Por tanto, dentro de los pasos fuera de línea pueden encontrarse los instrumentos, las políticas, las reglas y los manejadores; resumidos y detallados en la tabla 3.1, donde se exponen, además, los problemas que estos componentes deben resolver. Es importante recalcar que tanto instrumentos y manejadores, como reglas y políticas, se han definido con la idea de ser revisados, mejorados y ampliados de forma cíclica, permitiendo la incorporación de nueva información o la realización de diferentes adaptaciones, pero también el perfeccionamiento de las existentes.

Tabla 3.1: Resumen de los pasos fuera de línea de RiAS.

Componente	**Problema**	**Alternativas**	**Responsable**
Instrumentos	Cómo medir	Sensor	Decisor
		Sonda	
	Qué medir	Directa	
		Elaborada	
		Monitorizada	
Reglas	Qué adaptar	Parámetros	
		Arquitectura	
		Conducta	
	Cuándo adaptar	Periódica	
		Controlada por eventos	
		Bajo demanda	
	Dónde adaptar	Centralizada	
		Distribuida	
		Híbrida	
Políticas	Cómo adaptar	Reactiva	Administrador de políticas
		Predictiva	
	Por qué adaptar	Cambios en contexto operativo	
		Cambios en activo protegido	
		Cambios en objetivos de seguridad	
Manejadores	Qué artefacto y control	*Plugin*	Propietario del control
		Interfaz	
		API	
		Middleware	

3.3.1.1. Instrumentos

Los dispositivos de instrumentación o instrumentos se corresponden con sensores y sondas, capaces de realizar mediciones o extraer información del entorno, los activos o los controles de seguridad. También se engloban aquí los disparadores de las reglas de tipo «controladas por eventos», de vital importancia para que el modelo actúe adecuadamente, pudiendo nutrirse, cuando es necesario, de las métricas calculadas a partir de la información contextual recopilada.

Los diferentes instrumentos necesarios para la recogida de información pueden ser sensores y sondas. En ambos casos, se trata de dispositivos o software para recopilar información de activos o del entorno, cuya principal diferencia recae en que, normalmente, los sensores operan de forma autónoma y las sondas se integran en un hardware o software específico.

Los Decisores son responsables de diseñar e implementar estos instrumentos, con el fin de decidir «cómo medir el riesgo o recopilar los valores necesarios» para actuar en consecuencia y adaptar cuando así se haya determinado. La principal misión de estos elementos es realizar las mediciones requeridas y su seguimiento, obteniendo, de esta forma, la plena comprensión del entorno y el conocimiento del estado del activo o activos que se pretenden asegurar, así como el de los controles de seguridad o, incluso, el de la organización u otras instituciones interesantes o con ciertas similitudes a la que se quiere proteger. Calculadas basándose en esa información, posteriormente, surgen las métricas, que permiten tomar decisiones, una vez procesadas o analizadas, sobre la necesidad de adecuar —o no— el control o controles de seguridad o los propios activos.

Por otro lado, los disparadores, siendo su diseño e implementación también responsabilidad de los Decisores, son los encargados de indicar que las reglas de tipo «controladas por eventos» se deben evaluar. Estos elementos están diseñados para ejecutarse cuando ciertos sucesos preestablecidos ocurren —normalmente calculados a partir de las mediciones o las métricas—. Para su desarrollo se debe estudiar y plasmar, a bajo nivel, las necesidades en cada caso concreto.

Con el objetivo de cubrir la finalidad prevista, es primordial la comunicación entre Decisores y Administradores de políticas para llegar a un consenso entre las posibilidades, las necesidades y el problema que necesita ser resuelto.

Además, para la creación de estos elementos —los instrumentos— se deben contemplar los estándares impuestos; así, la capa «Medición» de RiAS (véase la figura 3.2), es capaz de almacenar la información recopilada y, además, esta puede recibir —cuando así lo requiera— una retroalimentación que le permita saber si el guardado se ha realizado correctamente o no. Esta normalización facilita el uso de cualquiera de los dispositivos de instrumentación por parte de la capa «Medición», siempre y cuando se haya seguido el estándar predefinido.

3.3.1.2. Reglas

Las reglas se han definido como pequeñas piezas de información encargadas de dar respuesta a las preguntas «qué», «cuándo» y «dónde» adaptar. Se trata de elementos atómicos que son consultados cuando se requiere su aplicación o análisis, no siendo necesario su almacenamiento en memoria o su precarga. El motivo por el cual el funcionamiento se ha determinado de esta manera es, principalmente, facilitar la modificación en vivo, así como la posibilidad de incorporar nuevos manejadores para la adaptación sin que exista una interrupción en la protección de los activos.

Al diseñar las reglas, es importante tener en cuenta «qué se debe cambiar en los controles de seguridad o contramedidas» para que la adaptación deseada se aplique correctamente. En esta línea, se ha considerado que la adaptación puede categorizarse en tres grupos diferentes: parámetros, arquitectura o conducta; todos ellos determinados a partir de la observación de las distintas posibilidades de modificación que pueden existir en los entornos estudiados. Con esta categorización, el cambio que puede efectuar la ejecución de una regla puede verse de forma más clara e inmediata.

- **Parámetros**. En este caso, la adaptación se obtiene mediante la modi-

ficación de la configuración del control de seguridad o del activo, de tal forma que, aunque el control siga siendo el mismo, este actúe de una forma diferente. Se aplica cuando se requiere un ajuste de parámetros o es necesario poner a punto elementos internos o componentes con distintos valores que los que tenía antes de la adaptación. Un ejemplo de esto sería modificar la configuración del servicio SSH para que bloquee los accesos fallidos a partir de cinco intentos, o cambiar la configuración de Gluu [154] de tal forma que comience a solicitar un segundo factor de autenticación a todos los usuarios.

- **Arquitectura**. Este tipo de adaptación se basa en modificaciones estructurales tanto del activo como del control de seguridad encargado de su protección, englobando, además, un despliegue o una interacción diferente de los que ya existen. Las variaciones pueden ser, por ejemplo, la eliminación, la desactivación o la adición de un nuevo elemento, activo o control dentro del contexto que se está protegiendo, pero también un cambio en cómo los diferentes componentes se comunican entre sí. Tendría cabida, en esta categoría, la inserción de un WAF dentro de la infraestructura de una aplicación web para mejorar su protección o el despliegue de más contenedores Docker [155] para minimizar las consecuencias de un ataque de denegación de servicio.

- **Conducta**. Este tipo corresponde cuando la adaptación se realiza gestionando o usando el control de seguridad de forma diferente. Las modificaciones pueden estar encaminadas a restringir o cambiar el protocolo utilizado, modificar el flujo del tráfico de red, alterar una política o un procedimiento, etc. Ejemplos de esto serían modificar la política de contraseñas para requerir que estas sean más complejas o actualizar el listado de dispositivos permitidos para un usuario al utilizar su cuenta corporativa.

La segunda cuestión que los Decisores deben ser capaces de responder, mediante las reglas de adaptación, es «cuándo se deben aplicar cambios en los controles de seguridad o contramedidas» para que la adaptación deseada sea coherente, útil y no suponga un riesgo para los activos. De esta forma se permite que, aunque una regla haya sido activada por una política, esta pueda no ejecutarse en el momento, abriéndose así un gran abanico de posibilidades a la hora de tomar decisiones. Para facilitar la respuesta a esta pregunta, se ha considerado necesario ofrecer tres estrategias de sincronización: periódica, controlada por eventos o bajo demanda; establecidas con base en los riesgos que puede suponer para un sistema la aplicación de ciertos cambios o modificaciones sin tener en cuenta el momento de su ejecución.

- **Periódica**. En este caso, la adaptación se realiza siempre en ventanas de tiempo específicas, determinadas por un periodo fijo. Por ejemplo, una vez cada hora, diariamente, solo una vez en un día determinado, etc. Esto implica que, aunque se decida que es necesaria una adaptación del control de seguridad, esta no se realiza en un momento arbitrario, sino que, de alguna manera, está programada. Con ello, por ejemplo, se pueden evitar adaptaciones que supongan un corte de servicio (reinicio, parada, reconfiguración, etc.) en momentos críticos o de alta concurrencia para el activo o el control de seguridad adaptado por RiAS.

- **Controlada por eventos**. En esta estrategia, tras la decisión de que la adaptación del control de seguridad es necesaria, se requiere que ocurra un evento específico (traducido como un disparador e indicado en la regla de adaptación, que, tras su activación, permanece a la escucha). A menudo, este evento o disparador puede corresponderse con el desencadenante de la adaptación (un cambio en el contexto, en el activo, en el riesgo objetivo, etc.), implicando la aplicación inmediata

tras la decisión de adaptación; aunque no necesariamente debe ser así y puede requerirse que un evento diferente se dé para que la regla pueda aplicarse.

- **Bajo demanda**. En este caso, tras la confirmación de adaptación, es necesario que el Propietario del control dé permiso para efectuar dicha adaptación. Suele utilizarse en entornos o en adaptaciones en las que el cambio que va a efectuarse tiene grandes implicaciones o cuando se requiere la intervención humana previa a su aplicación, por ejemplo, si conlleva el apagado de un servicio crítico.

Otro de los interrogantes que es necesario considerar al crear reglas es «dónde se debe aplicar la adaptación». La respuesta a esta cuestión ayuda a que la adaptación se realice en el lugar correcto, teniendo en cuenta la arquitectura que siguen tanto los activos como los controles de seguridad encargados de protegerlos y las implicaciones que puede tener su ejecución en un emplazamiento inadecuado. Este modelo, como se ha mencionado con anterioridad, consta de tres capas, que pueden implementarse de forma centralizada o distribuida; a su vez, el «Motor de adaptación» de la capa «Decisión», por medio de las reglas —y utilizando los manejadores que interactúan con el control o controles de seguridad—, puede decidir si la adaptación debe ser centralizada, distribuida o híbrida. Esto depende de los recursos disponibles, la demanda del sistema, su arquitectura, el nivel de seguridad en ese momento, etc.

Por lo tanto, para que puedan cubrirse por completo las exigencias de cada caso contemplado, es necesario conocer y combinar las necesidades y las posibilidades —en forma de instrumentos y manejadores—. Las reglas deben especificar estos aspectos ejecutando la adaptación adecuada, en el momento correcto, pero también en el lugar conveniente.

3.3.1.3. Políticas

Las políticas, al igual que las reglas, se han definido para ser elementos atómicos con información relevante para RiAS y su forma de trabajar. Su responsabilidad recae en determinar el motivo —coherente y útil para la mejora de la seguridad— por el que es necesario aplicar una adaptación y cómo llevarla a cabo (apoyándose en las reglas).

La adición, eliminación o modificación de una política no requiere la parada o el reinicio de RiAS, consiguiéndose así una mayor flexibilidad y permitiendo, sin influir en la seguridad de los activos protegidos, utilizar nuevas métricas o eventos para el cálculo del desencadenante o, incluso, introducir reglas modificadas o creadas para cubrir nuevos aspectos.

El proceso de diseño de las políticas conlleva determinar «cómo se deben adaptar los controles de seguridad o contramedidas». La ambición de este nuevo modelo por abarcar todas las bondades de los sistemas de Seguridad Adaptativa analizados en el capítulo 2, así como satisfacer sus carencias, ha empujado a ofrecer políticas que pueden responder a adaptaciones cuando se detectan ciertos cambios, pero también cuando se prevé que estos se van a dar en un futuro próximo o cuando la probabilidad de que ocurran es alta. En consecuencia, la forma de adaptar de las políticas se puede clasificar en: reactiva y predictiva.

- **Reactiva**. Son aquellas que deciden sobre la adaptación cuando se observan cambios específicos en el contexto; estos, actúan como desencadenantes para que la política se evalúe. Un claro ejemplo de este tipo de políticas es cuando el desencadenante se corresponde con un incremento del número de peticiones a una URL (*Uniform Resource Locator*) determinada.

- **Predictiva**. Con la información contextual, son capaces de predecir los posibles cambios o acciones específicas que pueden darse a corto plazo, tratando de ser proactivas y reduciendo, de esta forma, el nivel de riesgo antes de que este incremente. Un ejemplo es aquella política cuyo desencadenante es calculado a partir de TTPs (*Tactics, Techniques, and Procedures*) asociados a un tipo de ataque concreto, habiéndose observado estas en un activo similar al que se está protegiendo.

En segundo lugar, los Administradores de políticas deben ser capaces de conocer «cuál es el motivo por el que es necesario realizar una adaptación»; de esta forma se consigue garantizar que la adaptación deseada tiene una finalidad clara y es útil para mantener los activos con un nivel de protección adecuado. Dentro de las políticas, también es necesario tener en cuenta qué cambios se deben considerar a la hora de decidir sobre la adaptación de los controles. Estos, después, son traducidos en desencadenantes y se encargan, cuando ocurren, de accionar los consecuentes asociados (interpretados en forma de reglas). Para facilitar su comprensión se catalogan en tres tipos diferentes de cambios: en el contexto operativo, en el activo protegido o en los objetivos de seguridad.

- **Cambios en el contexto operativo**. Esta categoría engloba aquellas políticas que deben reaccionar ante variaciones en el entorno del activo o activos protegidos, pero también aquellas que tienen que ver con los controles de seguridad o, incluso, la propia organización u organizaciones con características similares. Como ejemplo, pueden tenerse en cuenta las políticas que prestan atención a ataques perpetrados en instituciones con ciertos paralelismos comerciales a la protegida por RiAS.
- **Cambios en el activo protegido**. En este caso, se trata de políticas que atienden a variaciones, modificaciones o hechos dados en el propio

activo que se está protegiendo (no en su entorno o contexto). Un ejemplo de este tipo son aquellas políticas que tienen en cuenta los niveles de consumo de recursos para aplicar las adaptaciones que corresponda acorde con estos valores o el riesgo calculado a partir de ellos.

- **Cambios en los objetivos de seguridad**. Aquí se engloban las políticas que responden a necesidades asociadas a ajustes de los propósitos o fines predefinidos, como puede ser el cambio en el riesgo tolerado o la persecución de diferentes metas de seguridad. Claro ejemplo de esto son las políticas que condicionan su ejecución a un incremento de la facturación de la organización y, por tanto, se considera que el riesgo de sufrir un ataque aumenta y los activos deben incrementar su nivel de protección.

3.3.1.4. Manejadores

Los manejadores se han diseñado para facilitar la integración y la comunicación entre el gran conglomerado de activos, controles de seguridad y otros elementos —con un alto grado de heterogeneidad— con la capacidad de adaptación que ofrece RiAS. De esta manera, en esta sección se incluyen las interfaces, *plugin*, *middleware* y APIs necesarios para aplicar los cambios y adaptaciones en los controles de seguridad que deban o puedan requerir adaptarse durante alguno de los pasos del flujo de adaptación.

Los Propietarios del control —responsables de los manejadores ligados a la lógica de adaptación—, se encargan de especificar «qué artefactos y acciones se deben utilizar para realizar las diferentes adaptaciones». Pero también son responsables de desarrollar y proporcionar los manejadores requeridos para ello; estos agentes escriben el código fuente y la lógica empleados para aplicar, a bajo nivel, los ajustes que se han determinado necesarios en algún momento

concreto —las acciones—, así como los indispensables para comunicar el control con el modelo —los artefactos—.

En este proceso, el contacto entre Propietarios del control y Decisores es primordial, ya que estos últimos son los que plantean, por medio de las reglas, las dificultades que debe ser capaz de afrontar el modelo y, los Propietarios del control, quienes buscan una solución aplicable mediante sus herramientas; siempre teniendo en cuenta las posibilidades y las limitaciones existentes (sistemas operativos de los activos o los controles de seguridad, lenguajes de programación y protocolos disponibles, capacidades de cómputo de los dispositivos, etc.).

La creación de estas herramientas debe hacerse siguiendo ciertos estándares, de tal forma que facilite tanto la comunicación y las peticiones desde el «Motor de adaptación» cuando deban aplicarse, como la retroalimentación con los resultados tras su aplicación (códigos de error/satisfacción). Esta estandarización permite que el «Motor de adaptación» sea capaz de utilizar cualquier *plugin*, interfaz, API, *middleware*, etc. creado siguiendo los aspectos predefinidos y que tenga como misión adaptar los activos, controles y contramedidas de seguridad en función de la evolución del riesgo.

3.3.2. Pasos en línea

Además de los pasos fuera de línea («3.3.1. Pasos fuera de línea»), se ha definido, diseñado y concebido el flujo, los estados y los procesos que sigue la información —desde sus inicios—, para determinar si es necesaria una adaptación o no cuando se utiliza RiAS. En este proceso, dividido en tres fases (para asemejarlo a la arquitectura de tres capas descrita en la sección «3.2. Arquitectura subyacente y supuestos»), los distintos componentes interactúan

con los instrumentos, las reglas, las políticas y los manejadores para recibir y evaluar la información contextual (etapa de «Medición»), analizar la necesidad de adaptación («Decisión») y, en caso de requerirse, aplicar la modificación pertinente en activos o controles de seguridad («Adaptación»).

Esta rama, aunque más automatizada que la anterior y ligada al sistema autónomo de RiAS, también requiere de los Decisores, los Administradores de políticas y los Propietarios del control. Los roles participantes, además de estar involucrados en las diferentes fases, deben mejorar —gracias a la observación y la comprensión de la ejecución y los resultados— las decisiones y sus motivos (plasmadas en reglas y políticas), las adaptaciones (en forma de manejadores), las mediciones a realizar y los disparadores (utilizando los instrumentos adecuados para ello). Se trata de un sistema vivo, que acepta mejoras y modificaciones en todo momento, de tal forma que la optimización del modelo, así como de la seguridad de los activos que custodia, lo haga más robusto, útil y flexible, ajustándose a los cambios en el contexto o a los nuevos requerimientos.

Por lo tanto, es importante destacar cada una de las partes y lógicas expuestas en las próximas subsecciones, que incluyen la recogida de información del contexto (con los instrumentos), la determinación de si se debe adaptar o no (impulsada por el conocimiento del contexto y especificada en políticas y reglas) y la aplicación de las adaptaciones (utilizando los manejadores diseñados para ello), reflejadas, de forma resumida, en la figura 3.4.

3.3.2.1. Medición

En primer lugar, en este proceso en línea, se requiere establecer unos objetivos claros y comprender la importancia de los elementos responsables de la «Medición». Para garantizar un desempeño adecuado de estos y del

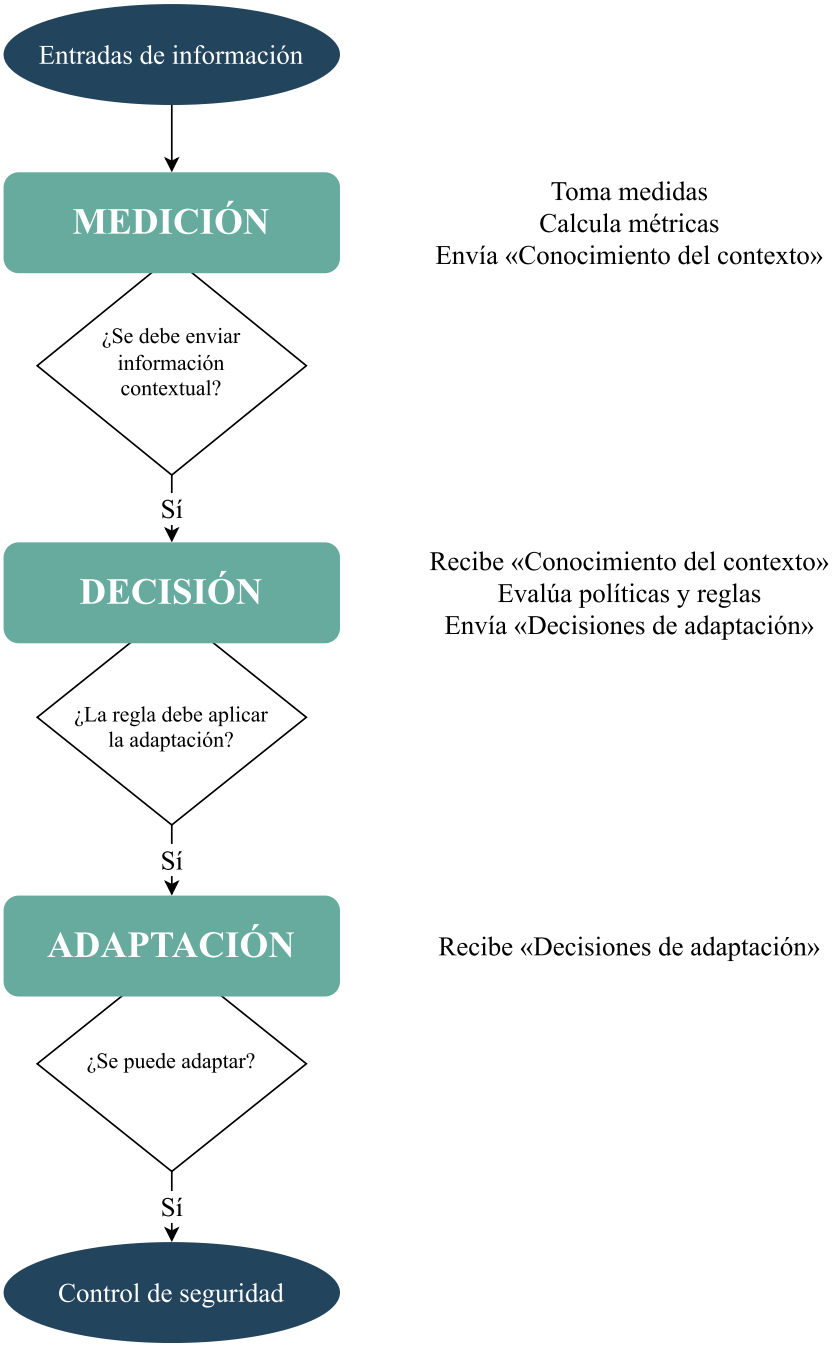

Figura 3.4: Resumen de los pasos en línea de RiAS.

resto de los componentes, es esencial definir la información que RiAS debe recopilar y cómo hacerlo, así como determinar la forma de procesar los datos —transformándolos en métricas— para que resulten útiles al resto de componentes del modelo. Sin una medición adecuada, que incluya la recopilación y el procesamiento de información relevante y valiosa y, por ende, sin un buen conocimiento del contexto, los componentes dependientes de este no

son capaces de cumplir con su función correctamente, pudiendo derivar a actuaciones tardías o respuestas erróneas a la información disponible. Una inadecuada interpretación de las métricas o un cálculo incorrecto del riesgo pueden conducir a resultados similares, aplicándose adaptaciones innecesarias o no haciéndolo cuando sí lo sea.

Dentro de la medición existen dos conceptos relevantes: la monitorización y las métricas; que, aunque estrechamente relacionados, es necesario detallar por separado.

Monitorización. En el ámbito de la ciberseguridad, la monitorización implica recopilar información sobre eventos, alertas o estados específicos en un momento determinado. Estos datos proporcionan un contexto significativo que ayuda a comprender y evaluar, de manera más efectiva, la situación actual. El contexto puede incluir, entre otras, la utilización de memoria de un nodo o el número de peticiones por minuto a un servidor web; pero también información más amplia como la situación de alerta ante ataques terroristas de un determinado país o la opinión pública sobre la organización que se está protegiendo [156, 157].

RiAS ha sido diseñado para considerar no solo los valores recolectados, sino también los desencadenantes definidos por los Decisores, lo que incluye la evaluación de riesgos, mediciones compuestas o eventos puntuales. El conjunto resultante de esta información (métricas de diferentes tipos y eventos) es constantemente monitorizado y sirve de traducción de lo que está ocurriendo en el contexto para la siguiente capa, la de «Decisión». Por tanto, la capa de «Medición», una vez recopilada la información —empleando para ello los instrumentos— y habiendo sido esta traducida a métricas de diversa índole, es capaz de trasladar, cuando corresponde, —para evitar saturar a los demás componentes del modelo— el «Conocimiento del contexto» (en forma de

métricas y disparadores) a la capa de «Decisión» quien, una vez recibido, lo examina para comprobar si se cumple algún desencadenante o disparador asociado a políticas o a reglas.

Métricas. Creadas a partir de las diferentes entradas de información contextual y eventos recopilados mediante los distintos instrumentos, las métricas tratan de traducir los datos a un lenguaje más comprensible y adecuado para RiAS y sus componentes. Sin embargo, todas estas métricas deben reunir características que las hagan útiles para el modelo y faciliten su escalado. Por ello, deben ser precisas, simples, relevantes y estandarizadas.

- **Precisas**. Basándose en datos oportunos y confiables, objetivamente verificables y permitiendo tanto la mejora del conocimiento del contexto, como facilitar futuros cálculos y análisis. Esta característica también contribuye a tomar decisiones más efectivas [158].
- **Simples**. Implica métricas fáciles de medir y comprender, evitando redundancias y correlaciones fuertes entre ellas. Se debe considerar siempre la métrica más útil, permitiendo una toma de decisiones informada y comprensible.
- **Relevantes**. Deben estar altamente relacionadas con el riesgo específico que se quiere evaluar, considerando tanto la probabilidad como el impacto. Además, es necesario establecer un intervalo de tiempo adecuado para la recopilación de los datos.
- **Estandarizadas**. Implica la definición y documentación de las métricas, estableciendo una estructura clara, rastreable y comparable que permita su evaluación diferencial. La estandarización también facilita la colaboración y compartición.

Además de sus características, es importante destacar y conocer los tipos de

métricas que pueden darse. A pesar de que estas categorías no se especifican en las diferentes investigaciones estudiadas en el estado del arte (≪2. Estado del arte≫), se han considerado relevantes para RiAS por la facilidad con la que pueden combinarse, la versatilidad que proporcionan y por su gran estandarización. Sin embargo, las utilizadas en la propuesta no tienen por qué pertenecer a ninguno de estos tipos; se trata de una herramienta con grandes posibilidades, abierta a la mejora y la aplicación de Seguridad Adaptativa en diversos entornos, cambiantes y heterogéneos, aceptando cualquier medición, siempre y cuando los instrumentos de recogida y la creación posterior de las métricas a partir de los datos recopilados sigan los estándares requeridos y se hayan tenido en cuenta las características descritas para su elección. Entre estos tipos se pueden encontrar las métricas directas, las elaboradas y las monitorizadas.

- **Directas**. Consideran aquellas que miden un aspecto específico de un proceso o actividad. Son medidas concretas y particulares que proporcionan información precisa sobre una característica determinada. Por ejemplo, la velocidad de procesamiento de los datos o el número de ventas de un producto.

- **Elaboradas**. Son un conjunto de indicadores generales que permiten evaluar el desempeño, la salud de un sistema/proceso o el riesgo que se está corriendo mediante una perspectiva amplia y global; útiles para tener una visión general del contexto e identificar tendencias y patrones. Se forman a partir de la combinación de otras métricas, siendo ejemplo de ello el índice de exposición al riesgo de la organización o el tiempo medio entre fallos (MTBF o *Mean Time Between Failures*).

- **Monitorizadas**. Consisten en diferentes eventos ocurridos en el contexto que permiten conocer la situación actual o que algo concreto ha

sucedido. Estas métricas, por ejemplo, podrían ser tan simples como alcanzar un determinado nivel de alerta terrorista para un país o la recepción de un correo electrónico.

La toma de decisiones en ciberseguridad se basa, cada vez más, en diferentes indicadores, cuya recopilación y análisis permite evaluar la efectividad de los esfuerzos y conocer la evolución de los riesgos. Además, es útil conservar históricos confiables que respalden la toma de decisiones presentes y futuras [159, 160]. Sin embargo, no existen métricas, indicadores o estándares predefinidos que toda organización deba seguir, sino que estos dependerán de las necesidades y los recursos disponibles de cada una de ellas [161]. La utilidad de estas herramientas se refuerza si sus valores se comparten entre distintas organizaciones, permitiéndose así conocer sucesos que ya han sido experimentados con anterioridad y facilitándose, en el futuro, su detección temprana.

En las investigaciones de Seguridad Adaptativa estudiadas en el capítulo 2, la toma de decisiones suele respaldarse en indicadores, y, aunque en la mayoría de los casos no se menciona el tipo de métrica necesaria o utilizada, se ha conseguido aplicar una catalogación: IoC (*Indicator of Compromise* o Indicador de Compromiso), IoA (*Indicator of Attack* o Indicador de Ataque) y KRI (*Key Risk Indicator* o Indicador Clave de Riesgo) [162, 163].

- **IoCs**. Se corresponden con datos específicos que permiten identificar amenazas en un sistema o red mediante evidencias de que la seguridad ha sido comprometida. No están ligados a un ataque concreto o conocido y son guiados por acciones, herramientas, archivos, protocolos, etc. poco frecuentes percibidos en un activo. Ejemplos de IoCs son direcciones IP sospechosas, características de archivos o comportamientos poco comunes [164–166].

- **IoAs**. Son datos concretos, utilizados para detectar actividades maliciosas, que se conocen por haber sido empleados en ataques o intrusiones con anterioridad. Su uso se destina a evaluar si el activo protegido está siendo atacado o si se están dando las fases previas a ello. Ejemplos de IoAs incluyen direcciones IP, dominios, *hash* de archivos, patrones de comportamiento, TTPs, etc. marcadas como maliciosas o conocidas por haberse visto involucradas en algún ataque [167, 168].

- **KRIs**. Herramientas de gestión que permiten medir y monitorizar los riesgos más críticos a los que se enfrenta una organización, activo o conjunto de estos. Proporcionan una visión general del riesgo mediante medidas específicas y cuantificables que indican la probabilidad y gravedad de un evento potencial que puede afectar negativamente. Permiten determinar qué partes de un proces deben cambiarse para reducir la exposición al riesgo. Ejemplos de ello son el tiempo de actividad del sistema, el porcentaje de correos marcados como *spam* o la cantidad de incidentes de seguridad [169–172].

3.3.2.2. Decisión

RiAS puede identificar la necesidad, momento, lugar, manera y razón para llevar a cabo una adaptación; o lo que es lo mismo, es capaz de tomar una «Decisión» basándose en su conocimiento del contexto operativo. Esta segunda fase, en continuo juicio por parte del «Motor de adaptación» (véase la figura 3.2), es la encargada de recibir las métricas y los disparadores. Utilizando esta información y considerando las políticas y reglas, es capaz de evaluar: (1) si debe aplicarse una adaptación o no a partir del cálculo del desencadenante utilizando el «Conocimiento del contexto»; (2) cuándo es necesario hacerlo; (3) qué es lo que se adaptará; y (4) dónde se aplicará la

adaptación.

Se ha determinado que la definición de reglas y políticas necesarias para la toma de decisiones se haga con semántica común. Con esto, se simplifica la comprensión y la implementación, eliminando el requisito de una documentación exhaustiva —gracias a que la propia definición es autodescriptiva—; así, de un solo vistazo, se puede entender el objetivo de estos elementos. De la misma forma, si es necesaria su modificación (ya sea por el Decisor o Administrador de políticas que las creó o por otro diferente), la persona responsable puede comprenderlas sin dificultad y sin requerir una gran dedicación. La lógica de decisión, por su parte, gracias a su claridad y estructura, es capaz de validarlas, explorarlas y evaluarlas sin necesidad de una gran inversión de recursos computacionales. Este diseño permite, entre otras, mejorar la escalabilidad del modelo y reducir la curva de aprendizaje.

El formato propuesto para la redacción de reglas y políticas en RiAS es JSON (*JavaScript Object Notation*). Esta decisión, además de contemplar lo descrito en el párrafo anterior, ha sido impulsada por ser un lenguaje que permite el intercambio de objetos de datos (pares atributo-valor y matrices u otros elementos que pueden serializarse) utilizando texto legible por humanos. Además, es un formato independiente del lenguaje de programación y puede ser generado y analizado también por aplicaciones y servidores. De esta forma, la definición de políticas y reglas se hace más sencilla, sin requerir una transformación para que RiAS pueda comprenderlas.

La sección «3.3.1. Pasos fuera de línea» brinda una descripción detallada para entender cómo deben redactarse o los requerimientos que deben tenerse en cuenta en las políticas y las reglas. A continuación, se describe cómo el modelo puede evaluarlas, así como cada uno de los atributos que deben contener para asegurar buenos resultados.

Políticas. La lógica de decisión requiere que estos elementos estén formados por distintos atributos, que, en su creación, son completados acorde con el objetivo buscado. Estas propiedades se corresponden con: el nombre, el propietario, el tipo, el identificador de los controles y las condiciones de adaptación.

- **Nombre**. Se trata del identificador, un nombre descriptivo —único— que permite reconocer una política concreta y su funcionalidad. El «Motor de adaptación» de RiAS únicamente utiliza este atributo para su descubrimiento.

- **Propietario**. Indica el correo electrónico del Administrador de políticas responsable; encargado de su definición, gestión o actualización. En este caso, sirve al «Motor de adaptación» para enviar correos electrónicos, de forma automática, cuando así se requiera (por ejemplo, si existen errores en su ejecución, si la política está mal formada, etc.).

- **Tipo**. En este atributo se indica el tipo de política de la que se trata. En el caso de RiAS, se permiten políticas reactivas (cuando reaccionan ante cambios específicos del contexto) o predictivas (si su posible evaluación depende de predicciones sobre ciertos cambios o acciones). El «Motor de adaptación» no utiliza esta información, ya que es traducida por Decisores en métricas y disparadores, de tal forma que la activación de la política se hace a partir de un desencadenante calculado con estos datos, sin importar, para esta lógica central —la capa de «Decisión»—, cómo ha sido estimado.

- **Controles**. Se trata de un atributo meramente informativo que ayuda a comprender a qué afecta la política en sí. En él se indica el control o conjunto de controles afectados por la política. Es posible utilizar

diferentes esquemas para especificarlo, compuestos por números e identificadores en distintos escenarios, de tal forma que, de un simple vistazo, permita conocer el objetivo buscado. Estos pueden coincidir, o integrarse con, por ejemplo, soluciones de inventario de activos o de escaneo de vulnerabilidades.

- **Condiciones de adaptación**. Este elemento lo componen los diferentes predicados que forman la política, especificando, por pares, el cuándo reaccionar (antecedente) y el cómo hacerlo (consecuente).

    - El antecedente es el conjunto de normas que accionan el consecuente. Se trata de los desencadenantes que, calculados a partir de las métricas recibidas, permiten evaluar el consecuente. Es importante recalcar que se requiere que todos los antecedentes de un mismo predicado se cumplan para que los consecuentes relacionados sean evaluados.
    - El consecuente se traduce en las diferentes reglas ligadas a la política, que son llamadas para su evaluación cuando se observan todos los antecedentes asociados.

Reglas. Se trata del consecuente de las condiciones de adaptación de las políticas, aplicable cuando se dan todos los antecedentes vinculados. Para su evaluación, necesitan ciertos atributos, acorde con la finalidad esperada y que se corresponden con: el nombre, el propietario, la sincronización, la categoría y el control.

- **Nombre**. Se trata de su identificador, que facilita conocer, a simple vista, su utilidad. Su valor debe ser único, ya que es utilizado por el ≪Motor de adaptación≫ para encontrarla la regla fácilmente.

- **Propietario**. Indica el correo electrónico del responsable de su correcta definición, gestión o actualización. Se trata de un atributo que permite comunicar errores de validación de la regla, fallos en la ejecución o informar de los pasos a seguir cuando se debe aplicar una regla de tipo «bajo demanda».

- **Sincronización**. Permite especificar el tipo de regla tratada, en relación con el momento en el que debe ejecutarse la adaptación, según lo cual actuará el «Motor de adaptación». Aquí, también se incluye la información necesaria para su actuación: un periodo de tiempo —si se trata de reglas «periódicas»—; el disparador o disparadores —cuando se corresponden con reglas «controladas por eventos»—; o ninguna información adicional —en las reglas «bajo demanda»—.

- **Categoría**. En este apartado se detalla la clase de la regla. Esta puede ser «paramétrica», si la modificación es sobre la configuración del control de seguridad o el activo; «arquitectónica», cuando se dan modificaciones estructurales; o «de conducta», en los casos en los que se utiliza el control de seguridad de una forma diferente. Es un campo meramente informativo que sirve a los Decisores para comprender la propia regla y las implicaciones que conlleva aplicar un control.

- **Control**. Incluye una lista de controles necesarios para aplicar la adaptación. Estos se forman a partir de pares artefacto-acción.

    - El artefacto consiste en el conjunto de interfaces, APIs o *middleware* específicos que se requieren para la comunicación entre el «Motor de adaptación» y el activo o el control de seguridad, a través de la cual se aplica la acción.

    - La acción se corresponde con el *script* o *plugin* que debe ejecutarse directamente en el control de seguridad para aplicar la modificación

Tabla 3.2: Ejemplo de política para RiAS.

Nombre: Adaptación del *Firewall*
Propietario: apoliticas@apoliticas.local
Control: *Firewall* corporativo
Tipo: Predictiva
Condiciones de adaptación:
Predicado 1: *Observado*(Desencadenante1 cuándo $R1 > X$)$\rightarrow$ *Regla*1
Predicado 2: *Observado*(Desencadenante2 cuándo $R2 > Y$)$\rightarrow$ *Regla*2

o reajuste. Se trata de un elemento a bajo nivel capaz de editar configuraciones, gestionar servicios, desplegar recursos, etc.

El «Motor de adaptación» de RiAS es el responsable de la evaluación de políticas y, en él, es donde los predicados de estas son verificados. Si la parte antecedente de un predicado es verdadera (si el cálculo del desencadenante resulta cierto), se evalúa la regla o el conjunto de reglas especificadas en la parte consecuente. Después, cuando una regla es examinada, este componente de la capa de decisión tiene la capacidad de mantenerse a la escucha para la activación manual de los controles (en reglas «bajo demanda»); de prestar atención a la capa de «Medición», esperando los disparadores y reaccionando a ellos cuando ocurran (en reglas «controladas por eventos»); e, incluso, de trabajar con el reloj, activando controles en un momento determinado (para reglas de tipo «periódica»).

De esta forma, se pretende considerar las posibilidades más frecuentes identificadas a través del estudio realizado en el capítulo 2. Además, la estructura de políticas y reglas, junto con el diseño y la capacidad de análisis del «Motor de adaptación», permite contemplar otros modos de adaptación en el futuro, si estos se consideran necesarios o útiles. Así, RiAS estará preparado para afrontar nuevos retos y adecuarse a ellos de manera efectiva.

Tabla 3.3: Ejemplo de reglas para RiAS.

Nombre: Regla1 **Propietario**: decisor@decisor.local **Sincronización**: *Controlado por eventos*: *Observado*(Evento1) **Categoría:** Paramétrica **Controles:** **Acción:** *Firewall* corporativo *Aplicar*(modo restrictivo) **Artefacto:** *Usar*(API de configuración del *firewall*)
Nombre: Regla2 **Propietario**: decisor@decisor.local **Sincronización**: *Bajo demanda* **Categoría**: Arquitectónica **Controles**: **Acción**: *Firewall* corporativo *Aplicar*(módulo anti-DDoS) **Artefacto**: *Usar*(*middleware* de orquestación de la red)

Para ejemplificar lo descrito en esta sección, se parte del supuesto: *Un Administrador de políticas y un Decisor, utilizando RiAS, necesitan escribir ciertas políticas y reglas para que la configuración de un control del perímetro de la red se modifique, de forma proactiva, cuando se detecten cambios en el contexto que indiquen que se está produciendo un ataque DDoS o que este puede llegar a producirse y, con ello, minimizar los daños ocasionados.* Para ello, de forma general, deberían seguir los siguientes pasos:

1. Escribir una política de adaptación similar a la que se expone en la tabla 3.2. En ella, se establece que, si una puntuación de riesgo (≪R1≫) se encuentra por encima de un umbral ≪X≫, esta generará un desencadenante, denominado ≪Desencadenante1≫ (estos indicadores o

métricas, para el Administrador de políticas, ponen de manifiesto que un ataque DDoS se está produciendo o puede llegar a producirse). Por tanto, a la recepción del valor de riesgo y tras la comprobación de que se supera ese límite preestablecido, se evalúa cierta regla (≪Regla1≫). Por otro lado, el segundo predicado de esta política determina que, si la puntuación de un segundo riesgo (≪R2≫) sobrepasa un umbral ≪Y≫, se corresponde con un desencadenante ≪Desencadenante2≫ (cuando se estima que el ataque ha aumentado su intensidad), el cual indica que la ≪Regla2≫ se debe evaluar.

2. Redactar una primera regla (≪Regla1≫) similar a la expuesta en la tabla 3.3. Se trata de una regla ≪controlada por eventos≫ que, con su evaluación, activa un determinado control, encargado de configurar un *firewall* para que este actúe de una forma más restrictiva (la acción del control dentro de la definición de la regla). La adaptación, en este caso, se realiza a través de la API de configuración del propio *firewall* (el artefacto del control en la descripción de la regla) tan pronto como la capa de ≪Medición≫ lance el disparador ≪Evento1≫.

3. Escribir una segunda regla (≪Regla2≫) conforme a la expuesta en la tabla 3.3. En este caso, de tipo ≪bajo demanda≫, capaz de modificar la arquitectura del *firewall* (la acción del control), agregando un nuevo módulo anti-DDoS cuando se evalúa y ejecuta. Aquí, la adaptación se aplica a través del *middleware* de orquestación de la red (artefacto del control), cuando el propio Decisor otorga su permiso para llevarlo a cabo (debido a que esta adaptación tiene una gran complejidad y requiere intervenciones manuales).

3.3.2.3. Adaptación

Cuando se recibe la solicitud de adaptación, durante la tercera y última fase, esta es aplicada en el control o controles de seguridad que corresponde (también determinados en la propia solicitud), empleando para ello los manejadores —artefactos y acciones—. Estas adaptaciones se ejecutan desde la capacidad de adaptación —RiAS— (fuera del propio control o activo), atendiendo a las características y posibilidades disponibles (sistema operativo, protocolos, puertos, software, etc.) de los componentes a adaptar.

Es necesario hacer hincapié en la relevancia de la «Adaptación», en su funcionalidad y sus capacidades. Estas adaptaciones son las encargadas de realizar los cambios en los activos, controles de seguridad, infraestructura, etc. de la organización cuando corresponde, ajustándose dinámicamente, de esta forma, a los requisitos del contexto operativo dados en cada momento; analizados con base en las mediciones, determinados por las decisiones y ejecutados por las adaptaciones.

Si bien la lógica de esta fase no es compleja, es fundamental entender que la dificultad aquí recae en los manejadores, que deben ser desarrollados —contemplando las limitaciones existentes— con el objetivo de: (1) establecer comunicaciones precisas entre la capacidad de adaptación y los controles de seguridad o los activos —correspondientes a los artefactos definidos en las reglas—; y (2) realizar las modificaciones necesarias cuando se haya tomado la decisión de adaptar —las denominadas acciones dentro de las reglas utilizadas por RiAS—. Esto convierte a los artefactos en un mero puente de comunicación entre el «Motor de adaptación» y el activo o control de seguridad en el que se debe aplicar la adaptación; la acción, por su parte, consiste en un *script* o fragmento de código que, actuando directamente sobre el activo o el control de seguridad, es capaz de ejecutar las modificaciones

para las que ha sido diseñado.

RiAS se ha concebido para que tanto la reutilización como el uso general del sistema en cualquier activo, servicio, dominio o arquitectura sean sus principales características; la separación en artefactos y acciones contribuye a facilitar y mejorar estas propiedades. En el caso de los artefactos, es posible su aprovechamiento para la comunicación de diferentes controles de seguridad con el «Motor de adaptación», pero también para que distintas organizaciones puedan mantener esa interacción; además, puede darse la situación en la que se requieran los mismos artefactos para adaptar diferentes controles de seguridad, pero también que se necesiten diversos artefactos para realizar distintas adaptaciones —de tipo paramétrico, arquitectónico o de conducta— sobre el mismo control. Por otro lado, las acciones también pueden ser reutilizadas en la adaptación de distintos activos o controles de seguridad, siempre y cuando estos tengan atributos similares.

Es importante tener en cuenta una serie de características al momento de desarrollar los manejadores, de tal forma que estos sean útiles y reutilizables, pero también para que la integración con el resto de los componentes de RiAS sea satisfactoria y que su utilización requiera los mínimos recursos. Cumplir con estas propiedades es sencillo para los artefactos, por ser independientes del control de seguridad y emplearse únicamente para la comunicación; sin embargo, para las acciones, estas cualidades son difíciles de alcanzar por su vinculación directa con el control de seguridad y los cambios que han de realizarse en él. En cualquiera de los casos, no son imprescindibles, pero sí aconsejables y deben perseguirse siempre y cuando sea posible. Estas características incluyen la sencillez, el desacoplamiento, la organización y su enfoque hacia la automatización.

- **Sencillo**. Deben ser elementales y fáciles de comprender, permitiendo

abstracciones efectivas y una lógica clara y uniforme para que el ≪Motor de adaptación≫ de RiAS pueda descubrirlos y utilizarlos fácilmente. Cuanto más simplificado y abstracto, menor será el consumo de recursos por parte de los demás componentes.

- **Desacoplado**. Se debe perseguir que los indicadores no estén estrechamente vinculados al control de seguridad o al activo, de manera que puedan ser utilizados para múltiples adaptaciones diferentes. Esta propiedad puede dividirse, a su vez, en:

  - Desacoplamiento de controles de seguridad concretos, tratando de emplear especificaciones estándar, protocolos, semántica, etc.; no depender de un producto concreto, intentar llegar a la genericidad.
  - Desacoplamiento de los detalles de implementación específicos. Dado que la decisión de adaptación puede ser centralizada, distribuida o híbrida, las capas de arquitectura que soportan el modelo presentan la posibilidad de implementarse de diferentes formas. Es fundamental que los manejadores puedan consumirse desde la capa de adaptación, con independencia de donde se esté ejecutando.

- **Organizado**. La organización tiene que ser rigurosa, estructurándose en función de las capacidades o su funcionalidad, no de la tecnología; el código y la lógica de adaptación deben dividirse en torno al propósito. Artefactos y acciones requieren una implementación de software de pila amplia para conseguir esa capacidad. En consecuencia, estos manejadores se consideran multifuncionales, aumenta la reutilización y evita cambios complejos cuando se agrega o actualiza una regla o un control de seguridad.

- **Centrado en la automatización**. La idea principal debe ser la automatización, minimizando la intervención humana para realizar la

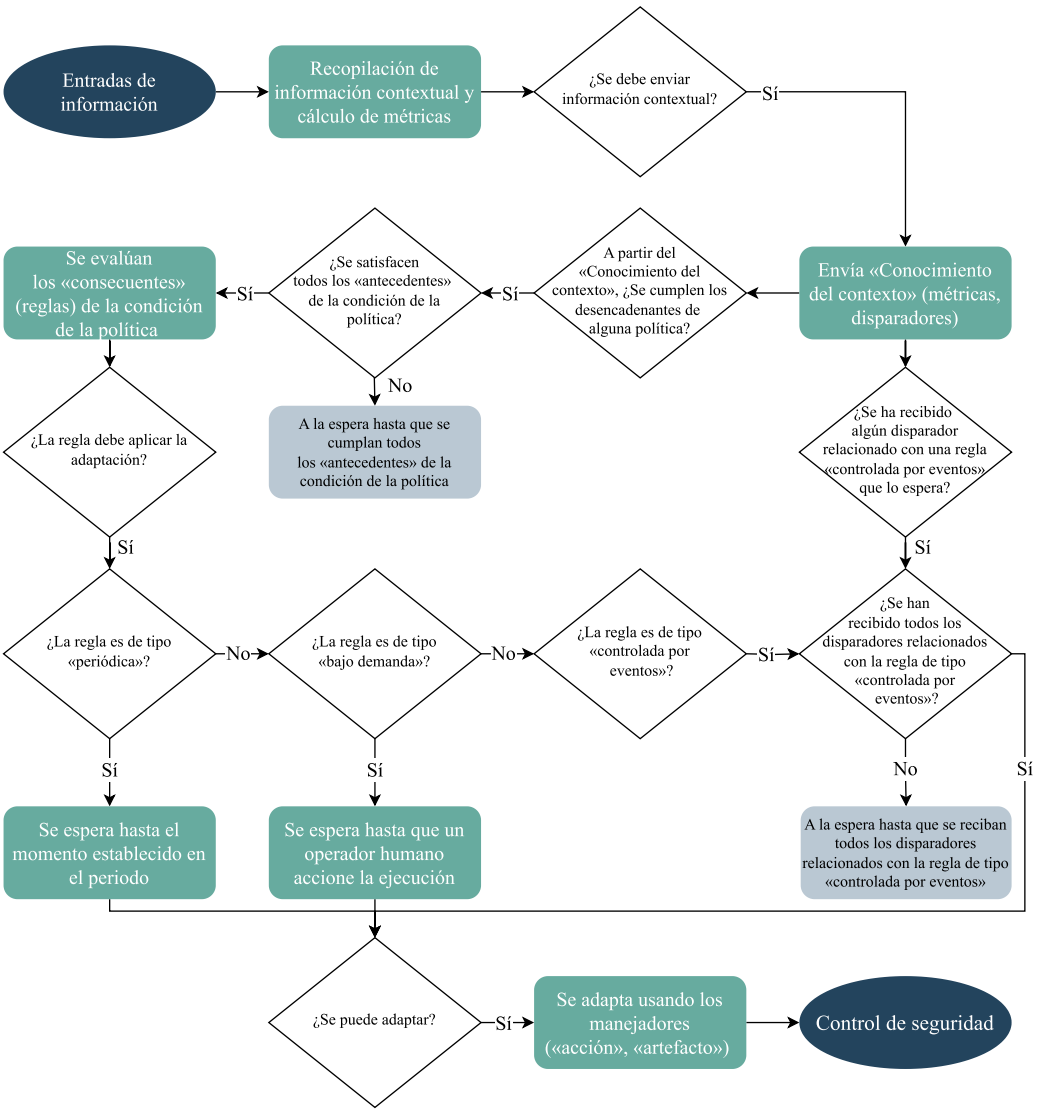

Figura 3.5: Resumen completo del flujo de adaptación de RiAS.

adaptación. Además, su diseño debe ser tolerante a fallos, evitando bloqueos en los controles o los activos y siendo capaces de gestionar diferentes versiones y de volver a estados anteriores en caso de error durante el proceso de adaptación.

3.4. Resumen funcional

El funcionamiento de RiAS, explicado de forma resumida a continuación, puede verse de una forma más detallada en la figura 3.5.

El despliegue y funcionamiento del modelo propuesto comienza con un riguroso entendimiento del contexto operativo, así como de los activos que deben protegerse y de todas las variables y atributos que pueden ser útiles para determinar el riesgo que la organización o sus activos pueden estar corriendo en un momento específico. Tras el análisis de toda la información necesaria, es importante realizar los desarrollos que corresponda con respecto a los instrumentos de la lógica de medición (sensores y sondas), así como los disparadores para la activación de reglas «controladas por eventos». Al mismo tiempo, después de evaluar y comprender los activos que han de ser protegidos, así como los controles de seguridad necesarios o disponibles, se deben desarrollar los manejadores para la adaptación, entre los que se encuentran las interfaces, *plugin*, *middleware* y APIs. Por último, en esta primera fase («3.3.1. Pasos fuera de línea»), se escriben las reglas y políticas, aprovechando en ellas los instrumentos y manejadores creados y el exhaustivo análisis del contexto.

Por otro lado, durante la segunda fase («3.3.2. Pasos en línea»), una vez se han recopilado y evaluado las mediciones (en la capa de «Medición» de RiAS), se calcula, a partir de ellas, las métricas y disparadores necesarios para, cuando el componente «Monitorización» lo estime oportuno, realice el traslado del «Conocimiento del contexto» a la siguiente capa, la de «Decisión». Este proceso es constante, la medición se hace de forma periódica para que RiAS pueda conocer el estado del contexto en todo momento y, si es necesario, indique estos detalles al resto de los componentes. Cuando el «Motor de adaptación» recibe las métricas y los disparadores, se encarga de analizarlos para conocer si se cumple algún desencadenante y, a partir de ello, analizar las políticas y reglas asociadas, de tal forma que determina si es necesario aplicar cierta adaptación o no. En caso de requerirse, esta necesidad se transmite a la capa «Adaptación» que, en consecuencia y empleando los

manejadores indicados en las reglas, realiza los cambios pertinentes para la correcta protección del activo o activos supervisados.

Capítulo 4

Validación mediante casos de uso

En el presente capítulo, se exponen los escenarios utilizados para la validación del modelo definido en esta tesis, RiAS. El propósito es demostrar tanto su correcto funcionamiento como los beneficios aportados. Para ello, se emplean situaciones reales y se resuelven problemas cotidianos del ámbito de la ciberseguridad a los que RiAS debe ser capaz de enfrentarse.

En las siguientes secciones se detallan los dos Casos de Uso (CU) utilizados para ello. Antes de abordar el tema en cuestión, se introduce con un contexto y el estado del arte relacionado con cada uno. Después, se presenta la arquitectura desplegada, así como el prototipo implementado para la validación de RiAS, siguiendo las diferentes capas del modelo («Medición», «Decisión» y «Adaptación»). Por último, se discuten los resultados obtenidos.

4.1. CU 1: Adaptación de una capacidad de protección

Los *Web Application Firewalls*, también conocidos como WAFs, son herramientas de seguridad que protegen aplicaciones web de manera eficiente y económica. Estas soluciones tienen la función de filtrar, supervisar y bloquear el tráfico HTTP. Los WAFs tradicionales se basan en reglas y políticas fijas, lo que ha demostrado tener importantes limitaciones en contextos específicos; en la protección de aplicaciones web no existen configuraciones estáticas útiles, eficaces y lo suficientemente generales como para enfrentarse a los riesgos en todas las situaciones y escenarios posibles [173]. Además, la variedad y complejidad de los nuevos modelos de ataque y agentes de amenaza hacen que las organizaciones tengan que invertir grandes esfuerzos —económicos y humanos— en mantener y actualizar las reglas de los WAFs y las firmas asociadas a los ataques o patrones de tráfico [174].

Ante esto, se propone, implementa, valida y evalúa la adaptación dinámica del WAF mediante el modelo RiAS, propuesto en esta tesis. Con este caso de uso para la adaptación de una capacidad de protección, las reglas y políticas del WAF se ajustan al entorno y al estado del activo. El objetivo es modificar, de forma autónoma, el comportamiento de estas contramedidas, monitorizando el propio WAF, la aplicación web y su contexto para, con ello, cuantificar el riesgo y mantenerlo en el nivel predefinido y deseado.

4.1.1. Contexto

Los WAFs ofrecen un nivel adicional de protección mediante la inspección del tráfico en la capa de aplicación del modelo OSI (*Open System Interconnection*),

con el fin de proteger los sitios, aplicaciones y servicios web; examinando, principalmente, el flujo HTTP. Estas herramientas se despliegan entre el servidor que provee el sitio/aplicación/servicio web e Internet, analizando las peticiones antes de su llegada y de la salida de las respuestas, actuando como un escudo para el activo protegido [175, 176].

Existen principalmente tres tipos de WAFs: aquellos basados en firmas, los que lo hacen en anomalías y los que se apoyan en aprendizaje automático.

- Los WAFs basados en firmas son los más comunes. Su funcionamiento se respalda en la búsqueda de firmas asociadas a patrones de ataque conocidos, lo que requiere una actualización constante, especialmente cada vez que se descubren nuevas técnicas ofensivas. Además, estos sistemas no son capaces de identificar ataques *zero-day* [177].
- Los WAFs basados en anomalías requieren un conocimiento previo sobre el tráfico normal o legítimo que va hacia/desde el servidor web protegido. En este tipo de soluciones, las reglas se aplican cuando se detecta una anomalía en el tráfico; sin embargo, sigue siendo necesaria la actualización de los patrones normales o legítimos, sobre todo cuando el contexto cambia. Si se configuran correctamente, pueden hacer frente a ataques *zero-day* [178, 179].
- Los WAFs basados en aprendizaje automático utilizan modelos entrenados con ML para clasificar las peticiones como maliciosas o no. Este enfoque les permite superar las limitaciones de los otros tipos de WAFs, disminuir las tasas de falsos positivos/negativos y reducir la carga relacionada con las actualizaciones de firmas y estados. No obstante, es necesario entrenar y reentrenar los modelos [177, 180].

La mejora de los WAFs a través del uso de técnicas de ML es una tendencia

creciente en la investigación de la seguridad web [181–187]. Según los estudios analizados, el uso de aprendizaje automático en los WAFs ofrece buenos resultados al detectar ataques *zero-day* y evitar actualizaciones constantes de firmas y patrones de tráfico. Sin embargo, esto también presenta nuevos desafíos que pueden ser difíciles de resolver y requieren una gran inversión de recursos humanos, económicos y de conocimiento.

Uno de los problemas más destacados es la optimización a gran escala, observado principalmente al entrenar modelos matemáticos utilizados en el aprendizaje automático. La preocupación por el envenenamiento de datos de entrenamiento es otra dificultad a la que se enfrentan los WAFs de este tipo. Es difícil saber si la información utilizada para entrenar los algoritmos ha sido manipulada, y esto puede conducir a falsos positivos/negativos, lo que se traslada a la ineficacia de los WAFs [188, 189].

Además, otra complejidad es la necesidad de reentrenar los modelos periódicamente y el coste computacional que esto implica. Estos procesos también requieren una gran inversión de tiempo y recursos humanos, así como la disponibilidad de conjuntos de datos adecuados que cumplan las expectativas de la organización [190, 191].

Para abordar estos desafíos, se propone un enfoque alternativo, aplicando el modelo propuesto en esta tesis, RiAS; un WAF gestionado por un motor de adaptación externo. Esta solución supera algunas de las limitaciones observadas en las herramientas tradicionales, sin necesidad de usar ML y evitando así todas las dificultades asociadas al uso de este tipo de técnicas.

4.1.2. Arquitectura propuesta

En el primer caso de uso, RiAS se valida empleándolo para adaptar una capacidad de protección. Gracias a la estrategia de separación de preocupaciones [192] utilizada, el WAF tradicional puede adquirir competencias de adaptación desde fuera del sistema, sin necesidad de rediseñarlo ni añadir complejidad o sobrecoste. Esto, además, permite adaptar diferentes capacidades de forma concurrente, no únicamente las protecciones que brinda el WAF.

RiAS cuenta, en este caso, con las tres capas necesarias para aplicar las adaptaciones pertinentes —«Medición», «Decisión» y «Adaptación»—. En este sentido, no existen variaciones estructurales o grandes cambios con respecto al modelo original, desplegándose todas las capas en un mismo servidor —de forma centralizada—.

Además, dentro de la arquitectura propuesta, se incluyen los controles de seguridad necesarios para aumentar o disminuir la protección y sobre los que se aplica la adaptación cuando corresponde —el WAF— y el activo o activos protegidos —servidores de bases de datos y web—.

De esta forma, cuando se detectan cambios significativos en el contexto, correspondientes al entorno, los *logs* de los activos protegidos, así como el rendimiento de estos, se trasladan mediante los instrumentos a la capa de «Medición» de RiAS. Esta, con el proceso habitual, los convierte en métricas y disparadores que recibe, cuando así se estima, la siguiente capa —la de «Decisión»—. Por último, se decide si es necesario modificar el WAF en función de las políticas y reglas establecidas, realizándose los cambios desde la capa de «Adaptación».

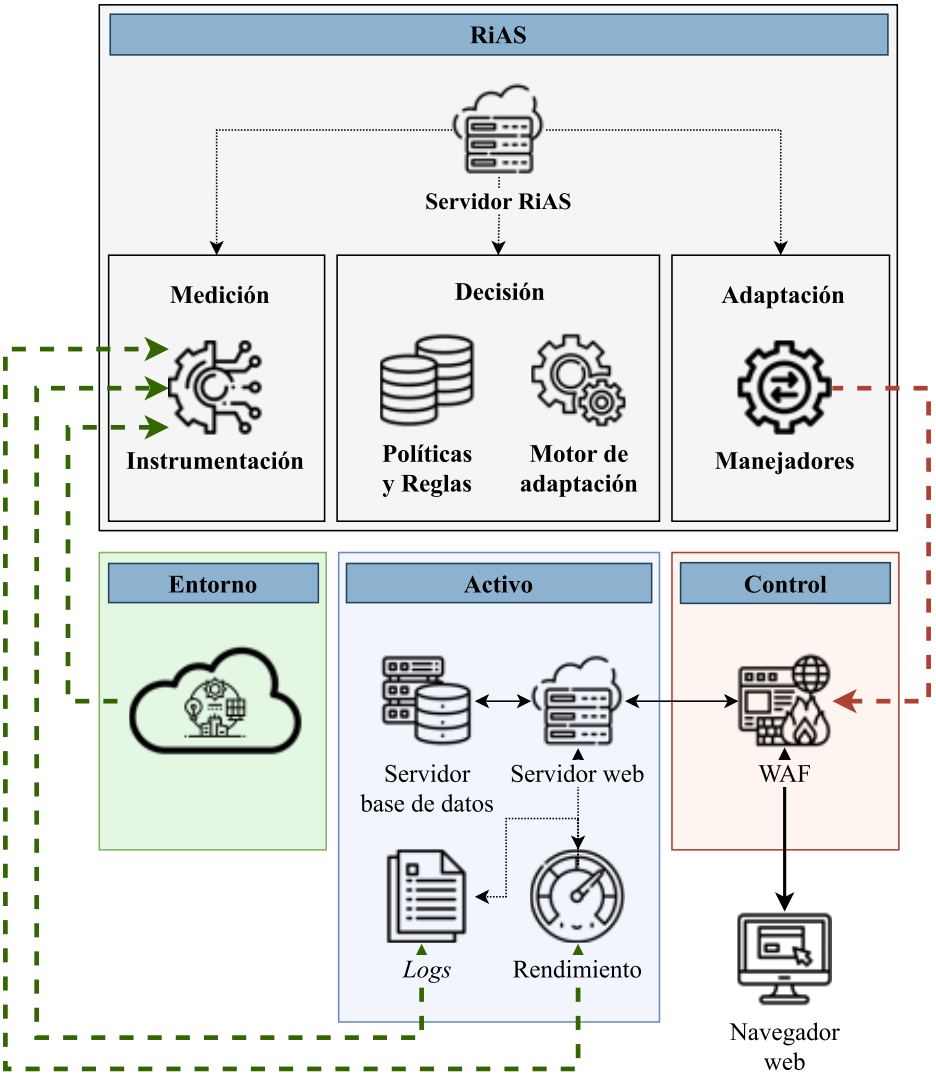

Figura 4.1: Arquitectura propuesta para RiAS en el caso de uso 1.

En la figura 4.1 puede encontrarse la arquitectura propuesta para este primer caso de uso, donde se consigue la adaptación de una capacidad de protección. En ella se exponen los diferentes elementos y componentes que la conforman, entre los que cabe destacar el entorno, los activos, el control de seguridad y RiAS —desplegado de forma centralizada—.

4.1.3. Prototipo

En este caso de uso en particular, se persigue la protección de un activo web específico. Se trata de un portal que exhibe productos provenientes de

diferentes tiendas online, conformando un amplio catálogo con enlaces de afiliados. Los administradores de la plataforma pueden registrar distintos artículos, acompañados de información que abarca aspectos como el título, la descripción, el precio, las características y varias fotografías, entre otros; asimismo, es posible proporcionar una valoración personal de los productos, en la que se destacan tanto sus virtudes como sus defectos. Los usuarios que visitan el catálogo, por su parte, pueden adquirir los artículos a través de enlaces de afiliados que los direccionan a las tiendas donde se venden. Cada una de estas redirecciones, que se realiza a través de un vínculo específico, se traduce en una pequeña ganancia económica para los propietarios de la web.

La aplicación que se desea proteger ha sido desarrollada utilizando tecnologías tales como Vue.js [193], NodeJS [194] y MySQL [195]. Se encuentra desplegada en dos servidores, ambos ejecutando el sistema operativo Ubuntu Server 20.04 y dotados de 2 núcleos virtuales, 2 Gigabytes de memoria RAM (*Random Access Memory*) y 80 Gigabytes de almacenamiento en disco duro. El primero cuenta con el servidor web Nginx [196], siendo responsable de alojar tanto el *frontend* como el *backend* de la aplicación y actuando como entrada del usuario a la misma. El segundo servidor, por su parte, tiene desplegado MySQL Community Server [197], encargándose de almacenar toda la información relacionada con los diferentes productos que se muestran en el catálogo de afiliación, entre otras. Por su parte, RiAS se hospeda en un único servidor con sistema operativo Ubuntu Server 20.04, provisto de 2 núcleos virtuales y 4 Gigabytes de memoria RAM.

El inventario de activos objeto de protección se compone de tres páginas principales: una lista que contiene la totalidad de los productos existentes en la base de datos, una página que muestra información detallada de cada uno de ellos y una sección de búsqueda destinada a facilitar el descubrimiento de artículos del catálogo. El listado de productos se encuentra disponible

en `https://example.com/products` y ha sido diseñado con una función de paginación que tiene por finalidad evitar la sobrecarga de la red y el bloqueo del navegador del usuario. La información detallada relativa a cada artículo se obtiene a través de `https://example.com/product/ID`, en la que cada uno de los identificadores de producto reemplaza la cadena «ID». Finalmente, la búsqueda se localiza en `https://example.com/search`, cuyos parámetros se transmiten mediante una solicitud de tipo POST.

Para la protección de la aplicación web, se ha optado por el uso del WAF ModSecurity [198]. Se trata de una herramienta de código abierto que proporciona monitorización en tiempo real, registro y filtrado de solicitudes HTTP basándose en reglas definidas por el usuario y escritas en un lenguaje de configuración denominado «SecRules». La elección de ModSecurity se debe a la amplia documentación existente, su facilidad de uso y flexibilidad, así como a su capacidad de integración con la aplicación que necesita ser protegida y el servidor web que la aloja. Se utiliza la versión 3 de ModSecurity, que incluye un motor independiente construido desde cero en C++ [199] y es integrable con Nginx, aprovechando la detección basada en firmas. El WAF se instala y configura en el mismo servidor que alberga el *backend* y el *frontend*.

4.1.3.1. Medición

El WAF adaptativo —el WAF tradicional junto al modelo RiAS— requiere una medición adecuada del riesgo para tomar decisiones de adaptación oportunas y efectivas.

Para llevar a cabo las mediciones, se pueden aplicar distintos enfoques, extrayéndose la información de los activos que requieren protección —la aplicación web—, el control —el WAF— o el entorno. Entre las fuentes de datos útiles para este caso de uso se encuentran los *logs* (de la aplicación,

servidores, WAF, etc.), los analizadores de tráfico, los IDPSs (*Intrusion Detection and Prevention Systems*) o SIEMs, las soluciones anti-*malware*, los sistemas de protección de *end-points*, las fuentes abiertas (redes sociales, datos gubernamentales, prensa e información compartida por otras organizaciones) o los productos de inteligencia y descubrimiento de amenazas.

Antes de abordar las definiciones concretas utilizadas en este escenario de aplicación, es útil presentar algunas medidas específicas en el contexto de los WAFs, tales como el número de peticiones HTTP recibidas, los errores HTTP (404, 500, etc.), las peticiones a URIs (*Uniform Resource Identifiers*) malformadas o inesperadas, las solicitudes desde direcciones IP bloqueadas o no permitidas, las consultas a la base de datos, la cantidad de funciones de actualización y eliminación ejecutadas en la base de datos, el número de usuarios conectados simultáneamente o la cantidad de ataques DDoS, todo ello medido en un intervalo de tiempo. Pero también otras como el volumen de interacciones en redes sociales sobre un tema específico o el nivel de alerta terrorista en un país en un momento determinado.

En el presente caso de uso, las medidas se obtienen de cuatro fuentes de datos diferentes, relacionadas con el activo y el entorno.

- En primer lugar, se recopilan las peticiones HTTP de tipo 400 (medida interna). Esta información se extrae de los registros o *logs* del servidor Nginx y se envía a la capa de «Medición» de RiAS. Las métricas relacionadas, que consisten en un valor numérico que indica la cantidad de nuevas peticiones HTTP de tipo 400 en el último intervalo de tiempo, se construyen calculando la media de este tipo de peticiones en el último minuto y en los últimos treinta minutos. Con el objetivo de obtener esta información en tiempo real, se ha desarrollado un *script* en Python [200] que lee la información del *log* y la envía por medio de HTTP al

servidor que hospeda RiAS.

- Otra medición necesaria es el consumo de recursos en el servidor web, es decir, el porcentaje de CPU (*Central Processing Unit*) y memoria RAM consumidos (medida interna). La información se extrae del propio servidor, obteniéndose dos valores numéricos correspondientes a dichos recursos que, después, son transformadas a dos métricas con el cálculo del promedio de consumo durante la última hora. Para realizar esta tarea, se ejecuta un *script* en Python que realiza mediciones y envía la información por medio de HTTP a RiAS.

- En tercer lugar, se toman los *tweets* que contienen una URL del catálogo de productos (medición externa). Para ello, se emplea un *script* que utiliza la API de Twitter [201] y analiza la información en tiempo real en busca de *tweets* que contengan un enlace perteneciente al sitio web protegido. La métrica calculada a partir de estos datos consiste en un valor numérico que corresponde a la cantidad de nuevas apariciones en Twitter por hora.

- Finalmente, el servidor web envía información con respecto al estado de funcionamiento del WAF. Para ello, se trasladan diferentes valores según si la restricción de peticiones por país está activa o no (el valor 0 en caso negativo, 1 en caso afirmativo); y si está trabajando con una lista de permitidos o con una de denegados (0 en caso de denegados, 1 en caso de permitidos). Esta medición o evento en sí no requiere ningún cálculo, por lo que se considera una métrica en sí misma. La recuperación de esta información se realiza mediante dos *scripts* desarrollados en Python, que envían los datos por medio de HTTP.

El código de estado de respuesta HTTP 400, que indica «Solicitud incorrecta» (en inglés *Bad Request*), se produce cuando el servidor no puede procesar

la solicitud enviada por el cliente debido a una sintaxis no válida. Este código puede ser un indicio de un posible ataque de inyección SQL (*Structured Query Language*) o de un intento de extracción de datos de la aplicación web por parte de un atacante. Estas técnicas, al ser empleadas por adversarios, suelen generar numerosos errores, por lo que se considera un indicador relevante para este caso de uso. Por otro lado, el número de URLs enlazadas desde Twitter puede ser utilizado como un indicador de riesgo de un ataque DDoS —voluntario o involuntario—, el cual se caracteriza por un elevado consumo de recursos en el servidor web. Un aumento de la demanda por parte de usuarios legítimos e ilegítimos puede provocar una caída del servicio si no se actúa con rapidez.

Por tanto, la capa «Medición» recibe cinco variables de entrada: las peticiones HTTP de tipo 400, el consumo de recursos (CPU y RAM), los *tweets* que contienen una URL del catálogo de productos y el estado de funcionamiento del WAF en ambos modos (con restricción por país y con lista de denegados/permitidos). Estos datos se obtienen mediante los instrumentos y se almacenan en una base de datos MySQL que se encuentra desplegada en el mismo servidor y que forma parte de la herramienta de adaptación. Después, en la misma capa y a partir de esta información, se elaboran las métricas pertinentes que son enviadas, cuando corresponde, a la siguiente capa —la de «Decisión»— para que, si lo estima oportuno, tras valorar los desencadenantes de las políticas, ejecute las adaptaciones adecuadas.

4.1.3.2. Decisión

Cada vez que la capa de «Decisión» recibe una métrica o un disparador, procede a verificar si esta se utiliza en el desencadenante de alguna política o coincide con el disparador de una regla de tipo «controlada por eventos».

En caso afirmativo, comprueba si los valores se encuentran dentro de los umbrales preestablecidos; de esta forma, se determina si es necesario continuar evaluando la política/regla o no.

En este sentido, se han determinado los siguientes umbrales en relación con los desencadenantes/disparadores y las métricas recibidas:

- Las solicitudes HTTP de tipo 400 que superan las 50 peticiones por minuto, activan el desencadenante «HIGH-HTTP_400_Cod». Por el contrario, cuando este valor es inferior a 50 durante más de 30 periodos de medición —equivalentes a más de 30 minutos—, se activa «LOW_HTTP_400_Cod».

- El número de URLs enlazadas desde Twitter inicia un desencadenante si se detectan más de 100 publicaciones que contienen enlaces al catálogo de productos por hora. Para identificarlo, se ha asignado el nombre de «HIGH-URL_on_Tweets».

- Si la media de utilización de CPU y memoria RAM en la última hora se encuentra por debajo del 55 %, se activa el desencadenante «LOW-WS_Consumption».

- El estado de funcionamiento del WAF se asocia con cuatro desencadenantes, según el modo de operación, que se evalúan cuando este cambia. El primero, «ON-WAF-ALLOWLIST», se cumple cuando el WAF está funcionando en modo lista de permitidos, mientras que «OFF-WAF-ALLOWLIST» lo hace cuando ocurre lo contrario. El desencadenante «ON-WAF-COUNTRY» es verdadero cuando el WAF tiene activadas las restricciones por país, mientras que «OFF-WAF-COUNTRY» indica lo opuesto.

De acuerdo con la definición del modelo, si se cumplen todos los antecedentes establecidos en las políticas —los desencadenantes—, posteriormente, se evalúan las reglas asociadas como consecuentes. Similar ocurre con los disparadores especificados en las reglas de tipo «controlada por eventos», que, al recibirse o cumplirse, y si dichas reglas se encuentran a la espera de aplicación, ejecutan los controles asociados a ellas utilizando un artefacto para la comunicación entre RiAS y el control (en este caso, el WAF). Por lo tanto, para comprender el proceso de toma de decisiones, es fundamental conocer la estructura y el comportamiento de las políticas y reglas definidas en este caso de uso.

- **Política 1**. Su evaluación comienza si se cumplen alguna de las condiciones siguientes, traducidas en desencadenantes:

  1. el número de peticiones HTTP de tipo 400 supera el umbral establecido (desencadenante «`HIGH-HTTP_400_Cod`») y el WAF no está trabajando en modo lista de permitidos («`OFF-WAF-ALLOWLIST`»); o
  2. el número de peticiones HTTP de tipo 400 cae por debajo del umbral establecido («`LOW_HTTP_400_Cod`») y el WAF está trabajando en modo lista de permitidos («`ON-WAF-ALLOWLIST`»).

  El cumplimiento de la primera condición resulta en la aplicación de la «Regla 1», mientras que el de la segunda en la de la «Regla 2».

  - **Regla 1**. Se evalúa automáticamente cuando se cumple uno de los desencadenantes que activa la política correspondiente (la superación del umbral para peticiones HTTP de tipo 400, «`HIGH-HTTP_400_Cod`»). Al aplicarse, ejecuta la acción (o *plugin*) «`waf-allowlist.py`» en el WAF, utilizando el artefacto «`ssh-connection.js`» para establecer la conexión entre este y RiAS.

Tabla 4.1: Políticas 1 y 2 del caso de uso 1 escritas en formato JSON.

```
[ {
    "name": "Política 1",
    "conditions": [{
        "antecedent": [
            "HIGH-HTTP_400_Cod",
            "OFF-WAF-ALLOWLIST"
        ],
        "consequent": ["Regla 1"]
    },{
        "antecedent": [
            "LOW-HTTP_400_Cod",
            "ON-WAF-ALLOWLIST"
        ],
        "consequent": ["Regla 2"]
    }]
},{
    "name": "Política 2",
    "conditions": [{
        "antecedent": [
            "HIGH-URL_on_Tweets",
            "OFF-WAF-COUNTRY"
        ],
        "consequent": ["Regla 3"]
    },{
        "antecedent": [
            "LOW-WS_Consumption",
            "ON-WAF-COUNTRY"
        ],
        "consequent": ["Regla 4"]
    }]
}]
```

  - **Regla 2**. En este caso, la regla espera hasta la hora en punto para ejecutarse (atributo «*Timing*», establecido por el Decisor). Una vez llegado el momento, se aplica la acción (o *plugin*) «`waf-normal.py`» utilizando el artefacto «`ssh-connection.js`» para la comunicación con el WAF.

- **Política 2**. Su evaluación comienza si se cumple una de las siguientes condiciones, correspondientes a una combinación de desencadenantes calculados a partir de las métricas enviadas por la capa de «Medición»:

1. se detecta que el número de *tweets* con URLs del catálogo de productos supera el umbral establecido (desencadenante «`HIGH-URL_on_Tweets`») y el WAF no aplica filtrado geográfico («`OFF-WAF-COUNTRY`»); o
2. el consumo de recursos en el servidor web está por debajo del límite predefinido («`LOW-WS_Consumption`») y, además, el WAF trabaja con restricciones de acceso por país («`ON-WAF-COUNTRY`»).

Como resultado de la primera condición, se aplica la «Regla 3» y, en consecuencia de la segunda, la «Regla 4».

- **Regla 3**. Se evalúa cuando se cumple uno de los desencadenantes que activa la política relacionada (superación del umbral de número de *tweets* que contienen URLs al catálogo, «`HIGH-URL_on_Tweets`»), por lo que lo hace de forma inmediata. Esta regla ejecuta la acción (o *plugin*) «`waf-restricted.py`», utilizando el artefacto «`ssh-connection.js`» para la comunicación entre RiAS y el control de seguridad.
- **Regla 4**. Se analiza cuando el servidor web ha mantenido un consumo medio de CPU y RAM inferior al 55 % en la última hora (desencadenante «`LOW-WS_Consumption`»). Si se cumple esta condición, ejecuta la acción (o *plugin*) «`waf-not-restricted.py`», utilizando el artefacto «`ssh-connection.js`» para establecer la comunicación.

La tabla 4.1 presenta las dos políticas descritas en formato JSON, cada una compuesta por un nombre y diferentes condiciones (antecedentes o desencadenantes y consecuentes o reglas relacionadas). Por su parte, la tabla 4.2 detalla las cuatro reglas propuestas, cada una identificada por un nombre, una especificación temporal y los controles necesarios. En este prototipo, la

Tabla 4.2: Reglas 1, 2, 3 y 4 del caso de uso 1 escritas en formato JSON.


```
[ {
    "name": "Regla 1",
    "timing": {
        "period": null,
        "on-demand": false,
        "trigger": "HIGH-HTTP_400_Cod",
    },
    "controls": [ {
        "action": "waf-allowlist.py",
        "artefact": "ssh-connection.js",
    }],
}, {
    "name": "Regla 2",
    "timing": {
        "period": "0 * * * *",
        "on-demand": false,
        "trigger": null
       },
    "controls": [ {
        "action": "waf-normal.py",
        "artefact": "ssh-connection.js",
    }],
}, {
    "name": "Regla 3",
    "timing": {
        "period": null,
        "on-demand": false,
        "trigger": "HIGH-URL_on_Tweets"
    },
    "controls": [ {
        "action": "waf-count-restricted.py",
        "artefact": "ssh-connection.js",
    }],
}, {
    "name": "Regla 4",
    "timing": {
        "period": null,
        "on-demand": false,
        "trigger": "LOW-WS_Consumption"
    },
    "controls": [ {
        "action": "waf-not-restricted.py",
        "artefact": "ssh-connection.js",
    }],
}]
```

«Regla 1», «Regla 3» y «Regla 4» son de tipo «controladas por eventos». La «Regla 2» es periódica y, en consecuencia, dirigida por un temporizador o *cron* que se expresa con cinco dígitos: minuto, hora, día, mes, día de la semana (el «*» representa cualquier valor, mientras que el «0» indica que solo se ejecuta en horas en punto). Las cuatro reglas utilizan el mismo fragmento de código como artefacto, permitiendo la conexión a través de SSH entre RiAS y el WAF; además, cada una de ellas realiza una acción diferente gracias a los *plugin*, que se presentan en la siguiente sección.

4.1.3.3. Adaptación

Existen diversos ejemplos de adaptaciones que podrían ser útiles para este dominio de aplicación —dependiendo de las necesidades de cada caso concreto—, tales como la activación o desactivación del WAF, la adición o eliminación de reglas, la selección de listas de permitidos o bloqueados, la migración de un WAF local a una solución como servicio (o viceversa) o la adición de una capa anti-DDoS, entre otras.

Para lograr las adaptaciones, definidas mediante reglas y políticas, se requieren diferentes acciones específicas, traducidas a *scripts* o fragmentos de código, encargados de realizar las modificaciones pertinentes. En el caso de uso expuesto, estas acciones se aplican mediante los siguientes elementos:

- **waf-allowlist.py**. Este fragmento de código, desarrollado en el lenguaje de programación Python, se asocia como acción a la «Regla 1». Al ejecutarse, realiza las siguientes operaciones: (1) consulta la base de datos del catálogo para extraer todos los identificadores de productos existentes; (2) modifica el fichero de configuración del WAF activando su funcionamiento mediante una lista de permitidos que incluye, además

de las URLs estáticas, todas las formadas a partir de los identificadores de producto previamente extraídos de la base de datos; (3) reinicia el WAF para aplicar los cambios.

- **waf-normal.py**. Este otro *script*, también desarrollado en Python, se asocia a la «Regla 2». A su ejecución, realiza las siguientes acciones: (1) modifica el fichero de configuración del WAF desactivando su funcionamiento mediante lista de permitidos y eliminando, si las hubiera, todas las reglas asociadas (las creadas mediante «`waf-allowlist.py`»); (2) reinicia el servicio del WAF para que los cambios surtan efecto.

- **waf-count-restricted.py**. El tercer fragmento de código se asocia a la «Regla 3» y está desarrollado en Python. Es el encargado de: (1) modificar el fichero de configuración del WAF para restringir el acceso a la aplicación web sólo desde algunos países concretos; (2) reiniciar el servicio del WAF para que los cambios se apliquen.

- **waf-not-restricted.py**. Este último *script*, también escrito en Python, se asocia a la «Regla 4», y realiza las siguientes acciones: (1) modificar el fichero de configuración del WAF, desactivando la restricción de acceso por países; (2) reiniciar el servicio del WAF para que los cambios se apliquen correctamente.

Por otro lado, para que la acción pueda realizarse, se requieren diferentes artefactos —en este caso únicamente uno— capaces de establecer la conexión entre RiAS y el WAF o el servidor donde este se encuentra desplegado. El artefacto necesario para el caso de uso expuesto se detalla a continuación:

- **ssh-connection.js**. Escrito en NodeJS, se ejecuta en la capa de adaptación de RiAS y se encarga de establecer la conexión vía SSH entre este y el servidor web donde se encuentra desplegado el WAF. A través de

él, se envían las diferentes solicitudes para la ejecución de las acciones anteriormente descritas.

Es importante destacar que los fragmentos de código presentados forman parte de la estrategia propuesta en el caso de uso en cuestión. Cada uno de ellos desempeña una tarea específica para realizar la adaptación del WAF en función de la regla que se activa.

De acuerdo con los roles de RiAS y teniendo en cuenta las responsabilidades dentro del entorno de este caso de uso antes de la utilización del modelo de Seguridad Adaptativa, el administrador de seguridad es capaz de asumir el papel de Decisor. Su tarea consiste en determinar que eventos y métricas pueden ser útiles para detectar la necesidad de adaptación, así como definir las reglas de adaptación necesarias, por lo que su conocimiento en materia de seguridad y de la infraestructura en general lo hacen idóneo para aceptar este rol. Por otro lado, por su cercanía y conocimiento del funcionamiento de la aplicación, el administrador del portal web encaja en el papel de Administrador de políticas de RiAS, siendo responsable de definir las políticas necesarias para responder a los eventos y mediciones. Por último, el rol de Propietario del control es asumido por el administrador del WAF, quien normalmente gestiona este control de seguridad, siendo capaz de crear los manejadores necesarios para su modificación.

4.1.4. Resultados experimentales

Con el fin de validar y evaluar el primer prototipo de RiAS implementado, se realizan diversos experimentos, cuyos resultados se presentan en la tabla 4.3. Estas pruebas protegen toda la aplicación web durante una sema-

Tabla 4.3: Resultados experimentales del caso de uso 1.

	WAF relajado	**WAF restrictivo**	**WAF adaptativo**
Consumo CPU servidor web (%)	16.48	17.01	16.73
Consumo RAM servidor web (%)	40.51	49.98	46.3
Consumo CPU servidor RiAS (%)	-	-	8.21
Consumo RAM servidor RiAS (%)	-	-	24.18
Nº ataques bloqueados	1792	2381	2214
Tasa ataques bloqueados – positivos reales (%)	64	85	79
Nº ataques exitosos	1008	419	586
Nº solicitudes legítimas bloqueadas	134	1567	359
Tasa solicitudes legítimas bloqueadas – falsos positivos (%)	4.8	56	12.8
Trabajo para el administrador del WAF	Configuración del WAF y actualización de firmas	Configuración del WAF y actualización de firmas	Reglas, políticas y actualización de firmas

na, aunque únicamente se exponen los resultados obtenidos sobre la URL `https://example.com/product/ID`. Se consideran dos despliegues estáticos de ModSecurity como línea base para la evaluación del rendimiento, con diferentes niveles de paranoia: (1) una configuración relajada que trabaja con una lista de bloqueados y permite solicitudes desde cualquier país; (2) una configuración restrictiva que opera con una lista de permitidos y solo acepta solicitudes de direcciones IP de países seleccionados. Por otro lado, el WAF adaptativo corresponde al caso de uso, controlando la configuración del WAF mediante RiAS, tal como se describe en secciones anteriores.

Durante la semana de experimentos se simula la actividad de 50 usuarios legítimos de la aplicación, cada uno realizando 400 solicitudes diarias, mez-

cladas con 400 posibles ataques, incluyendo inyecciones SQL y DDoS —para los cuales se han diseñado las reglas y políticas del caso de uso evaluado—. Estos ataques se efectúan utilizando las herramientas OWASP ZAP [202] y Burp Suite [203], con diferentes cargas útiles conocidas.

En la tabla 4.3 se exponen los siguientes resultados obtenidos: el consumo promedio de CPU y RAM debido al uso del WAF en el servidor web, así como estos datos en relación con la ejecución de RiAS; el número de ataques bloqueados y exitosos durante la semana, así como la cantidad de solicitudes legítimas bloqueadas; y el tipo de tarea que el administrador del WAF tiene que realizar antes de los experimentos.

Los resultados alcanzados en el estudio indican que la configuración relajada del WAF consume menos recursos del servidor web; sin embargo, esta permite que un mayor número de ataques tengan éxito en comparación con la configuración restrictiva. Por otro lado, la configuración restrictiva consume significativamente más recursos de CPU y RAM. En contraposición, el enfoque adaptativo propuesto en este caso de uso logra bloquear una cantidad de ataques casi idéntica a la configuración restrictiva, con cifras ligeramente inferiores debido a que la adaptación del WAF de la configuración relajada a la restrictiva no se realiza hasta superar el umbral de riesgo establecido. No obstante, las diferencias con la configuración estática más restrictiva son mínimas. Se pueden explorar otras políticas o reglas para acercar aún más los resultados del WAF adaptativo a los del WAF estático con la configuración restrictiva en términos de tasa de ataques bloqueados (del 79 % actual al 85 %).

Es importante destacar que el uso del enfoque adaptativo tiene una ventaja adicional en la mejora de la accesibilidad de la aplicación. Cuando se utiliza la configuración restrictiva de manera continua, es posible que los usuarios

legítimos de ciertos países sean bloqueados por el WAF. Sin embargo, con el WAF adaptativo, la configuración restrictiva solo se aplica cuando es estrictamente necesario (mediante la «Regla 3»), permitiendo que los usuarios puedan utilizar la aplicación sin limitaciones la mayor parte del tiempo.

Asimismo, se mide el tiempo promedio que tarda el WAF adaptativo en modificar su configuración de manera automática, es decir, cuando no requiere de la intervención de un operador humano y se ejecuta al recibir un disparador, como ocurre en la «Regla 1», la «Regla 3» y la «Regla 4» (en la «Regla 2» el cambio es periódico). Se toma en cuenta el tiempo transcurrido desde la evaluación de una política hasta la decisión de que el WAF debe ser modificado y la aplicación correspondiente de la adaptación. El tiempo promedio es de 16.1 segundos, con una desviación estándar de 8.4 segundos. Cabe señalar que el rendimiento en tiempo de ejecución del ModSecurity adaptativo es idéntico al rendimiento del ModSecurity estático, ya que cada petición recibida se analiza en un rango de 1 a 15 milisegundos.

No se dispone de ninguna versión estable de ModSecurity basada en aprendizaje automático, por lo que no es posible incluirlo en las pruebas. No obstante, se puede inferir que el uso de recursos del servidor web, tanto de CPU como de RAM, sería elevado debido a la complejidad de los modelos empleados por estas tecnologías. Además, para el entrenamiento y reentrenamiento de los modelos fuera de línea, se requeriría un servidor adicional. En el caso del WAF adaptativo, también es necesario un servidor extra para el despliegue de RiAS, pero, además de tener un consumo de recursos limitado, puede ser empleado para aplicar Seguridad Adaptativa a otros controles, no solo al WAF.

Por otro lado, en el WAF basado en aprendizaje automático, el trabajo del administrador del WAF o del especialista en ciencia de datos que lo apoye

estaría relacionado con la construcción del conjunto de datos adecuado y el entrenamiento y reentrenamiento de los modelos; en su contra, el WAF adaptativo requiere de un administrador del WAF que gestione el servicio y, por otro lado, un Decisor, un Administrador de políticas y un Propietario del control encargados de RiAS. Sin embargo, esta supervisión no solo se aplica al WAF, sino a todos los controles gestionados mediante el modelo.

Para este supuesto WAF basado en aprendizaje automático, se debe tener en cuenta que la latencia adicional en cada solicitud sería superior a unos pocos milisegundos, debido a la complejidad de los modelos empleados para clasificar las peticiones como legítimas o maliciosas. La pregunta que surge en este caso es cuánto mejorarían las capacidades del WAF para impedir ataques —especialmente los de *zero-day*— que no se pueden bloquear mediante firmas y que podrían evadir la detección.

4.2. CU 2: Adaptación de una capacidad de detección

En ciberseguridad, una correcta detección es vital para identificar y responder rápidamente a los incidentes. La mayoría de las capacidades de detección se centran en descubrir el suceso una vez que ha ocurrido, mediante mecanismos de vigilancia continua y monitorización, lo que permite una respuesta y recuperación oportunas. Sin embargo, debido a la constante evolución del panorama de las amenazas y de los patrones de ataque, la selección, despliegue y configuración de estos mecanismos y procedimientos representa un gran desafío. Además, detectar de forma temprana eventos y anomalías de seguridad ha demostrado ser un reto considerable, ya que los agentes de amenaza pueden permanecer ocultos en infraestructuras comprometidas

durante más de 100 días, siendo el tiempo medio de detección de 21 días en el año 2021 [204]. Por lo tanto, resulta crucial determinar qué necesita ser supervisado, qué anomalías y eventos deben ser buscados y dónde, así como cuándo emitir una alerta.

Con el objetivo de mejorar el rendimiento de las capacidades de detección una vez que se observa un conjunto de Tácticas, Técnicas y Procedimientos (TTPs o *Tactics, Techniques, and Procedures*) dentro de una infraestructura, este caso de uso propone la utilización de RiAS. Con ello, se permite introducir, de forma dinámica, nuevas capacidades de detección o reconfigurar adecuadamente las existentes.

4.2.1. Contexto

En los últimos años, se ha vuelto cada vez más necesario contar con capacidades de detección, debido, en gran parte, al creciente aumento en el número de amenazas y su nivel de sofisticación; pero también gracias a la mayor concienciación de las organizaciones en materia de ciberseguridad. Dichas capacidades permiten identificar y responder con rapidez ante diversas amenazas, incluyendo la monitorización constante de los activos y la detección proactiva, así como la implementación de soluciones y tecnologías de seguridad para mitigar o minimizar el impacto de las amenazas detectadas o de las futuras [205].

Uno de los marcos de trabajo más útiles y ampliamente utilizados para mejorar las capacidades de detección es MITRE ATT&CK (*Adversarial Tactics, Techniques, and Common Knowledge*) [206], que describe las diferentes TTPs utilizadas por los atacantes para comprometer sistemas e información. MITRE ATT&CK proporciona un marco para ayudar a las organizaciones a

identificar y mitigar las amenazas a través de la comprensión de los patrones y comportamientos utilizados por los ciberdelincuentes. Esta matriz es la base de conocimiento más completa y actualizada de tácticas y técnicas adversarias basadas en observaciones del mundo real, mantenida por la corporación MITRE. En este sentido, varias investigaciones han puesto de manifiesto su gran utilidad para mejorar todo tipo de capacidades de detección, obteniéndose buenos resultados a la hora de encontrar y responder ante diversas amenazas y en diferentes entornos [207–213].

Sin embargo, estas soluciones se centran en la monitorización constante de todas las TTPs relacionadas con los ataques conocidos y registrados en MITRE ATT&CK, combinándose, en la mayoría de los casos, con aprendizaje automático o profundo. Esto conlleva un aumento considerable en el consumo de recursos de los activos monitorizados para identificar amenazas, sin tener en cuenta la relación coste-beneficio, apostando por el intento de una detección total y sin atender al riesgo que se está corriendo en cada momento.

Para hacer frente a los desafíos expuestos, se propone un enfoque adaptativo, aplicando RiAS a las capacidades de detección, de tal forma que se reduzcan los costes computacionales que supone mantener detecciones constantes de todas las amenazas conocidas (algo poco viable para la mayor parte de las organizaciones). Además, a diferencia de las tradicionales, la detección adaptativa no requiere una actualización constante de las reglas o firmas que permita identificar los nuevos patrones, yendo esta actualización en línea con la de la matriz de MITRE ATT&CK.

4.2.2. Arquitectura propuesta

En el segundo caso de uso, se valida RiAS utilizándolo para la adaptación de una capacidad de detección. Aprovechando la estrategia de separación de preocupaciones, la detección de ciberataques se adapta a las TTPs previstas. En este sentido, y tal y como ocurre con los controles de seguridad para la protección de activos, RiAS es capaz de hacer frente, simultáneamente, a la adaptación de diferentes capacidades de detección, pero también combinar la adaptación de la detección con la de la protección.

Siguiendo la arquitectura de tres capas de RiAS, el caso de uso escogido cuenta con el módulo de «Medición», el de «Decisión» y el de «Adaptación». La gran diferencia aquí con respecto al caso de uso 1 y al modelo original, está en la necesidad de incluir, además de los presentados durante la definición de RiAS, otros componentes que permiten la adaptación. De esta forma, la capa de «Medición» debe incluir:

1. Un *sandbox* o una herramienta de análisis de *malware*. Con ella, se pretende ejecutar y observar software, artefactos o código potencialmente malicioso (como *malware* o *exploits*) sin afectar a la plataforma en la que se ejecuta. Este elemento devuelve, tras el análisis, un informe en formato de texto con los resultados obtenidos.

2. Un componente capaz de preprocesar los datos resultantes del análisis, de tal forma que se transformen a un formato en el que el siguiente elemento, el modelo de identificación de TTPs, sea capaz de interpretar. Para ello, se ha empleado la propuesta de [214].

3. Un modelo de identificación de TTPs basado en aprendizaje profundo que reciba los datos preprocesados del informe de análisis producido

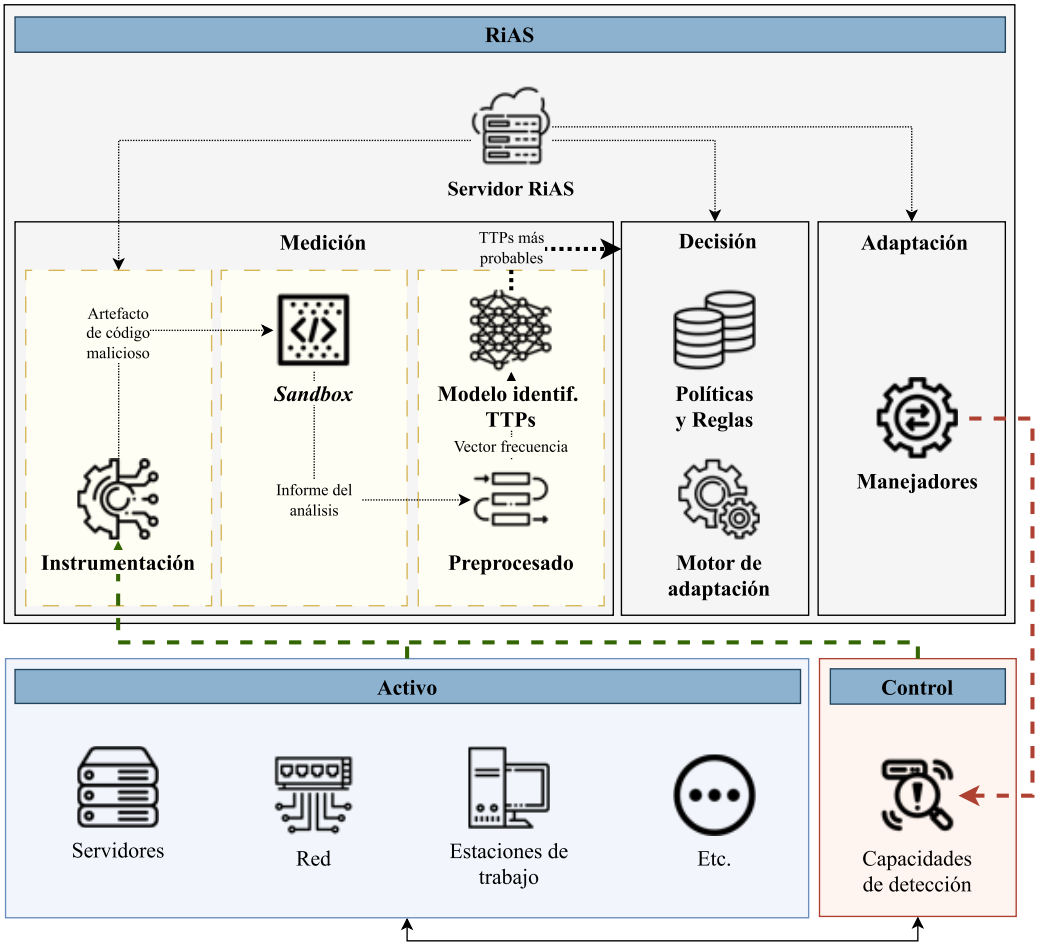

Figura 4.2: Arquitectura propuesta para RiAS en el caso de uso 2.

por el *sandbox* y prediga las TTPs que probablemente están utilizando los adversarios. En este caso de uso se ha utilizado el propuesto en [214].

Además, la arquitectura incluye el activo o activos sobre los que se monitorizan las diferentes amenazas (en este caso servidores, dispositivos de red, estaciones de trabajo, etc.) y las capacidades de detección que, cuando corresponde, sufren las adaptaciones.

De esta manera, cada vez que se materializa una amenaza a través de algún código malicioso y ocurre un incidente de seguridad, el modelo propuesto es capaz de: (1) elaborar un informe con los resultados del análisis de cualquier código asociado al incidente utilizando una herramienta de análisis de

malware o *sandboxing*; (2) predecir las TTPs utilizadas por los atacantes; (3) adaptar los mecanismos de detección de la organización para que en el futuro puedan detectarse nuevos ataques que utilicen las mismas TTPs. Es decir, se persigue que, como mínimo, sea posible detectar ataques que utilicen TTPs ya empleados en el pasado contra la organización.

La figura 4.2 muestra la arquitectura propuesta para este segundo caso de uso, en el que se consigue la adaptación de una capacidad de detección. En ella se incluye tanto RiAS —desplegado de forma centralizada—, como los complementos adicionales necesarios en su capa de «Medición», los activos monitorizados y el control de seguridad.

4.2.3. Prototipo

El objetivo de este caso de uso es adaptar los mecanismos de detección de «Microsoft Defender for Endpoint» (MDE) en un servidor que utiliza el sistema operativo Windows Server. Este está equipado con 2 núcleos virtuales y 4 Gigabytes de memoria RAM.

Por otro lado, el modelo RiAS se despliega en una máquina que utiliza el sistema operativo Ubuntu Server 20.04 y cuenta con 2 núcleos virtuales y 4 Gigabytes de RAM.

La ampliación requerida para este caso de uso, en concreto el *sandbox* capaz de generar un informe de texto a partir del análisis de aplicaciones o fragmentos de código potencialmente maliciosos, se despliega en un servidor que cuenta con 2 núcleos virtuales y 4 Gigabytes de memoria RAM. Además, este se integra directamente con la capa de medición de RiAS.

Por otro lado, se emplea un servidor adicional tanto para el preprocesado del

informe de texto correspondiente al análisis, como para ejecutar el modelo de aprendizaje profundo encargado de la identificación de TTPs. Este servidor también se integra en la capa de medición de RiAS y se comunica con el *sandbox* mediante la API que este último proporciona, facilitando así el intercambio de información entre ellos. Por su parte, cuenta con 8 Gigabytes de RAM y 2 núcleos virtuales, y utiliza el sistema operativo Ubuntu Server 20.04.

4.2.3.1. Medición

En primer lugar, teniendo en cuenta la arquitectura propuesta en la sección anterior, es importante destacar que, en este escenario, el artefacto de software malicioso se inserta manualmente por parte de operadores humanos en la capa de «Medición», a diferencia de lo que normalmente se espera en entornos en los que se utiliza RiAS, que suele recopilar las diferentes mediciones de forma automatizada. Este artefacto es examinado mediante una herramienta de análisis de *malware* o *sandbox* capaz de generar un informe de texto —necesario como entrada, tras su preprocesado, para el modelo de identificación de TTPs—. Aunque esta solución podría ser cualquiera que ofrezca como salida un documento con los resultados del análisis, en este caso concreto se ha incorporado Cuckoo [215], una herramienta de inspección de *malware* automatizada, de código abierto, ampliamente utilizada y que ofrece una API para facilitar la interacción. Esta solución proporciona informes detallados que resumen el comportamiento de artefactos de software potencialmente maliciosos cuando se ejecutan dentro de un entorno realista pero aislado.

Por otro lado, el segundo componente adicional de la capa de «Medición» es el preprocesado de los datos resultantes del análisis [214] —el informe de texto—, de tal forma que: (1) se normalizan, transformando el texto a

minúsculas, eliminando los signos de puntuación, los espacios irrelevantes, las palabras vacías y las no alfanuméricas y *tokenizando* los datos; (2) se lematizan, agrupando las diferentes formas de una misma palabra para reducir el número de *tokens* considerados en el léxico del modelo; (3) se transforman a secuencias numéricas, convirtiendo el texto de entrada en un vector numérico basado en un léxico, donde cada elemento corresponde al identificador numérico del *token* que aparece en el propio texto, utilizando una estrategia de relleno para manejar secuencias de igual longitud. Estas cadenas resultantes del preprocesado, posteriormente, son las utilizadas por el modelo de identificación de TTPs.

El tercer componente necesario se corresponde con el modelo de identificación de TTPs basado en aprendizaje profundo [214]. La entrada a este es un vector que representa la frecuencia de cada palabra del léxico utilizado en el informe (extraído mediante el preprocesado de los datos). La salida es la probabilidad de que el informe se refiera o esté relacionado con alguna de las más de 400 técnicas o sub-técnicas contenidas en la matriz MITRE ATT&CK —concretamente, en el «Enterprise Model»—, utilizada para construir el léxico.

El modelo de identificación de TTPs está basado en una red neuronal recurrente LSTM (*Long Short-Term Memory*) y su objetivo es predecir la categoría de la secuencia de entrada. El conjunto de datos de entrenamiento de este modelo se ha basado en la página web de MITRE ATT&CK y se ha enriquecido utilizando técnicas de raspado o *scraping* para incorporar, además, datos de fuentes externas. El modelo LSTM se ha entrenado para aprender dependencias a largo plazo y para evitar problemas de gradiente como la desaparición o la explotación. El flujo de los datos, así como los distintos procesos aplicados a ellos, se resume en la figura 4.3.

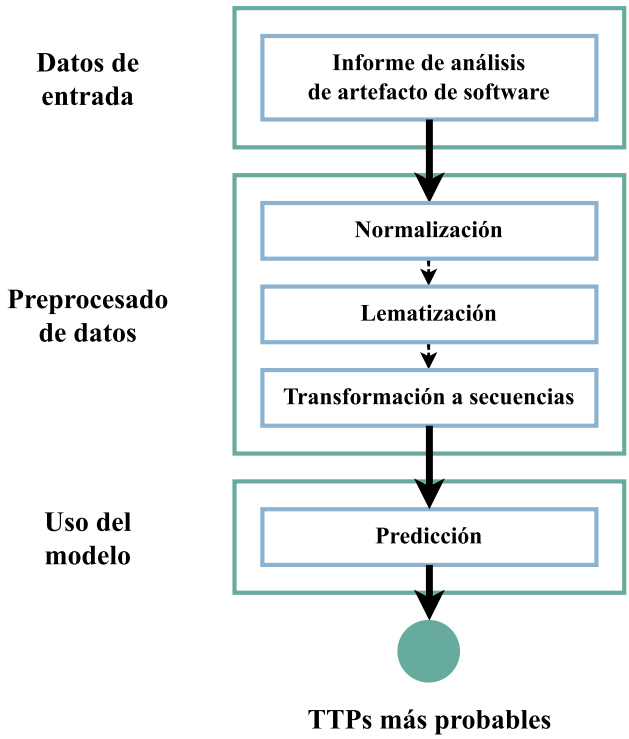

Figura 4.3: Flujo de los datos en la capa de medición del caso de uso 2.

Por tanto, una vez recibido este artefacto de código malicioso en la capa de «Medición» de RiAS:

Por tanto, una vez recibido este artefacto de código malicioso en la capa de «Medición» de RiAS:

- Primero, se genera un informe del análisis en formato JSON mediante el *sandbox* de Cuckoo.
- Después se preprocesa el informe resultante, tal como se muestra en la figura 4.3 y se detalla en párrafos anteriores.
- Posteriormente, el modelo de identificación de TTPs predice las diez TTPs más probables utilizadas por el adversario con este artefacto de software, proporcionando la probabilidad prevista para cada técnica.

Tras esta detección, las métricas —correspondientes a las probabilidades mencionadas—, se envían a la siguiente capa, la de «Decisión», para que

proceda a evaluar si se cumple algún desencadenante. Además, en este envío se incluyen los diferentes disparadores que posteriormente utilizan las reglas, correspondientes a cada una de las detecciones relacionadas con las TTPs predichas —extraídas también de la matriz MITRE ATT&CK—.

4.2.3.2. Decisión

Tras la predicción de TTPs realizada por la capa de «Medición» y su recepción en la de «Decisión», si estas están contenidas en el desencadenante de alguna política, se procede a su evaluación. Los desencadenantes de este caso de uso se han diseñado para considerar técnicas y sub-técnicas previstas con una probabilidad superior al 10 % y, por lo tanto, se cumplen si este umbral se supera.

Además, de igual forma, junto a las métricas necesarias para el cálculo del desencadenante —las 10 TTPs más probables—, se reciben, en forma de disparadores, las detecciones relacionadas con ellas (un disparador por cada capacidad de detección asociada a la técnica/sub-técnica), para que las reglas, de tipo «controladas por eventos», sean ejecutadas de forma inmediata tras la evaluación positiva de la política.

Por tanto, cuando se recibe una medición y tras comprobar que se cumple un desencadenante —correspondiente a los antecedentes—, se evalúa la política asociada a él. En este caso de uso, ha sido necesaria la creación de una política por cada una de las técnicas y sub-técnicas de la matriz MITRE ATT&CK. De esta forma, existe una relación 1 : 1 entre técnicas/sub-técnicas y políticas; aunque tedioso, se ha considerado la mejor manera de conseguir un modo coherente de adaptar las capacidades de detección utilizando RiAS.

Después de la evaluación positiva de las políticas, se valoran las reglas

asociadas —consecuentes—. Estas se accionan mediante los disparadores, ligados a las diferentes capacidades de detección existentes en la matriz MITRE ATT&CK, relacionadas $1 : n$ con las técnicas y sub-técnicas. Esta elección permite reutilizar una misma regla (capacidades de detección) en distintas políticas (técnicas y sub-técnicas utilizadas en el fragmento de código malicioso).

A continuación, se muestra un ejemplo de reglas y políticas asociadas a un módulo del *malware* CHOPSTICK [216] (una conocida familia de software malicioso utilizada por APT28 o Sofacy Group) para Windows.

- **Política 1**. Su evaluación comienza si se detecta, en el artefacto de código analizado, el uso de la técnica T1012 (≪Consulta del registro≫) y esta supera el umbral preestablecido del 10 %, traducido en un desencadenante y correspondiente al antecedente de la política. Una vez se ha comprobado su cumplimiento, se evalúan las siguientes reglas (consecuente):
  - **Regla 1**. Al responder a eventos (concretamente al disparador ≪`CommandExecution`≫), se analiza inmediatamente después de que lo haga la política a la que está asociada, ya que en el momento de su evaluación este se cumple también. Al aplicarse, se encarga de ejecutar la acción (o *plugin*) ≪`WIN_Detection-CommandExecution.ps1`≫ en el servidor Windows, utilizando el artefacto ≪`winrm-connection.js`≫ para establecer la conexión entre este y RiAS.
  - **Regla 2**. Es evaluada inmediatamente después de que lo haga la política a la que se asocia, respondiendo al disparador ≪`OSAPIExecution`≫. Por su parte, al aplicarse, es capaz de ejecutar la acción ≪`WIN_Detection-OSAPIExecution.ps1`≫ en el servidor

Windows, utilizando el artefacto «`winrm-connection.js`» para conseguir una comunicación con RiAS.

- **Regla 3**. Su evaluación comienza tras la de la política relacionada (por haberse recibido ya el disparador «`ProcessCreation`»). Es capaz de ejecutar la acción «`WIN_Detection-ProcessCreation.ps1`» en el servidor Windows mediante el artefacto «`winrm-connection.js`».
- **Regla 4**. Después de evaluar la política y detectar la ocurrencia del disparador «`WRegistryKeyAccess`», se encarga de ejecutar la acción «`WIN_Detection-WRegistryKeyAccess.ps1`» en el servidor Windows, aprovechando el artefacto «`winrm-connection.js`» para establecer la comunicación.

- **Política 2**. Se evalúa cuando la capa de decisión comprueba que se cumple el desencadenante relacionado con la sub-técnica T1056.001 («Captura de entrada: registro de teclas»), correspondiente al antecedente de la política. Contiene las siguientes reglas (consecuente):
  - **Regla 5**. Es analizada tras la evaluación de la política a la que se asocia, respondiendo al disparador «`DriverLoad`». Cuando se aplica, ejecuta la acción «`WIN_Detection-DriverLoad.ps1`» a través del artefacto «`winrm-connection.js`», encargado de establecer la conexión entre el servidor Windows y RiAS.
  - **Regla 2**. Evaluada después de que lo haga la política a la que se vincula, gracias al disparador «`OSAPIExecution`». Es capaz de ejecutar la acción «`WIN_Detection-OSAPIExecution.ps1`» en el servidor Windows, a través del artefacto «`winrm-connection.js`».
  - **Regla 6**. Se analiza tras hacerlo la política a la que acompaña,

Tabla 4.4: Políticas 1 y 2 del caso de uso 2 escritas en formato JSON.

```
[ {
    "name": "Política 1",
    "conditions": [{
        "antecedent": [
            "T1012"
        ],
        "consequent": ["Regla 1", "Regla 2", "Regla 3", "Regla 4"]
    }]
},{
    "name": "Política 2",
    "conditions": [{
        "antecedent": [
            "T1056.001"
        ],
        "consequent": ["Regla 5", "Regla 2", "Regla 6"]
    }]
}]
```

por cumplirse el disparador «`WRegistryKeyModification`». Su acción se corresponde con el *plugin* «`WIN_Detection-WRegistry KeyModification.ps1`», que se ejecuta en el servidor Windows aprovechando la comunicación establecida mediante el artefacto «`winrm-connection.js`».

Estas políticas, redactadas en formato JSON, se muestran en la tabla 4.4, cada una formada por un nombre y diferentes pares antecedente-consecuente. Las reglas de este caso de uso, en el mismo formato, se presentan en las tablas 4.5 y 4.6; cada uno de estos elementos se identifica por un nombre, los controles necesarios y la especificación temporal, que, en todos los casos, se corresponde con un disparador.

Tabla 4.5: Reglas 1, 2, 3 y 4 del caso de uso 2 escritas en formato JSON.

```
[ {
    "name": "Regla 1",
    "timing": {
        "period": null,
        "on-demand": false,
        "trigger": "CommandExecution",
    },
    "controls": [ {
        "action": "WIN_Detection-CommandExecution.ps1",
        "artefact": "winrm-connection.js",
    }],
}, {
    "name": "Regla 2",
    "timing": {
        "period": null,
        "on-demand": false,
        "trigger": "OSAPIExecution",
       },
    "controls": [ {
        "action": "WIN_Detection-OSAPIExecution.ps1",
        "artefact": "winrm-connection.js",
    }],
}, {
    "name": "Regla 3",
    "timing": {
        "period": null,
        "on-demand": false,
        "trigger": "ProcessCreation",
    },
    "controls": [ {
        "action": "WIN_Detection-ProcessCreation.ps1",
        "artefact": "winrm-connection.js",
    }],
}, {
    "name": "Regla 4",
    "timing": {
        "period": null,
        "on-demand": false,
        "trigger": "WRegistryKeyAccess",
    },
    "controls": [ {
        "action": "WIN_Detection-WRegistryKeyAccess.ps1",
        "artefact": "winrm-connection.js",
    }],
}]
```

Tabla 4.6: Reglas 5 y 6 del caso de uso 2 escritas en formato JSON.

```
[{
    "name": "Regla 5",
    "timing": {
        "period": null,
        "on-demand": false,
        "trigger": "DriverLoad",
    },
    "controls": [ {
        "action": "WIN_Detection-DriverLoad.ps1",
        "artefact": "winrm-connection.js",
    }],
}, {
    "name": "Regla 6",
    "timing": {
        "period": null,
        "on-demand": false,
        "trigger": "WRegistryKeyModification",
    },
    "controls": [ {
        "action": "WIN_Detection-WRegistryKeyModification.ps1",
        "artefact": "winrm-connection.js",
    }],
}]
```

4.2.3.3. Adaptación

Con la información proporcionada por la capa de «Medición» y la de «Decisión», la última capa de RiAS, la de «Adaptación», se encarga de realizar las modificaciones que corresponden en cada momento. Para ello, tal y como ocurre en el primer caso de uso, se requieren artefactos —para la comunicación entre RiAS y el Servidor Windows— y acciones —*plugin* encargados de interactuar con la API de «Microsoft Defender for Endpoint» («Microsoft Defender ATP API») [217]—.

Las adaptaciones, en los ejemplos de reglas mostrados, se corresponden con la activación de distintas detecciones: ejecución de comandos, ejecución de la API del sistema operativo, creación de procesos, acceso a la clave del

Tabla 4.7: *Scripts* creados para aplicar las acciones del caso de uso 2 (1/2).

Script: WIN_Detection-CommandExecution.ps1 **Nombre regla MDE**: Ejecución de comandos **Condiciones de detección**: **Evento**: Ejecución de comandos **Acción**: Notificar **Comandos**: cmd.exe, powershell.exe, wscript.exe, cscript.exe **Acciones automáticas**: Poner en cuarentena, Notificar
Script: WIN_Detection-OSAPIExecution.ps1 **Nombre regla MDE**: Ejecución de API del sistema operativo **Condiciones de detección**: **Evento**: Ejecución de API del sistema operativo **Acción**: Bloquear proceso **API**: Cualquier API del sistema operativo **Acciones automáticas**: Poner en cuarentena, Notificar
Script: WIN_Detection-ProcessCreation.ps1 **Nombre regla MDE**: Creación de procesos sospechosos **Condiciones de detección**: **Evento**: Creación de proceso **Acción**: Notificar **Proceso**: Cualquier proceso **Acciones automáticas**: Poner en cuarentena, Notificar

registro de Windows, modificación de la clave del registro de Windows y carga del controlador. Para lograrlas, son necesarios diferentes *scripts*, escritos en PowerShell [218] y encargados de crear reglas para MDE utilizando su API (detalladas en las tablas 4.7 y 4.8), relacionados 1 : 1 con las reglas de RiAS que los contienen.

Por otro lado, se requiere un artefacto para establecer la conexión entre RiAS y el servidor Windows, de tal forma que puedan ejecutarse en él los distintos *plugin* que interaccionan con la ≪Microsoft Defender ATP API≫. Este artefacto necesario para el caso de uso presentado es:

Tabla 4.8: *Scripts* creados para aplicar las acciones del caso de uso 2 (2/2).

Script: WIN_Detection-WRegistryKeyAccess.ps1 **Nombre regla MDE**: Acceso a la clave del registro de Windows **Condiciones de detección**: **Evento**: Acceso a clave del registro de Windows **Acción**: Notificar **Clave**: Cualquier clave del registro de Windows **Acciones automáticas**: Notificar
Script: WIN_Detection-WRegistryKeyModification.ps1 **Nombre regla MDE**: Modificación de la clave del registro de Windows **Condiciones de detección**: **Evento**: Modificación de la clave del registro de Windows **Acción**: Notificar **Clave**: Cualquier clave del registro de Windows **Acciones automáticas**: Poner en cuarentena, Notificar
Script: WIN_Detection-DriverLoad.ps1 **Nombre regla MDE**: Carga de controladores sospechosos **Condiciones de detección**: **Evento**: Carga de controlador **Acción**: Bloquear proceso **Controlador**: No firmado o desconocido **Acciones automáticas**: Poner en cuarentena, Notificar

- **winrm-connection.js**. Escrito en NodeJS, se ejecuta en RiAS y es el encargado de establecer una conexión vía *Windows Remote Management* (WinRM) entre la capa de adaptación del modelo y el servidor Windows donde se requiere modificar la capacidad de detección. A través de él, se envían las diferentes solicitudes para la ejecución de las acciones anteriormente descritas.

De esta forma, cuando se requiere una adaptación, RiAS se conecta al Servidor Windows utilizando el artefacto «`winrm-connection.js`» y ejecuta en él las

acciones que corresponde («WIN_Detection-CommandExecution.ps1», «WIN_Detection-OSAPIExecution.ps1», «WIN_Detection-ProcessCreation.ps1», «WIN_Detection-WRegistryKeyAccess.ps1», «WIN_Detection-WRegistryKeyModification.ps1», «WIN_Detection-DriverLoad.ps1»).

Teniendo en cuenta las responsabilidades previas de cada perfil en el entorno del caso de uso que se describe, el rol Decisor de RiAS se asigna al analista de seguridad, siendo este el encargado de estudiar el *malware* detectado en la infraestructura o los sucesos relacionados y, por ende, quien mejor encaja en el papel. Por su parte, el administrador de seguridad se ajusta al rol de Administrador de políticas, siendo responsable de definir las políticas necesarias para responder a los eventos detectados y coordinar la respuesta a los mismos, cercano a su desempeño habitual cuando se trabaja sin RiAS. Finalmente, el rol de Propietario del control se asigna al administrador de sistemas, quien crea los manejadores necesarios para la adaptación de la herramienta de detección instalada en el servidor.

4.2.4. Resultados experimentales

Para ejemplificar este caso de uso, se ha utilizado un módulo del *malware* CHOPSTICK para Windows encontrado en un servidor después de un incidente de seguridad; sin embargo, si se crean todas las políticas, reglas, y manejadores de RiAS necesarios, la propuesta sería capaz de responder a cualquier *malware* que utilice TTPs registradas en la matriz de MITRE ATT&CK.

En una de las ejecuciones realizadas, tras la inserción del módulo de CHOPSTICK en la capa de «Medición» de RiAS, después del análisis con Cuckoo y

el preprocesado, el modelo de identificación de TTPs expone los siguientes resultados:

1. T1012: Consultar Registro [0.82667]
2. T1056.001: Captura de entrada: Registro de teclas [0.10932]
3. T1113: Captura de pantalla [0.02844]
4. T1105: Transferencia de herramienta de ingreso [0.02346]
5. T1059: Intérprete de secuencias de comandos y comandos [0.01323]
6. T1090.001: Proxy: Proxy interno [0.00754]
7. T1071.001: Protocolo de capa de aplicación: Protocolos web [0.00549]
8. T1573.002: Canal cifrado: Criptografía asimétrica [0.00432]
9. T1518.001: Descubrimiento de software: Descubrimiento de software de seguridad [0.00301]
10. T1497: Virtualización/Evasión de Sandbox [0.00198]

Es importante destacar que CHOPSTICK es uno de los software incluidos en los materiales MITRE ATT&CK (con identificador S0023), y estas diez técnicas predichas por el modelo están incluidas en el conjunto de técnicas identificadas para APT28 al utilizar este *malware*. Esto funciona como una primera validación suave o ligera para los componentes implementados en este prototipo de RiAS, que identifican técnicas que la comunidad ha establecido como las empleadas por los adversarios en esta conocida pieza de código.

En consecuencia, y de acuerdo con lo descrito en secciones anteriores (atendiendo a las políticas y reglas ejemplificadas en las tablas 4.4, 4.5 y 4.6), las

Tabla 4.9: Resultados experimentales del caso de uso 2.

	Tiempo (segundos)	**Consumo CPU** (%)	**Consumo RAM** (%)
Generación del informe	216.3	64.32	83.74
Preprocesado del informe	1.09	5.93	8.24
Identificación de TTPs	2.23	52.20	75.31
Evaluación de reglas y políticas de RiAS	0.93	9.14	23.87
Adaptación de las capacidades de detección	3.51	2.36	4.69
Total	224.06	-	-

capacidades de detección se adaptan para la primera y la segunda técnica de la lista anterior.

Estas adaptaciones, recuperando información de la matriz MITRE ATT&CK, requieren las siguientes capacidades de detección para identificar ataques similares o iguales en el futuro (reflejadas, en forma de manejadores, en las tablas 4.7 y 4.8):

1. T1012 - Consulta del registro. Ejecución de comandos, ejecución de la API del sistema operativo, creación de procesos, acceso a la clave del registro de Windows.

2. T1056.001 - Captura de entrada: registro de teclas. Carga del controlador, ejecución de la API del sistema operativo, modificación de la clave del registro de Windows.

Los resultados derivados de la validación y evaluación de RiAS en el presente caso de uso, donde se emplea para la adaptación de una capacidad de detección, se resumen en la tabla 4.9. Dichos experimentos analizan un módulo del

malware CHOPSTICK para Windows y adaptan la capacidad de detección mediante la creación de nuevas reglas en MDE asociadas a las diversas técnicas y sub-técnicas utilizadas por el *malware* analizado y detectadas, con una probabilidad mayor al 10 %, por el modelo de identificación de TTPs.

Dado que no existen herramientas similares en el mercado, no es posible realizar una comparativa con otras soluciones de naturaleza análoga. Por ende, los beneficios derivados pueden considerarse significativos, particularmente en virtud de los tiempos empleados para la adaptación, calculados desde la inserción del *malware* en el *sandbox* hasta la finalización del proceso. En el presente caso de uso, no se producen retrasos o pausas generados por las reglas de RiAS, ya que todas ellas son de tipo «controladas por eventos», enviándose los disparadores asociados a la par que las métricas necesarias para el cálculo de los desencadenantes de las políticas.

En particular, en la tabla 4.9, se detallan los siguientes aspectos:

- El consumo de recursos (CPU y RAM) del proceso de análisis mediante el *sandbox* Cuckoo, así como el tiempo medio en segundos de este trabajo, encargado de la generación del informe.
- Las necesidades de CPU y RAM para las tareas de preprocesado del informe e identificación de TTPs, además del tiempo medio en segundos de estas labores.
- El consumo de CPU y RAM para la evaluación de reglas y políticas por parte de RiAS, así como el tiempo medio de este proceso.
- El tiempo medio en segundos y el consumo de recursos (CPU y RAM) de la tarea de adaptación en el servidor Windows, contado desde el momento en que RiAS determina la necesidad de modificación y hasta que esta concluye.

Se calcula el tiempo promedio —con base en 20 iteraciones— necesario para la adaptación de las capacidades de detección, teniendo en cuenta tanto el análisis de *malware* con Cuckoo, como el preprocesado del informe generado por la herramienta, la predicción realizada por el modelo de identificación de TTPs, la evaluación de reglas y políticas de RiAS y la adaptación en sí misma. Este tiempo resulta ser de 224.06 segundos (3.73 minutos), con una desviación estándar de 35.9 segundos.

La duración de cada tarea indica que el preprocesado del informe y la evaluación de reglas y políticas de RiAS son los trabajos más rápidos, con 1.09 y 0.93 segundos respectivamente. Por otro lado, la generación del informe y la identificación de TTPs son los procesos que consumen más tiempo y recursos, con valores promedio de 216.3 y 2.23 segundos respectivamente, y consumos de CPU y RAM superiores al 52 % y 75 %; esto se debe al gran esfuerzo que supone la evaluación del artefacto de *malware* en un entorno controlado, así como la operación realizada por el modelo de aprendizaje profundo. En cuanto a la adaptación de las capacidades de detección, se observa un tiempo promedio de 3.51 segundos, con un bajo consumo de recursos (CPU y RAM por debajo del 5 %). En general, se puede concluir que RiAS es capaz de realizar la adaptación de capacidades de detección de manera eficiente y efectiva, con tiempos de respuesta razonables y un bajo consumo de recursos en la adaptación propiamente dicha.

Capítulo 5

Conclusiones

El presente capítulo tiene como objetivo presentar las conclusiones derivadas de la investigación realizada en el marco de esta tesis doctoral. En las siguientes secciones se abordan tanto las conclusiones generales que emergen del trabajo en su conjunto, como aquellas específicas sobre el modelo propuesto, el prototipo desarrollado y su validación. También se incluyen en esta sección las líneas de investigación futura. Se espera que las conclusiones presentadas contribuyan de manera significativa al avance del conocimiento en el ámbito de la Seguridad Adaptativa, sirviendo de guía para investigadores interesados en profundizar en este tema.

5.1. Conclusiones generales

De forma general, la conclusión más significativa que puede extraerse de los resultados de esta tesis es la verificación de la hipótesis de partida. Concretamente, se ha logrado desarrollar un modelo para la Seguridad Adaptativa basado en el riesgo capaz de hacer frente a los problemas presentados en entornos heterogéneos y cambiantes en los que se requiere proteger, de forma

dinámica, activos con características y capacidades muy diferentes entre sí. Dicho modelo surge de: (1) un estudio en profundidad de la necesidad de mejorar la seguridad en dominios de aplicación dinámicos y diversos, en los que, cada día, conviven activos de características y propiedades más dispares; (2) la revisión crítica de las soluciones existentes en el estado del arte, enfocadas en solventar problemas específicos o brindar soporte a un activo o tipo de activo en particular.

Tal y como se observa en el capítulo de estado del arte de esta tesis (≪2. Estado del arte≫), las investigaciones con respecto a la Seguridad Adaptativa, a pesar de haber demostrado en muchos casos ser eficaces y útiles para resolver problemas de seguridad, se centran en dar cobertura a un único dominio o ámbito de aplicación; ofreciendo soluciones especializadas, pero poco flexibles o que no pueden ser aplicadas en entornos heterogéneos donde conviven gran cantidad de activos y controles de seguridad, combinando múltiples características y limitaciones. Con el modelo propuesto se busca abordar y resolver estos problemas de manera eficiente, mediante el uso de una solución capaz de proteger cualquier tipo de activo, recurso o entorno —sin importar su heterogeneidad—; se pretende una solución de Seguridad Adaptativa genérica, donde la variedad no sea un problema.

Por este motivo, se ha considerado imprescindible establecer una nueva arquitectura de Seguridad Adaptativa basada en el riesgo que permita resolver la problemática planteada de manera externa a los controles de seguridad o a los propios activos, sin que sea necesario modificar su diseño original. Este enfoque contribuye a cubrir, con una única solución, la mayoría de las situaciones y escenarios que, hasta el momento, requerían soluciones individuales. La desvinculación de la capacidad de adaptación del resto de elementos de la arquitectura, así como la filosofía de confianza del ≪Motor de adaptación≫ del modelo en el resto de sus componentes y la estandarización

de interfaces de entrada y salida de todos ellos, es una pieza clave para que esta propuesta se pueda adecuar a cualquier sistema, entorno o activo, sin importar su complejidad.

En este sentido, la metodología propuesta en el primer capítulo de esta tesis ha demostrado ser efectiva para alcanzar los objetivos planteados. Es importante destacar que la validación del modelo de Seguridad Adaptativa se ha llevado a cabo apoyándose en casos de uso reales en los que se han resuelto, de manera eficaz y con buenos resultados, dos de los problemas o escenarios en los que RiAS puede ser de gran utilidad para mejorar la seguridad de las organizaciones, sin necesidad de invertir grandes cantidades de recursos o tiempo y adaptándose a contextos muy cambiantes. Asimismo, estas validaciones han permitido demostrar que la capacidad de adaptación propuesta puede ser implementada de manera viable y efectiva desde fuera del control de seguridad encargado de proteger el activo o conjunto de activos, evitando así la necesidad de realizar modificaciones en software/hardware ya existente o de diseñar o crear uno nuevo desde cero.

5.2. Modelo propuesto

En el capítulo 3, se ha presentado un nuevo modelo para la Seguridad Adaptativa basado en el riesgo que aborda las problemáticas identificadas en la revisión bibliográfica realizada en el capítulo 2. Este, permite la adaptación dinámica de diferentes controles de seguridad en entornos y contextos cambiantes en los que conviven dispositivos muy heterogéneos. Además, el modelo actúa como una capacidad de adaptación externa a los controles de seguridad y se ha diseñado en tres capas («Medición», «Decisión» y «Adaptación») para facilitar su despliegue, que puede realizarse en una sola máquina o

en varias, y de forma local o remota. La separación en capas también ha permitido la definición de tres roles para su gestión (Decisor, Administrador de políticas y Propietario del control), cada uno responsable de la creación de los elementos necesarios para la puesta en marcha (instrumentos, reglas, políticas y manejadores) y de supervisar las tareas de cada capa. El modelo se ha diseñado para favorecer la creación y modificación de reglas y políticas, elementos clave de este, de tal forma que sean comprensibles tanto para humanos como para máquinas. Para lograr este objetivo, se ha elegido el lenguaje JSON.

De acuerdo con las distintas categorías de Seguridad Adaptativa que se exponen en el capítulo 2, es posible afirmar que el modelo propuesto se ajusta a la Seguridad Adaptativa basada en riesgos al atender a su desencadenante, a la Seguridad Adaptativa predefinida si se tiene en cuenta el mecanismo de decisión y a ninguna de las englobadas en la categorización con respecto a la adaptación —por tratarse de un modelo genérico y útil para diversas adaptaciones de forma concurrente—.

A continuación, se presentan las conclusiones más relevantes extraídas del trabajo realizado:

- La elección de desarrollar el modelo como una capacidad de adaptación externa a los controles de seguridad, es un punto clave para posibilitar la adaptación dinámica de múltiples controles y activos, sin importar sus características o limitaciones. Además, esta opción permite mantener los controles ya desplegados en la infraestructura, sin necesidad de migrar a nuevas tecnologías o realizar cambios que impliquen una inversión significativa.
- La utilización de MAPE-K como base para la definición del modelo facilita la separación de responsabilidades, lo que garantiza la claridad

y diferenciación de las tareas correspondientes a cada una de las capas de RiAS. Además, la adopción de esta lógica de adaptación, respaldada por diversos trabajos e investigaciones en el campo de la Seguridad Adaptativa, le aporta a RiAS una base sólida y confiable como punto de partida.

- La definición del modelo, separado en tres capas claramente diferenciadas, permite desplegar cada una de ellas en un servidor diferente, independientemente del emplazamiento (local o en la nube). Gracias a esto, se logra una gran escalabilidad y una mayor protección de los datos para instituciones que manejan información crítica o requieren que parte de su sistema se mantenga en las instalaciones de la organización. Por ejemplo, un hospital podría alojar la capa de «Medición» en un servidor local y desplegar el resto en la nube.

- La separación de la gestión de cada una de las capas en roles permite manejar, preparar y configurar cada una de ellas de forma independiente. A pesar de la división, la comunicación entre los roles es fluida y estandarizada, lo que facilita el despliegue y la utilidad de la herramienta.

- La selección adecuada de las métricas que deben ser recopiladas o calculadas es uno de los principales desafíos en el diseño y la configuración de RiAS. Estas deben cumplir con ciertas características para ser útiles y escalables, ya que en ellas se apoya la toma de decisiones; si no se establecen correctamente, su eficacia —y la del modelo— puede verse seriamente comprometida. La adecuación de las métricas a la situación y a los objetivos establecidos resulta esencial; una métrica errónea puede conducir a una representación inadecuada de la realidad, lo que a su vez podría impedir la aplicación oportuna de las adaptaciones planificadas o forzar modificaciones innecesarias.

- La separación de las configuraciones y decisiones en reglas y políticas permite una gran atomización, lo que facilita su modificación de forma independiente y sin necesidad de detener o reiniciar el sistema. Además, esta división aporta sencillez y escalabilidad al modelo, permitiendo que diferentes individuos se encarguen de conjuntos específicos de reglas y políticas, en lugar de tener que depender de un único sujeto para todas las decisiones. Esto es particularmente útil en el manejo de grandes volúmenes de dichos elementos.

- La reutilización de componentes, como los instrumentos de la capa de «Medición» o los manejadores de la de «Adaptación», no solo contribuye a mejorar la escalabilidad del modelo, sino que también permite que organizaciones pequeñas o con limitaciones presupuestarias puedan aprovechar sensores, sondas, *plugin*, APIs, *middleware*, y otros recursos desarrollados por entidades o miembros de la comunidad formada en torno a RiAS.

- Los problemas encontrados durante la adaptación recaen completamente en los artefactos, en lugar de en las acciones, como se esperaba. Esto se debe a la dificultad para conectar de manera remota los diferentes controles de seguridad con el modelo, ya que cada uno de ellos presenta características diferentes, en algunos casos sin una API de entrada, no pudiéndose utilizar ciertos protocolos u otras limitaciones.

- La introducción de una nueva solución de seguridad, como el modelo propuesto, en un ecosistema ya en funcionamiento, puede requerir un gran esfuerzo y una planificación detallada. Es fundamental considerar los riesgos y las implicaciones asociadas y ser consciente de que una configuración inadecuada puede tener graves consecuencias, incluyendo la paralización total de un activo o, incluso, la organización al completo. Por lo tanto, el despliegue del modelo debe ser cuidadosamente gestio-

nado para garantizar una correcta integración y minimizar cualquier interrupción o variación en la funcionalidad de las soluciones existentes.

La definición inicial de este modelo, así como sus características, los detalles más relevantes y su validación, han sido publicados en una revista internacional de reconocido prestigio, *Computers & Security* [219]. Esta publicación, además de las observaciones y comentarios realizados por los revisores expertos en la materia, ha contribuido a la mejora del modelo presentado, finalmente, en esta tesis.

5.3. Prototipo y validación

El modelo descrito en el capítulo 3 se ha implementado y validado a través de dos casos de uso. El primero de ellos tiene como objetivo demostrar los beneficios y resultados obtenidos a partir de la utilización de RiAS para la adaptación de una capacidad de protección —un WAF— destinada a salvaguardar una aplicación web, tal y como se detalla en «4.1. CU 1: Adaptación de una capacidad de protección». El segundo caso de uso, por su parte, se ha diseñado para adaptar, de forma dinámica, una capacidad de detección, concretamente, una herramienta encargada de identificar amenazas en un servidor, como se describe en «4.2. CU 2: Adaptación de una capacidad de detección».

La principal conclusión en relación con el prototipo y su validación es que RiAS demuestra la capacidad de incorporar dinamismo a los controles de seguridad con buenos resultados, tanto en términos de protección como de detección.

De forma general, quedan patentes los beneficios de involucrar a todos los roles

afectados por los cambios en los controles de seguridad durante la planificación y el diseño de las políticas y reglas, así como a la hora de desarrollar los diferentes instrumentos y manejadores. Por ejemplo, los responsables de la gestión del WAF o los analistas de seguridad a cargo de los mecanismos de detección de amenazas, son los más indicados para aconsejar y ayudar a los Decisores, Administradores de políticas y Propietarios del control en la creación de los complementos necesarios para el correcto funcionamiento de RiAS.

Por otro lado, la utilización de RiAS para la adaptación dinámica de controles de seguridad supone una reducción de recursos computacionales consumidos frente a los controles que actúan con configuraciones estáticas en altos niveles de paranoia. Sin embargo, es necesario contar con al menos un servidor más para su despliegue.

Al ser un modelo basado en el riesgo, RiAS permite garantizar la seguridad acorde con este, según el que se está corriendo en un determinado momento, ayudando a mejorar el equilibrio entre seguridad, rendimiento e inversión.

Adicionalmente, se demuestra que RiAS permite la adaptación concurrente de diferentes controles de seguridad —pudiendo ser para la protección y para la detección simultáneamente—, monitorizando cada uno de los contextos relacionados con el control y adaptando según los cambios observados en este, sin perjudicar o reducir las capacidades de los restantes.

Específicamente, en cuanto al primer caso de uso sobre el que se valida la propuesta —la adaptación de una capacidad de protección—, se extraen las siguientes conclusiones:

- Las mejoras con respecto al consumo de recursos, la cantidad de ataques bloqueados, la facilidad de configuración y la flexibilidad en la gestión

de riesgos en comparación con los WAFs tradicionales es significativa.

- El enfoque adaptativo no requiere grandes conjuntos de datos ni procesos de modelado complejos, como sí ocurre con los WAFs basados en aprendizaje automático, lo que se traduce en una menor inversión de recursos humanos y económicos cuando se emplea RiAS.
- El uso de RiAS para la adaptación del WAF ofrece una alta tasa de ataques bloqueados, sin perjudicar más de lo estrictamente necesario a los usuarios legítimos de la aplicación web.

Sobre el segundo caso de uso, la adaptación de las capacidades de detección, resultan las siguientes conclusiones:

- La detección constante de todos los peligros conocidos puede ser perjudicial para el rendimiento de los activos monitorizados. Sin embargo, el uso del enfoque adaptativo permite ajustar estas detecciones según el riesgo o acorde con la observación del entorno.
- Queda patente la utilidad de las capacidades de detección adaptativas cuando se aprovechan fragmentos de código maliciosos encontrados en otras organizaciones similares a la protegida. Esto permite adaptar dichas capacidades antes de sufrir un incidente de seguridad que emplee de nuevo las mismas TTPs.
- El uso de RiAS en este caso de uso ofrece un consumo de recursos ajustado, tanto en la máquina o máquinas necesarias para su despliegue, como en aquellos activos monitorizados. En contraposición a esto, son necesarios dos servidores adicionales para el despliegue del *sandbox* y el modelo de identificación de TTPs.

- El uso de políticas y reglas, así como su atomicidad, ayuda a que los cambios en la matriz de MITRE ATT&CK no conlleven una modificación o actualización completa de la capacidad de adaptación, requiriéndose únicamente la adición o variación de aquellas que se hayan incluido o cambiado.

Los casos de uso utilizados para validar el modelo demuestran su viabilidad, con una reducida inversión en términos económicos y de recursos, aportando un gran valor en materia de seguridad para las organizaciones. El primer caso de uso ya ha sido presentado y publicado en el congreso internacional SECRYPT (*International Conference on Security and Cryptography*) [220].

5.4. Líneas de investigación futura

A través del análisis de las conclusiones expuestas en las secciones precedentes, se constata que tanto los objetivos generales como los específicos, definidos en el capítulo 1, han sido resueltos satisfactoriamente. No obstante, la implementación del modelo de Seguridad Adaptativa basado en el riesgo y su correspondiente validación mediante casos de uso han originado diversas áreas de investigación que podrían contribuir a mejorar y complementar el modelo presentado en esta tesis. En este sentido, se identifican varias líneas que merecen ser mencionadas, tales como:

- Explorar la posibilidad de crear conjuntos de métricas base o *frameworks* con el propósito de facilitar la elección de indicadores o parámetros, así como para proveer una visión más amplia al medir el riesgo de un activo, un conjunto de ellos o su contexto. La literatura existente (véase «2. Estado del arte») no favorece la comprensión o mejora de las

métricas, limitándose a especificar aquellas que sus sistemas requieren o incluso, en algunos casos, sin aportar información al respecto.

- Validar el modelo en entornos que empleen dispositivos con limitaciones en cuanto a sus capacidades de cómputo, los protocolos y puertos habilitados o los sistemas operativos, como es el caso de IoT. Realizar dicha validación permitiría demostrar la utilidad del modelo para fortalecer la seguridad en activos que presenten restricciones específicas.

- Estudiar la viabilidad del modelo para su aplicación en capacidades de respuesta. Hasta ahora, los esfuerzos se han centrado en adaptar mecanismos para la protección o la detección, pero podría resultar beneficioso incluir capacidades de adaptación a medidas como la recuperación de datos, los análisis forenses, la identificación de la causa raíz de un problema de seguridad, etc.

- El modelo ha sido diseñado para actuar de manera predefinida, sin embargo, podría ser interesante explorar la posibilidad de adecuarlo para que la toma de decisiones se haga de manera inteligente, apoyándose en técnicas y herramientas novedosas como la Inteligencia Artificial, el aprendizaje profundo o el aprendizaje automático.

- Estudiar la viabilidad de aplicar el modelo para la mejora de la privacidad. Hasta el momento, la mayor parte de los esfuerzos en materia de adaptación se han enfocado en mejorar la seguridad, sin embargo, la Privacidad Adaptativa es un ámbito aún muy poco explorado. La aplicación del modelo para la mejora de la privacidad podría tener un impacto significativo en la protección de la información de los usuarios y en el cumplimiento de leyes y regulaciones relacionadas como el RGPD (Reglamento General de Protección de Datos) y la LOPDGDD (Ley Orgánica de Protección de Datos Personales y Garantía de los Derechos

Digitales).

- Asegurar la adaptación mediante la verificación remota de los diferentes activos o controles de seguridad implicados, con el propósito de garantizar que, antes de la adaptación, se tenga la certeza de que dichos activos o controles no han sido comprometidos. En este sentido, esta tesis ha impulsado la investigación al respecto, dando lugar a dos publicaciones presentadas en congresos de gran relevancia en el ámbito de la ciberseguridad. En la primera, se propone una solución para la atestación remota de dispositivos IoT con *Edge*, fundamentada en la utilización de TPMs (*Trusted Platform Modules*) [221]. En la segunda, se presenta un mecanismo de verificación remota para soluciones de Seguridad Adaptativa, con el propósito de garantizar que los controles de seguridad se encuentran en un estado seguro antes de proceder con cualquier adaptación [222].

Cabe destacar que gran parte de estas líneas de investigación futura ya han sido o están siendo exploradas en proyectos de investigación a nivel nacional o internacional, en nuevas publicaciones enviadas a revistas científicas de alto impacto y en nuevos trabajos [223] y tesis doctorales.

Bibliografía

[1] Didac Gil de la Iglesia and Danny Weyns. Mape-k formal templates to rigorously design behaviors for self-adaptive systems. *ACM Transactions on Autonomous and Adaptive Systems*, 10(3), 2015. doi:10.1145/2724719.

[2] Philip O'Kane, Sakir Sezer, and Domhnall Carlin. Evolution of ransomware. *IET Networks*, 7(5):321–327, 2018. doi:10.1049/iet-net.2017.0207.

[3] Eman Khaleefa and Dhahair Abdulah. Concept and difficulties of advanced persistent threats (apt): Survey. *International Journal of Nonlinear Analysis and Applications*, 13(1):4037–4052, 2022. doi:10.22075/ijnaa.2022.6230.

[4] Scott Steele Buchanan. *Cyber-Attacks to Industrial Control Systems since Stuxnet: A Systematic Review*. PhD thesis, Capitol Technology University, 2022. URL: https://dl.acm.org/doi/book/10.5555/AAI29163646.

[5] Santiago Quintero-Bonilla and Angel Martín del Rey. A new proposal on the advanced persistent threat: A survey. *Applied Sciences*, 10(11), 2020. doi:10.3390/app10113874.

[6] Harun Oz, Ahmet Aris, Albert Levi, and A. Selcuk Uluagac. A survey

on ransomware: Evolution, taxonomy, and defense solutions. *ACM ACM Computing Surveys*, 54(11), 2022. doi:10.1145/3514229.

[7] Matt Tatam, Bharanidharan Shanmugam, Sami Azam, and Krishnan Kannoorpatti. A review of threat modelling approaches for apt-style attacks. *Heliyon*, 7(1), 2021. doi:10.1016/j.heliyon.2021.e05969.

[8] Roaa Al Nafea and Mohammed Amin Almaiah. Cyber security threats in cloud: Literature review. In *2021 International Conference on Information Technology (ICIT)*, pages 779–786, 2021. doi:10.1109/ICIT52682.2021.9491638.

[9] Jianli Pan and Zhicheng Yang. Cybersecurity challenges and opportunities in the new edge computing + iot world. In *Proceedings of the 2018 ACM International Workshop on Security in Software Defined Networks & Network Function Virtualization*, SDN-NFV Sec'18, pages 29–32. Association for Computing Machinery, 2018. doi:10.1145/3180465.3180470.

[10] Yash Shah and Shamik Sengupta. A survey on classification of cyber-attacks on iot and iiot devices. In *2020 11th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, pages 406–413, 2020. doi:10.1109/UEMCON51285.2020.9298138.

[11] Ricardo Jorge Raimundo and Albérico Travassos Rosário. Cybersecurity in the internet of things in industrial management. *Applied Sciences*, 12(3), 2022. doi:10.3390/app12031598.

[12] Danda B. Rawat, Ronald Doku, and Moses Garuba. Cybersecurity in big data era: From securing big data to data-driven security. *IEEE Transactions on Services Computing*, 14(6):2055–2072, 2021. doi:10.1109/TSC.2019.2907247.

[13] Cheerag Kaura, Nidhi Sindhwani, and Alka Chaudhary. Analysing the impact of cyber-threat to ics and scada systems. In *2022 International Mobile and Embedded Technology Conference (MECON)*, pages 466–470, 2022. doi:10.1109/MECON53876.2022.9752425.

[14] Asif Iqbal Kawoosa and Deepak Prashar. A review of cyber securities in smart grid technology. In *2021 2nd International Conference on Computation, Automation and Knowledge Management (ICCAKM)*, pages 151–156, 2021. doi:10.1109/ICCAKM50778.2021.9357698.

[15] Fazel Mohammadi. Emerging challenges in smart grid cybersecurity enhancement: A review. *Energies*, 14(5), 2021. doi:10.3390/en14051380.

[16] Sen Tan, Yanpeng Wu, Peilin Xie, Josep M. Guerrero, Juan C. Vasquez, and Abdullah Abusorrah. New challenges in the design of microgrid systems: Communication networks, cyberattacks, and resilience. *IEEE Electrification Magazine*, 8(4):98–106, 2020. doi:10.1109/MELE.2020.3026496.

[17] Muhammad Usman, Mian Ahmad Jan, Xiangjian He, and Jinjun Chen. A survey on representation learning efforts in cybersecurity domain. *ACM Computing Surveys*, 52(6), 2019. doi:10.1145/3331174.

[18] Abdul Razaque, Fathi Amsaad, Meer Jaro Khan, Salim Hariri, Shujing Chen, Chen Siting, and Xingchen Ji. Survey: Cybersecurity vulnerabilities, attacks and solutions in the medical domain. *IEEE Access*, 7:168774–168797, 2019. doi:10.1109/ACCESS.2019.2950849.

[19] Aastha Verma and Charu Shri. Cyber security: A review of cyber crimes, security challenges and measures to control. *Vision*, 2022. doi:10.1177/09722629221074760.

[20] K. M Rajasekharaiah, Chhaya S Dule, and E Sudarshan. Cyber security challenges and its emerging trends on latest technologies. *IOP Conference Series: Materials Science and Engineering*, 981(2), 2020. doi:10.1088/1757-899X/981/2/022062.

[21] Niels Kubler, Tim Brunner, Konstantin Moser, Norina Braun, Christian Bieri, Ivana Mesié, Maximilian Achakri, Julius Willems, and Dario Akhavan Safa. Communication systems xiv. Technical Report IFI-2021.02, University of Zurich, 2021. URL: https://files.ifi.uzh.ch/CSG/teaching/FS21/IFI_2021_02.pdf.

[22] Sabina Baraković and Jasmina Baraković Husić. *Cyber Security Perspective of Top Future Technologies*, pages 85–98. IOS Press BV, 2022. doi:10.3233/NICSP220020.

[23] Barry Porter, Roberto Rodrigues Filho, and Paul Dean. A survey of methodology in self-adaptive systems research. In *2020 IEEE International Conference on Autonomic Computing and Self-Organizing Systems (ACSOS)*, pages 168–177, 2020. doi:10.1109/ACSOS49614.2020.00039.

[24] Flavius Graur. Dynamic network configuration in the internet of things. In *2017 5th International Symposium on Digital Forensic and Security (ISDFS)*, pages 1–4, 2017. doi:10.1109/ISDFS.2017.7916503.

[25] Mattias T. Gebrie and Habtamu Abie. Risk-based adaptive authentication for internet of things in smart home ehealth. In *Proceedings of the 11th European Conference on Software Architecture: Companion Proceedings*, ECSA '17, pages 102–108. Association for Computing Machinery, 2017. doi:10.1145/3129790.3129801.

[26] Irdin Pekaric, Raffaela Groner, Thomas Witte, Jubril Gbolahan Adigun, Alexander Raschke, Michael Felderer, and Matthias Tichy. A systematic

review on security and safety of self-adaptive systems. *Social Science Research Network*, 2022. doi:10.2139/ssrn.4029617.

[27] Giannis Tziakouris, Rami Bahsoon, and Muhammad Ali Babar. A survey on self-adaptive security for large-scale open environments. *ACM Computing Surveys*, 51(5), 2018. doi:10.1145/3234148.

[28] Reza Montasari, Amin Hosseinian-Far, and Richard Hill. *Policies, Innovative Self-Adaptive Techniques and Understanding Psychology of Cybersecurity to Counter Adversarial Attacks in Network and Cyber Environments*, pages 71–93. Springer International Publishing, 2018. doi:10.1007/978-3-319-97181-0_4.

[29] Aradea Aradea, Iping Supriana, Kridanto Surendro, Husni Mubarok, and Irfan Darmawan. Self-adaptive cybersecurity system. In *Proceedings of the 2018 International Conference on Industrial Enterprise and System Engineering (IcoIESE 2018)*, pages 37–42. Atlantis Press, 2019/03. doi:10.2991/icoiese-18.2019.7.

[30] Terence Wong, Markus Wagner, and Christoph Treude. Self-adaptive systems: A systematic literature review across categories and domains. *Information and Software Technology*, 148, 2022. doi:10.1016/j.infsof.2022.106934.

[31] Danny Weyns, M. Usman Iftikhar, Didac Gil de la Iglesia, and Tanvir Ahmad. A survey of formal methods in self-adaptive systems. In *Proceedings of the Fifth International C∗Conference on Computer Science and Software Engineering*, C3S2E '12, pages 67–79. Association for Computing Machinery, 2012. doi:10.1145/2347583.2347592.

[32] Mazeiar Salehie and Ladan Tahvildari. Self-adaptive software: Landscape and research challenges. *ACM Transactions on Autonomous and Adaptive Systems*, 4(2), 2009. doi:10.1145/1516533.1516538.

[33] Robert Laddaga. Active software. In *Proceedings of the First International Workshop on Self-Adaptive Software*, IWSAS' 2000, pages 11–26, Berlin, Heidelberg, 2000. Springer-Verlag. URL: https://dl.acm.org/doi/10.5555/375094.375105.

[34] Radu Calinescu, Raffaela Mirandola, Diego Perez-Palacin, and Danny Weyns. Understanding uncertainty in self-adaptive systems. In *2020 IEEE International Conference on Autonomic Computing and Self-Organizing Systems (ACSOS)*, pages 242–251, 2020. doi:10.1109/ACSOS49614.2020.00047.

[35] Maryam Sajjad, Aakash Ahmad, Asad Waqar Malik, Ahmed B. Altamimi, and Ibrahim Alseadoon. Classification and mapping of adaptive security for mobile computing. *IEEE Transactions on Emerging Topics in Computing*, 8(3):814–832, 2020. doi:10.1109/TETC.2018.2791459.

[36] Joyce Hoese Addae, Michael Brown, Xu Sun, Dave Towey, and Milena Radenkovic. Measuring attitude towards personal data for adaptive cybersecurity. *Information & Computer Security*, 25(5):560–579, 2017. doi:10.1108/ICS-11-2016-0085.

[37] Sumit Kumar, Ravi Shanker, and Sahil Verma. Context aware dynamic permission model: A retrospect of privacy and security in android system. In *2018 International Conference on Intelligent Circuits and Systems (ICICS)*, pages 324–329, 2018. doi:10.1109/ICICS.2018.00073.

[38] Eric Gyamfi, James Adu Ansere, Mohsin Kamal, Muhammad Tariq, and Anca Jurcut. An adaptive network security system for iot-enabled maritime transportation. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–10, 2022. doi:10.1109/TITS.2022.3159450.

[39] Mehdi Gheisari, Guojun Wang, Wazir Zada Khan, and Christian Fernández-Campusano. A context-aware privacy-preserving method for iot-based smart city using software defined networking. *Computers & Security*, 87, 2019. doi:10.1016/j.cose.2019.02.006.

[40] Xin Fan, Chenlu Li, and Xiaoju Dong. A real-time network security visualization system based on incremental learning (chinavis 2018). *Journal of Visualization*, 22(1):215–229, 2019. doi:10.1007/s12650-018-0525-z.

[41] Youbiao He, Gihan J. Mendis, and Jin Wei. Real-time detection of false data injection attacks in smart grid: A deep learning-based intelligent mechanism. *IEEE Transactions on Smart Grid*, 8(5):2505–2516, 2017. doi:10.1109/TSG.2017.2703842.

[42] Antonio Cuadra and Javier Aracil. Context-aware security framework based on traffic anomaly detection indicator. *Telecommunication Systems*, 65(2):319–330, 2017. doi:10.1007/s11235-016-0233-8.

[43] Ting Liu, Jue Tian, Yuhong Gui, Yang Liu, and Pengfei Liu. Sedea: State estimation-based dynamic encryption and authentication in smart grid. *IEEE Access*, 5:15682–15693, 2017. doi:10.1109/ACCESS.2017.2713440.

[44] Amel Arfaoui, Soumaya Cherkaoui, Ali Kribeche, Sidi Mohammed Senouci, and Mohamed Hamdi. Context-aware adaptive authentication and authorization in internet of things. In *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, pages 1–6, 2019. doi:10.1109/ICC.2019.8761830.

[45] Hassan Loulou, Sebastien Saudrais, Hassan Soubra, and Cherif Larouci. Adapting security policy at runtime for connected autonomous vehicles.

In *2016 IEEE 25th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)*, pages 26–31, 2016. doi:10.1109/WETICE.2016.16.

[46] Sudipta Ghosh and Swaminathan Seetharaman. Mechanism for adaptive and context-aware inter-iot communication. In *2015 IEEE International Conference on Advanced Networks and Telecommuncations Systems (ANTS)*, pages 1–6, 2015. doi:10.1109/ANTS.2015.7413653.

[47] Dipankar Dasgupta, Arunava Roy, and Abhijit Nag. Toward the design of adaptive selection strategies for multi-factor authentication. *Computers & Security*, 63:85–116, 2016. doi:10.1016/j.cose.2016.09.004.

[48] Shohreh Hosseinzadeh, Seppo Virtanen, Natalia Díaz-Rodríguez, and Johan Lilius. A semantic security framework and context-aware role-based access control ontology for smart spaces. In *Proceedings of the International Workshop on Semantic Big Data*, SBD '16. Association for Computing Machinery, 2016. doi:10.1145/2928294.2928300.

[49] Zhan Liu, Riccardo Bonazzi, and Yves Pigneur. Privacy-based adaptive context-aware authentication system for personal mobile devices. *Journal of Mobile Multimedia*, 12(1-2):159–180, 2016. URL: https://dl.acm.org/doi/10.5555/3177177.3177187, doi:10.5555/3177177.3177187.

[50] Simeon Veloudis, Yiannis Verginadis, Ioannis Patiniotakis, Iraklis Paraskakis, and Gregoris Mentzas. Context-aware security models for paas-enabled access control. In *Proceedings of the 6th International Conference on Cloud Computing and Services Science - Volume 1 and 2*, CLOSER 2016, pages 202–212. SCITEPRESS, 2016. doi:10.5220/0005918602020212.

[51] Eric Wang, Tsu-Yang Wu, Chien-Ming Chen, Yuming Ye, Zhujin Zhang, and Futai Zou. Mdpas: Markov decision process based adaptive security for sensors in internet of things. *Advances in Intelligent Systems and Computing*, 329:389–397, 2015. doi:10.1007/978-3-319-12286-1_40.

[52] Yosef Ashibani, Dylan Kauling, and Qusay H. Mahmoud. A context-aware authentication framework for smart homes. In *2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE)*, pages 1–5, 2017. doi:10.1109/CCECE.2017.7946657.

[53] Feng Yao, Suleiman Y. Yerima, BooJoong Kang, and Sakir Sezer. Continuous implicit authentication for mobile devices based on adaptive neuro-fuzzy inference system. In *2017 International Conference on Cyber Security And Protection Of Digital Services (Cyber Security)*, pages 1–7, 2017. doi:10.1109/CyberSecPODS.2017.8074846.

[54] Kamal Benzekki, Abdeslam El Fergougui, and Abdelbaki ElBelrhiti ElAlaoui. A context-aware authentication system for mobile cloud computing. *Procedia Computer Science, Proceedings of the First International Conference on Intelligent Computing in Data Sciences - ICDS2017*, 127:379–387, 2018. doi:10.1016/j.procs.2018.01.135.

[55] Daniel Petrov and Taieb Znati. Context-aware deep learning-driven framework for mitigation of security risks in byod-enabled environments. In *2018 IEEE 4th International Conference on Collaboration and Internet Computing (CIC)*, pages 166–175, 2018. doi:10.1109/CIC.2018.00032.

[56] Carlos Eduardo Da Silva, Thomás Diniz, Nelio Cacho, and Rogério de Lemos. Self-adaptive authorisation in openstack cloud platform. *Journal*

of Internet Services and Applications, 9(1), 2018. doi:10.1186/s13174-018-0090-7.

[57] Jose L. Hernandez Ramos, Jorge Bernal Bernabe, and Antonio F. Skarmeta. Managing context information for adaptive security in iot environments. In *2015 IEEE 29th International Conference on Advanced Information Networking and Applications Workshops*, pages 676–681, 2015. doi:10.1109/WAINA.2015.55.

[58] Aakash Ahmad, Asad Waqar Malik, Abdulrahman Alreshidi, Wilayat Khan, and Maryam Sajjad. Adaptive security for self-protection of mobile computing devices. *Mobile Networks and Applications*, 2019. doi:10.1007/s11036-019-01355-y.

[59] Tidiane Sylla, Mohamed Aymen Chalouf, Francine Krief, and Karim Samaké. Towards a context-aware security and privacy as a service in the internet of things. In Maryline Laurent and Thanassis Giannetsos, editors, *Information Security Theory and Practice*, pages 240–252. Springer International Publishing, 2020. doi:10.1007/978-3-030-41702-4_15.

[60] Evgenia Psarra, Yiannis Verginadis, Ioannis Patiniotakis, Dimitris Apostolou, and Gregoris Mentzas. A context-aware security model for a combination of attribute-based access control and attribute-based encryption in the healthcare domain. In Leonard Barolli, Flora Amato, Francesco Moscato, Tomoya Enokido, and Makoto Takizawa, editors, *Web, Artificial Intelligence and Network Applications*, pages 1133–1142. Springer International Publishing, 2020. URL: https://link.springer.com/chapter/10.1007/978-3-030-44038-1_104, doi:10.1007/978-3-030-44038-1_104.

[61] Bart Gijsen, Ruggero Montalto, Jeffrey Panneman, Federico Falconieri,

Paul Wiper, and Piotr Zuraniewski. Self-healing for cyber-security. In *2021 Sixth International Conference on Fog and Mobile Edge Computing (FMEC)*, pages 1–7, 2021. doi:10.1109/FMEC54266.2021.9732575.

[62] Rozhin Yasaei, Felix Hernandez, and Mohammad Abdullah Al Faruque. Iot-cad: Context-aware adaptive anomaly detection in iot systems through sensor association. In *Proceedings of the 39th International Conference on Computer-Aided Design*, ICCAD '20, New York, NY, USA, 2020. Association for Computing Machinery. doi:10.1145/3400302.3415672.

[63] Lorenzo Fernández Maimó, Ángel Luis Perales Gómez, Félix J. García Clemente, Manuel Gil Pérez, and Gregorio Martínez Pérez. A self-adaptive deep learning-based system for anomaly detection in 5g networks. *IEEE Access*, 6:7700–7712, 2018. doi:10.1109/ACCESS.2018.2803446.

[64] Fang Wang, Weimin Qi, and Tonghui Qian. A dynamic cybersecurity protection method based on software-defined networking for industrial control systems. In *2019 Chinese Automation Congress (CAC)*, pages 1831–1834, 2019. doi:10.1109/CAC48633.2019.8996244.

[65] Santhosh Parampottupadam and Arghir-Nicolae Moldovann. Cloud-based real-time network intrusion detection using deep learning. In *2018 International Conference on Cyber Security and Protection of Digital Services (Cyber Security)*, pages 1–8, 2018. doi:10.1109/CyberSecPODS.2018.8560674.

[66] Christos Constantinides, Stavros Shiaeles, Bogdan Ghita, and Nicholas Kolokotronis. A novel online incremental learning intrusion prevention system. In *2019 10th IFIP International Conference*

on New Technologies, Mobility and Security (NTMS). IEEE, 2019. doi:10.1109/ntms.2019.8763842.

[67] Ryan Heartfield, George Loukas, Anatolij Bezemskij, and Emmanouil Panaousis. Self-configurable cyber-physical intrusion detection for smart homes using reinforcement learning. *IEEE Transactions on Information Forensics and Security*, 16:1720–1735, 2021. doi:10.1109/TIFS.2020.3042049.

[68] Dimitrios Papamartzivanos, Félix Gómez Mármol, and Georgios Kambourakis. Introducing deep learning self-adaptive misuse network intrusion detection systems. *IEEE Access*, 7:13546–13560, 2019. doi:10.1109/ACCESS.2019.2893871.

[69] Pantaleone Nespoli, David Useche Pelaez, Daniel Díaz López, and Félix Gómez Mármol. Cosmos: Collaborative, seamless and adaptive sentinel for the internet of things. *Sensors*, 19(7), 2019. doi:10.3390/s19071492.

[70] Ahmad Mohsin, Sundas Asghar, and Tariq Naeem. Intelligent security cycle: A rule based run time malicious code detection technique for soap messages. In *2016 19th International Multi-Topic Conference (INMIC)*, pages 1–10, 2016. doi:10.1109/INMIC.2016.7840097.

[71] M. Abdelrazek, J. Grundy, and A. Ibrahim. Chapter 5 - adaptive security for software systems. In Ivan Mistrik, Nour Ali, Rick Kazman, John Grundy, and Bradley Schmerl, editors, *Managing Trade-Offs in Adaptable Software Architectures*, pages 99–127. Morgan Kaufmann, 2017. doi:10.1016/B978-0-12-802855-1.00005-8.

[72] Amit Kumar Sikder, Hidayet Aksu, and A. Selcuk Uluagac. 6thsense: A context-aware sensor-based attack detector for smart devices. In

Proceedings of the 26th USENIX Conference on Security Symposium, SEC'17, pages 397–414. USENIX Association, 2017. URL: https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/sikder.

[73] Amit Kumar Sikder, Leonardo Babun, Hidayet Aksu, and A. Selcuk Uluagac. Aegis: A context-aware security framework for smart home systems. In *Proceedings of the 35th Annual Computer Security Applications Conference*, ACSAC '19, pages 28–41. Association for Computing Machinery, 2019. doi:10.1145/3359789.3359840.

[74] Sergii Lysenko, Oleg Savenko, Kira Bobrovnikova, and Andrii Kryshchuk. Self-adaptive system for the corporate area network resilience in the presence of botnet cyberattacks. In Piotr Gaj, Michał Sawicki, Grażyna Suchacka, and Andrzej Kwiecień, editors, *Computer Networks*, pages 385–401. Springer International Publishing, 2018. doi:10.1007/978-3-319-92459-5_31.

[75] Daniel Díaz López, María Blanco Uribe, Claudia Santiago Cely, Andrés Vega Torres, Nicolás Moreno Guataquira, Stefany Morón Castro, Pantaleone Nespoli, Félix Gómez Mármol, and Constantinos Kolias. Shielding iot against cyber-attacks: An event-based approach using siem. *Wireless Communications and Mobile Computing*, 2018, 2018. doi:10.1155/2018/3029638.

[76] Edmilson P. da Costa Júnior, Carlos Eduardo da Silva, Marcos Pinheiro, and Silvio Sampaio. A new approach to deploy a self-adaptive distributed firewall. *Journal of Internet Services and Applications*, 9(1), 2018. doi:10.1186/s13174-018-0083-6.

[77] Vijay Varadharajan, Kallol Karmakar, Uday Tupakula, and Michael Hitchens. A policy-based security architecture for software-defined

networks. *IEEE Transactions on Information Forensics and Security*, 14(4):897–912, 2019. doi:10.1109/TIFS.2018.2868220.

[78] Kotaro Shibata, Hiroki Nakayama, Tsunemasa Hayashi, and Shingo Ata. Establishing pdca cycles for agile network management in sdn/nfv infrastructure. In *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, pages 619–625, 2015. doi:10.1109/INM.2015.7140346.

[79] Dayna Eidle, Si Ya Ni, Casimer DeCusatis, and Anthony Sager. Autonomic security for zero trust networks. In *2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON)*, pages 288–293, 2017. doi:10.1109/UEMCON.2017.8249053.

[80] Jorge da Silva, Alexandre Braga, Cecília Rubira, and Ricardo Dahab. An approach for adaptive security of cloud applications within the atmosphere platform. In *Anais do XIX Simpósio Brasileiro de Segurança da Informação e de Sistemas Computacionais*, pages 397–402. SBC, 2019. doi:10.5753/sbseg.2019.13988.

[81] Tewfiq El-Maliki, Nabil Abdennadher, and Mohamed Nizar Bouchedakh. Adaptive security in cloud and edge networks. In *Proceedings of the 13th International Conference on Systems, ICONS 2018, 22-26 April 2018, Athens, Greece*, 1, 2018. URL: https://www.semanticscholar.org/paper/2659e5cdb4867895b1b5c22212e1d77a7c164699.

[82] Olumide Malomo, Danda B. Rawat, and Moses Garuba. A federated cloud computing framework for adaptive cyber defense and distributed computing. In *2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 1–6, 2017. doi:10.1109/INFCOMW.2017.8376184.

[83] Jim Boehm, Nick Curcio, Peter Merrath, Lucy Shenton, and Tobias Stähle. The risk-based approach to cybersecurity. Technical report, McKinsey & Company, 2019. URL: https://www.mckinsey.com/capabilities/risk-and-resilience/our-insights/the-risk-based-approach-to-cybersecurity.

[84] Andrew W. M'manga. *Designing for cyber security risk-based decision making*. PhD thesis, Bournemouth University, 2021. URL: https://eprints.bournemouth.ac.uk/33280/.

[85] Igor Linkov, Elke Anklam, Zachary A. Collier, Daniel DiMase, and Ortwin Renn. Risk-based standards: integrating top–down and bottom–up approaches. *Environment Systems and Decisions*, 34(1):134–137, 2014. doi:10.1007/s10669-014-9488-3.

[86] Reijo M. Savola, Heimo Pentikäinen, and Moussa Ouedraogo. Towards security effectiveness measurement utilizing risk-based security assurance. In *2010 Information Security for South Africa*, pages 1–8, 2010. doi:10.1109/ISSA.2010.5588322.

[87] J. Patrick Ravenel. Effective operational security metrics. *EDPACS*, 33(12):10–19, 2006. doi:10.1201/1079.07366981/46050.33.12.20060601/93399.2.

[88] Changbo Ke, Jiayu Wu, Fu Xiao, Zhiqiu Huang, and Yunfei Meng. A privacy risk assessment scheme for fog nodes in access control system. *IEEE Transactions on Reliability*, pages 1–14, 2021. doi:10.1109/TR.2021.3103906.

[89] Mariusz Sepczuk and Zbigniew Kotulski. A new risk-based authentication management model oriented on user's experience. *Computers & Security*, 73:17–33, 2018. doi:10.1016/j.cose.2017.10.002.

[90] Roland H. Steinegger, Daniel Deckers, Pascal Giessler, and Sebastian Abeck. Risk-based authenticator for web applications. In *Proceedings of the 21st European Conference on Pattern Languages of Programs*, EuroPlop '16, New York, NY, USA, 2016. Association for Computing Machinery. doi:10.1145/3011784.3011800.

[91] Waqas Aman and Einar Snekkenes. Edas: An evaluation prototype for autonomic event-driven adaptive security in the internet of things. *Future Internet*, 7(3):225–256, 2015. doi:10.3390/fi7030225.

[92] Hany F. Atlam, Robert J. Walters, Gary B. Wills, and Joshua Daniel. Fuzzy logic with expert judgment to implement an adaptive risk-based access control model for iot. *Mobile Networks and Applications*, 26(6):2545–2557, 2021. doi:10.1007/s11036-019-01214-w.

[93] Shefiu Olusegun Ganiyu and Rasheed Gbenga Jimoh. *Extended Risk-Based Context-Aware Model for Dynamic Access Control in Bring Your Own Device Strategy*, pages 295–315. Springer International Publishing, 2021. doi:10.1007/978-3-030-66288-2_12.

[94] Hany Atlam, Ahmed Alenezi, Raid Hussein, and Gary Wills. Validation of an adaptive risk-based access control model for the internet of things. *International Journal of Computer Network and Information Security*, 1:26–35, 2018. doi:10.5815/ijcnis.2018.01.04.

[95] Daniel Ricardo dos Santos, Roberto Marinho, Gustavo Roecker Schmitt, Carla Merkle Westphall, and Carlos Becker Westphall. A framework and risk assessment approaches for risk-based access control in the cloud. *Journal of Network and Computer Applications*, 74:86–97, 2016. doi:10.1016/j.jnca.2016.08.013.

[96] Amir Rahmati, Earlence Fernandes, Kevin Eykholt, and Atul Prakash. Tyche: A risk-based permission model for smart homes. In *2018 IEEE*

Cybersecurity Development (SecDev), pages 29–36, 2018. doi:10.1109/SecDev.2018.00012.

[97] Aiguo Chen, Hanwen Xing, Kun She, and Guiduo Duan. A dynamic risk-based access control model for cloud computing. In *2016 IEEE International Conferences on Big Data and Cloud Computing (BDCloud), Social Computing and Networking (SocialCom), Sustainable Computing and Communications (SustainCom) (BDCloud-SocialCom-SustainCom)*, pages 579–584, 2016. doi:10.1109/BDCloud-SocialCom-SustainCom.2016.90.

[98] Fabio Martinelli, Christina Michailidou, Paolo Mori, and Andrea Saracino. Too long, did not enforce: A qualitative hierarchical risk-aware data usage control model for complex policies in distributed environments. In *Proceedings of the 4th ACM Workshop on Cyber-Physical System Security*, CPSS '18, pages 27–37. Association for Computing Machinery, 2018. doi:10.1145/3198458.3198463.

[99] Nadia Metoui, Michele Bezzi, and Alessandro Armando. *Risk-Based Privacy-Aware Access Control for Threat Detection Systems*, pages 1–30. Springer Berlin Heidelberg, 2017. doi:10.1007/978-3-662-56266-6_1.

[100] Satiaseelan Selvan and Manmeet Mahinderjit Singh. Adaptive contextual risk-based model to tackle confidentiality-based attacks in fog-iot paradigm. *Computers*, 11(2), 2022. doi:10.3390/computers11020016.

[101] Daniel Díaz-López, Ginés Dólera-Tormo, Félix Gómez-Mármol, and Gregorio Martínez-Pérez. Dynamic counter-measures for risk-based access control systems: An evolutive approach. *Future Generation Computer Systems*, 55:321–335, 2016. doi:10.1016/j.future.2014.10.012.

[102] Wided Ben Daoud, Amel Meddeb-Makhlouf, and Faouzi Zarai. A model of role-risk based intrusion prevention for cloud environment. In *2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC)*, pages 530–535, 2018. doi:10.1109/IWCMC.2018.8450466.

[103] Jing Liu, Rongchao Liu, and Yingxu Lai. Risk-based dynamic identity authentication method based on the ucon model. *Security and Communication Networks*, 2022. doi:10.1155/2022/2509267.

[104] Maria Papaioannou, Georgios Mantas, Aliyah Essop, Phil Cox, Ifiok E. Otung, and Jonathan Rodriguez. Risk-based adaptive user authentication for mobile passenger id devices for land/sea border control. In *2021 IEEE 26th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, pages 1–6, 2021. doi:10.1109/CAMAD52502.2021.9617802.

[105] Emanuele Celoria. A context-aware risk-based authorization system. Bachelor's thesis, Politecnico di Torino, 2018. URL: https://webthesis.biblio.polito.it/secure/7517/1/tesi.pdf.

[106] Habtamu Abie, Trenton Schulz, and Reijo Savola. Adaptive security and trust management for autonomous messaging systems. *arXiv*, 2022. doi:10.48550/ARXIV.2203.03559.

[107] Alejandro Molina Zarca, Jorge Bernal Bernabe, Ruben Trapero, Diego Rivera, Jesus Villalobos, Antonio Skarmeta, Stefano Bianchi, Anastasios Zafeiropoulos, and Panagiotis Gouvas. Security management architecture for nfv/sdn-aware iot systems. *IEEE Internet of Things Journal*, 6(5):8005–8020, 2019. doi:10.1109/JIOT.2019.2904123.

[108] G. Gonzalez-Granadillo, S. Dubus, A. Motzek, J. Garcia-Alfaro, E. Alvarez, M. Merialdo, S. Papillon, and H. Debar. Dynamic risk management

response system to handle cyber threats. *Future Generation Computer Systems*, 83:535–552, 2018. doi:10.1016/j.future.2017.05.043.

[109] Bata Krishna Tripathy, Debi Prasad Das, Swagat Kumar Jena, and Padmalochan Bera. Risk based security enforcement in software defined network. *Computers & Security*, 78:321–335, 2018. doi:10.1016/j.cose.2018.07.010.

[110] Sharmin Jahan, Ian Riley, Charles Walter, Rose F. Gamble, Matt Pasco, Philip K. McKinley, and Betty H.C. Cheng. Mape-k/mape-sac: An interaction framework for adaptive systems with security assurance cases. *Future Generation Computer Systems*, 109:197–209, 2020. doi:10.1016/j.future.2020.03.031.

[111] Abdeljebar Mansour, Mohamed Sadik, Essaïd Sabir, and Mohamed Azmi. A context-aware multimodal biometric authentication for cloud-empowered systems. In *2016 International Conference on Wireless Networks and Mobile Communications (WINCOM)*, pages 278–285, 2016. doi:10.1109/WINCOM.2016.7777227.

[112] Yuanqing Qin, Qi Zhang, Chunjie Zhou, and Naixue Xiong. A risk-based dynamic decision-making approach for cybersecurity protection in industrial control systems. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 50(10):3863–3870, 2020. doi:10.1109/TSMC.2018.2861715.

[113] Nouri Alnahawi, Nicolai Schmitt, Alexander Wiesmaier, Andreas Heinemann, and Tobias Grasmeyer. On the state of crypto-agility. *Cryptology ePrint Archive*, 1(2023/487), 2023. URL: https://ia.cr/2023/487.

[114] Quang Bao Le, Roman Seidl, and Roland W. Scholz. Feedback loops and types of adaptation in the modelling of land-use decisions in an

agent-based simulation. *Environmental Modelling & Software*, 27-28:83–96, 2012. doi:10.1016/j.envsoft.2011.09.002.

[115] Yuriy Brun, Giovanna Di Marzo Serugendo, Cristina Gacek, Holger Giese, Holger Kienle, Marin Litoiu, Hausi Müller, Mauro Pezzè, and Mary Shaw. *Engineering Self-Adaptive Systems through Feedback Loops*, pages 48–70. Springer Berlin Heidelberg, 2009. doi:10.1007/978-3-642-02161-9_3.

[116] Birk Emil Karlsen-Bæck. Modelling control loops for sps-lhc beam transfer studies. Master's thesis, Norwegian University of Science and Technology, 2022. URL: https://hdl.handle.net/11250/3024801.

[117] Annette Löf. Exploring adaptability through learning layers and learning loops. *Environmental Education Research*, 16(5-6):529–543, 2010. doi:10.1080/13504622.2010.505429.

[118] Markus Fietz. Approaches to adaptive iteration: a comparative review. Discussion draft, 2013. URL: http://www.cernquest.com/wp-content/uploads/2013/08/Adaptive-iteration-review-Fietz.pdf.

[119] Simon Dobson, Spyros Denazis, Antonio Fernández, Dominique Gaïti, Erol Gelenbe, Fabio Massacci, Paddy Nixon, Fabrice Saffre, Nikita Schmidt, and Franco Zambonelli. A survey of autonomic communications. *ACM Transactions on Autonomous and Adaptive Systems*, 1(2):223–259, 2006. doi:10.1145/1186778.1186782.

[120] Stephen H. Haeckel. Adaptive enterprise design: The sense-and-respond model. *Planning Review*, 23(3):6–42, 1995. doi:10.1108/eb054506.

[121] Rogério de Lemos, Holger Giese, Hausi A. Müller, Mary Shaw, Jesper Andersson, Marin Litoiu, Bradley Schmerl, Gabriel Tamura, Norha M. Villegas, Thomas Vogel, Danny Weyns, Luciano Baresi, Basil Becker,

Nelly Bencomo, Yuriy Brun, Bojan Cukic, Ron Desmarais, Schahram Dustdar, Gregor Engels, Kurt Geihs, Karl M. Göschka, Alessandra Gorla, Vincenzo Grassi, Paola Inverardi, Gabor Karsai, Jeff Kramer, Antónia Lopes, Jeff Magee, Sam Malek, Serge Mankovskii, Raffaela Mirandola, John Mylopoulos, Oscar Nierstrasz, Mauro Pezzè, Christian Prehofer, Wilhelm Schäfer, Rick Schlichting, Dennis B. Smith, João Pedro Sousa, Ladan Tahvildari, Kenny Wong, and Jochen Wuttke. *Software Engineering for Self-Adaptive Systems: A Second Research Roadmap*, pages 1–32. Springer Berlin Heidelberg, 2013. doi:10.1007/978-3-642-35813-5_1.

[122] Alireza Shojaifar, Samuel A. Fricker, and Martin Gwerder. Automating the communication of cybersecurity knowledge: Multi-case study. In Lynette Drevin, Suné Von Solms, and Marianthi Theocharidou, editors, *Information Security Education. Information Security in Action*, pages 110–124. Springer International Publishing, 2020. doi:10.1007/978-3-030-59291-2_8.

[123] Shouhuai Xu. The cybersecurity dynamics way of thinking and landscape. In *Proceedings of the 7th ACM Workshop on Moving Target Defense*, MTD'20, pages 69–80. Association for Computing Machinery, 2020. doi:10.1145/3411496.3421225.

[124] Evangelina Lara, Leocundo Aguilar, Mauricio A. Sanchez, and Jesús A. García. *Adaptive Security Based on MAPE-K: A Survey*, pages 157–183. Springer International Publishing, 2019. doi:10.1007/978-3-030-17985-4_7.

[125] Mirko Sokovic, Dusko Pavletic, and K Kern Pipan. Quality improvement methodologies–pdca cycle, radar matrix, dmaic and dfss. *Journal of achievements in materials and manufacturing engineering*, 43(1):476–

483, 2010. URL: http://yadda.icm.edu.pl/yadda/element/bwmeta1.element.baztech-0bc1ff2d-3614-4c9b-894d-9478b3e9f72e.

[126] Michal Pietrzak and Joanna Paliszkiewicz. Framework of strategic learning: The pdca cycle. *Management*, 10(2):149–161, 2015. URL: https://ideas.repec.org/a/mgt/youmng/v10y2015i2p149-161.html.

[127] Hillary Sillitto. *Nature of an engineered system: illustrated from engineering artefacts and complex systems*, chapter 39, pages 983–1039. Springer Nature Singapore, 2021. doi:10.1007/978-981-15-0720-5.

[128] Wissam Abbass, Zineb Bakraouy, Amine Baina, and Mostafa Bella. Assessing the internet of things security risks. *Journal of Communications*, 14(10):958–964, 2019. doi:10.12720/jcm.14.10.958-964.

[129] Elastic. Online, 2023. Accessed: 2023-04-22. URL: https://www.elastic.co/.

[130] Najat Tissir, Said El Kafhali, and Noureddine Aboutabit. Cybersecurity management in cloud computing: semantic literature review and conceptual framework proposal. *Journal of Reliable Intelligent Environments*, 7(2):69–84, 2021. doi:10.1007/s40860-020-00115-0.

[131] John R Boyd. The essence of winning and losing. *Unpublished lecture notes*, 12(23):123–125, 1996.

[132] Tero Bodström and Timo Hämäläinen. A novel method for detecting apt attacks by using ooda loop and black swan theory. In Xuemin Chen, Arunabha Sen, Wei Wayne Li, and My T. Thai, editors, *Computational Data and Social Networks*, pages 498–509. Springer International Publishing, 2018. doi:10.1007/978-3-030-04648-4_42.

[133] Yit Loong Teh, Yao Tong Tan, and Siaw Lang Wong. 5g cybersecurity: Risk assessment and incident response in the healthcare industry. In *International Conference on Digital Transformation and Applications (ICDXA) 2021*, pages 145–152, 2021. doi:10.56453/icdxa.2021.1015.

[134] Paolo Arcaini, Elvinia Riccobene, and Patrizia Scandurra. Modeling and analyzing mape-k feedback loops for self-adaptation. In *2015 IEEE/ACM 10th International Symposium on Software Engineering for Adaptive and Self-Managing Systems*, pages 13–23, 2015. doi:10.1109/SEAMS.2015.10.

[135] J. Vizcarrondo, J. Aguilar, E. Exposito, and A. Subias. Mape-k as a service-oriented architecture. *IEEE Latin America Transactions*, 15(6):1163–1175, 2017. doi:10.1109/TLA.2017.7932705.

[136] Adrián Romero-Garcés, Alejandro Hidalgo-Paniagua, Martín González-García, and Antonio Bandera. On managing knowledge for mape-k loops in self-adaptive robotics using a graph-based runtime model. *Applied Sciences*, 12(17), 2022. doi:10.3390/app12178583.

[137] Christopher Rouff, Lanier Watkins, Roy Sterritt, and Salim Hariri. Sok: Autonomic cybersecurity - securing future disruptive technologies. In *2021 IEEE International Conference on Cyber Security and Resilience (CSR)*, pages 66–72, 2021. doi:10.1109/CSR51186.2021.9527908.

[138] Ricardo Bañuelas and Jiju Antony. Six sigma or design for six sigma? *The TQM Magazine*, 16(4):250–263, 2004. doi:10.1108/09544780410541909.

[139] Rama Shankar. *Process improvement using six sigma: a DMAIC guide*. Ilustrada. Quality Press, 2009.

[140] Mariam Lahami and Moez Krichen. A survey on runtime testing of dynamically adaptable and distributed systems. *Software Quality Journal*, 29(2):555–593, 2021. doi:10.1007/s11219-021-09558-x.

[141] Danny Weyns, Bradley Schmerl, Vincenzo Grassi, Sam Malek, Raffaela Mirandola, Christian Prehofer, Jochen Wuttke, Jesper Andersson, Holger Giese, and Karl M. Göschka. *On Patterns for Decentralized Control in Self-Adaptive Systems*, pages 76–107. Springer Berlin Heidelberg, 2013. doi:10.1007/978-3-642-35813-5_4.

[142] Ricardo Sanz Bravo and Julita Bermejo Alonso. Consciousness and understanding in autonomous systems. In *TOCAIS 2019 Towards Conscious AI Systems Papers of the 2019 Towards Conscious AI Systems Symposium co-located with the Association for the Advancement of Artificial Intelligence 2019 Spring Symposium Series (AAAI SSS-19)*, pages 1–9. CEUR-WS, 2019. URL: https://oa.upm.es/65225/.

[143] Curtis John Marshall, Blake Roberts, and Michael Grenn. Intelligent control & supervision for autonomous system resilience in uncertain worlds. In *2017 3rd International Conference on Control, Automation and Robotics (ICCAR)*, pages 438–443, 2017. doi:10.1109/ICCAR.2017.7942734.

[144] Snort. Online. Accessed: 2023-04-22. URL: https://www.snort.org/.

[145] Kismet. Online. Accessed: 2023-04-22. URL: https://www.kismetwireless.net/.

[146] Greenbone openvas. Online. Accessed: 2023-04-22. URL: https://www.openvas.org/.

[147] Open source cloud computing infrastructure. Online. Accessed: 2023-04-22. URL: https://www.openstack.org/.

[148] B. Claise. Cisco systems netflow services export version 9. Online, 2004. rfc 3954, Accessed: 2023-04-22. URL: https://www.rfc-editor.org/rfc/rfc3954.

[149] Yucong Duan, Guohua Fu, Nianjun Zhou, Xiaobing Sun, Nanjangud C. Narendra, and Bo Hu. Everything as a service (xaas) on the cloud: Origins, current and future trends. In *2015 IEEE 8th International Conference on Cloud Computing*, pages 621–628, 2015. doi:10.1109/CLOUD.2015.88.

[150] Sugam Sharma. Evolution of as-a-service era in cloud. *arXiv*, 2015. doi:10.48550/ARXIV.1507.00939.

[151] Nagios exchange. Online, 2009. Accessed: 2023-04-22. URL: https://exchange.nagios.org/.

[152] Fluentd - list of all plugins. Online, 2010. Accessed: 2023-04-22. URL: https://www.fluentd.org/plugins/all.

[153] Telegraf plugins. Online, 2023. Accessed: 2023-04-22. URL: https://www.influxdata.com/products/integrations/.

[154] Gluu: Open source identity and access management. Online, 2023. Accessed: 2023-04-22. URL: https://www.gluu.org/.

[155] Docker. Online, 2023. Accessed: 2023-04-22. URL: https://www.docker.com/.

[156] G. Judd and P. Steenkiste. Providing contextual information to pervasive computing applications. In *Proceedings of the First IEEE International Conference on Pervasive Computing and Communications 2003 (PerCom 2003)*, pages 133–142, 2003. doi:10.1109/PERCOM.2003.1192735.

[157] Anind K. Dey. Understanding and using context. *Personal and Ubiquitous Computing*, 5(1):4–7, 2001. doi:10.1007/s007790170019.

[158] Antti Evesti, Habtamu Abie, and Reijo Savola. Security measuring for self-adaptive security. In *Proceedings of the 2014 European Conference on Software Architecture Workshops*, ECSAW '14, New York, NY, USA, 2014. Association for Computing Machinery. doi:10.1145/2642803.2642808.

[159] Ryan Leirvik. *Measure the Problem*, pages 137–164. Apress, 2022. doi:10.1007/978-1-4842-7821-5_7.

[160] B. Jones Richard. *Risk-based management: a reliability-centered approach*. Houston US: Gulf Publishing Company, 1995.

[161] Yavor Valentinov Papazov. *Cybersecurity Metrics*, pages 1–18. NATO Science and Technology Organization (STO), 2019. doi:10.14339/STO-EN-IST-170.

[162] Yussuf Ahmed, Syed Naqvi, and Mark Josephs. Cybersecurity metrics for enhanced protection of healthcare it systems. In *2019 13th International Symposium on Medical Information and Communication Technology (ISMICT)*, pages 1–9, 2019. doi:10.1109/ISMICT.2019.8744003.

[163] Domenic Antonucci. *The cyber risk handbook: Creating and measuring effective cybersecurity capabilities*. John Wiley & Sons, 2017.

[164] Hyeisun Cho, Seulgi Lee, Nakhyun Kim, Byungik Kim, and Junhyung Park. Method of quantification of cyber threat based on indicator of compromise. In *2018 International Conference on Platform Technology and Service (PlatCon)*, pages 1–6, 2018. doi:10.1109/PlatCon.2018.8472733.

[165] Morey J. Haber and Darran Rolls. *Indicators of Compromise*, pages 103–105. Apress, 2020. doi:10.1007/978-1-4842-5165-2_9.

[166] K Paine, O Whitehouse, and J Sellwood. Indicators of compromise (iocs) and their role in attack defence. *UK National Cyber Security Centre*, 2022. Draft. URL: https://datatracker.ietf.org/doc/html/draft-paine-smart-indicators-of-compromise-04.

[167] Scott E. Jasper. U.s. cyber threat intelligence sharing frameworks. *International Journal of Intelligence and CounterIntelligence*, 30(1):53–65, 2017. doi:10.1080/08850607.2016.1230701.

[168] Aldin Vehabovic, Nasir Ghani, Elias Bou-Harb, Jorge Crichigno, and Aysegül Yayimli. Ransomware detection and classification strategies. In *2022 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom)*, pages 316–324, 2022. doi:10.1109/BlackSeaCom54372.2022.9858296.

[169] Fuat Özçakmak. Supplementing isrm models by kri implementation. Master's thesis, Middle East Technical University, 2019. URL: https://open.metu.edu.tr/handle/11511/43721.

[170] Paul Rohmeyer and Jennifer L. Bayuk. *How Do I Manage This?*, pages 125–156. Apress, 2019. doi:10.1007/978-1-4842-4194-3_6.

[171] Ann Rodriguez. *Monitoring and Review Using Key Risk Indicators (KRIs)*, chapter 11, pages 159–170. John Wiley & Sons, Ltd, 2017. doi:10.1002/9781119309741.ch11.

[172] Kevin Stine, Stephen Quinn, Greg Witte, and R.K. Gardner. Integrating cybersecurity and enterprise risk management (erm). Technical report, National Institute of Standards and Technology, 2020. nistir 8286, Accessed: 2023-04-22. doi:10.6028/NIST.IR.8286.

[173] Subhash V. Pingale and Sanjay R. Sutar. Analysis of web application firewalls, challenges, and research opportunities. In Amit Kumar, Sabrina Senatore, and Vinit Kumar Gunjan, editors, *ICDSMLA 2020*, pages 239–248. Springer Singapore, 2022. doi:10.1007/978-981-16-3690-5_21.

[174] Abdul Razzaq, Ali Hur, Sidra Shahbaz, Muddassar Masood, and H Farooq Ahmad. Critical analysis on web application firewall solutions. In *2013 IEEE Eleventh International Symposium on Autonomous Decentralized Systems (ISADS)*, pages 1–6, 2013. doi:10.1109/ISADS.2013.6513431.

[175] Bernhard Garn, Daniel Sebastian Lang, Manuel Leithner, D. Richard Kuhn, Raghu Kacker, and Dimitris E. Simos. Combinatorially xssing web application firewalls. In *2021 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW)*, pages 85–94, 2021. doi:10.1109/ICSTW52544.2021.00026.

[176] Hannes Holm and Mathias Ekstedt. Estimates on the effectiveness of web application firewalls against targeted attacks. *Information Management & Computer Security*, 21(4):250–265, 2013. doi:10.1108/IMCS-11-2012-0064.

[177] Simon Applebaum, Tarek Gaber, and Ali Ahmed. Signature-based and machine-learning-based web application firewalls: A short survey. *Procedia Computer Science*, 189:359–367, 2021. AI in Computational Linguistics. doi:10.1016/j.procs.2021.05.105.

[178] Ralf Funk, Nico Epp, and Cristian Cappo A. Anomaly-based web application firewall using http-specific features and one-class svm. *Revista Eletrônica Argentina-Brasil de Tecnologias da Informação e da Comunicação*, 2(1), 2018. doi:10.5281/zenodo.1336812.

[179] Carmen Torrano-Gimenez, Alejandro Perez-Villegas, and Gonzalo Alvarez. A self-learning anomaly-based web application firewall. In Álvaro Herrero, Paolo Gastaldo, Rodolfo Zunino, and Emilio Corchado, editors, *Computational Intelligence in Security for Information Systems*, pages 85–92. Springer Berlin Heidelberg, 2009. doi:10.1007/978-3-642-04091-7_11.

[180] Batuhan IŞiker and İbrahim SoĞukpinar. Machine learning based web application firewall. In *2021 2nd International Informatics and Software Engineering Conference (IISEC)*, pages 1–6, 2021. doi:10.1109/IISEC54230.2021.9672335.

[181] Andrea Valenza, Luca Demetrio, Gabriele Costa, and Giovanni Lagorio. Waf-a-mole: An adversarial tool for assessing ml-based wafs. *SoftwareX*, 11, 2020. doi:10.1016/j.softx.2019.100367.

[182] Manoel Domingues Junior and Nelson F.F. Ebecken. A new waf architecture with machine learning for resource-efficient use. *Computers & Security*, 106, 2021. doi:10.1016/j.cose.2021.102290.

[183] Gustavo Betarte, Eduardo Gimenez, Rodrigo Martinez, and Alvaro Pardo. Improving web application firewalls through anomaly detection. In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 779–784, 2018. doi:10.1109/ICMLA.2018.00124.

[184] Ali Moradi Vartouni, Matin Shokri, and Mohammad Teshnehlab. Auto-threshold deep svdd for anomaly-based web application firewall. *TechRxiv*, 2021. doi:10.36227/techrxiv.15135468.v1.

[185] Bronjon Gogoi, Tasiruddin Ahmed, and Hemanta Kumar Saikia. Detection of xss attacks in web applications: A machine learning ap-

proach. *International Journal of Innovative Research in Computer Science & Technology (IJIRCST)*, 9(1):2347–5552, 2021. doi:10.21276/ijircst.2021.9.1.1.

[186] Michiaki Ito and Hitoshi Iyatomi. Web application firewall using character-level convolutional neural network. In *2018 IEEE 14th International Colloquium on Signal Processing Its Applications (CSPA)*, pages 103–106, 2018. doi:10.1109/CSPA.2018.8368694.

[187] Waleed Bin Shahid, Baber Aslam, Haider Abbas, Saad Bin Khalid, and Hammad Afzal. An enhanced deep learning based framework for web attacks detection, mitigation and attacker profiling. *Journal of Network and Computer Applications*, 198:103270, 2022. doi:10.1016/j.jnca.2021.103270.

[188] Claudio Gambella, Bissan Ghaddar, and Joe Naoum-Sawaya. Optimization problems for machine learning: A survey. *European Journal of Operational Research*, 290(3):807–828, 2021. URL: https://doi.org/10.3233/IDT-190160, doi:10.1016/j.ejor.2020.08.045.

[189] Shiliang Sun, Zehui Cao, Han Zhu, and Jing Zhao. A survey of optimization methods from a machine learning perspective. *IEEE Transactions on Cybernetics*, 50(8):3668–3681, 2020. doi:10.1109/TCYB.2019.2950779.

[190] Fumihiro Kumeno. Sofware engneering challenges for machine learning applications: A literature review. *Intelligent Decision Technologies*, 13:463–476, 2019. doi:10.3233/IDT-190160.

[191] Sebastian Schelter, Felix Biessmann, Tim Januschowski, David Salinas, Stephan Seufert, and Gyuri Szarvas. On challenges in machine learning model management. *IEEE Data Engineering Bulletin*, 41:5–15, 2018. URL: https://api.semanticscholar.org/CorpusID:83459629.

[192] Mazen Saleh and Hassan Gomaa. Separation of concerns in software product line engineering. In *Proceedings of the 2005 Workshop on Modeling and Analysis of Concerns in Software*, MACS '05, pages 1–5. Association for Computing Machinery, 2005. doi:10.1145/1083125.1083139.

[193] Evan You. Vue.js - the progressive javascript framework. Online, 2014. Accessed: 2023-04-22. URL: https://vuejs.org/.

[194] Node.js. Online, 2009. Accessed: 2023-04-22. URL: https://nodejs.org/.

[195] Mysql. Online, 1995. Accessed: 2023-04-22. URL: https://www.mysql.com/.

[196] Nginx. Online, 2009. Accessed: 2023-04-22. URL: https://nginx.org/.

[197] Oracle Corporation. *MySQL: MySQL Community Server*, 2021. Accessed: 2023-04-22. URL: https://dev.mysql.com/doc/refman/8.0/en/.

[198] Modsecurity: Open source web application firewall. Online, 2023. Accessed: 2023-04-22. URL: https://github.com/SpiderLabs/ModSecurity.

[199] Bjarne Stroustrup. *The C++ programming language*. Addison-Wesley Professional, Boston, MA, 4th edition, 2013.

[200] Python. Online, 2001. Accessed: 2023-04-22. URL: https://www.python.org/.

[201] Twitter api. Online, 2006. Accessed: 2023-04-22. URL: https://developer.twitter.com/en/docs/twitter-api.

[202] Owasp zap - zed attack proxy project. Online, 2010. Accessed: 2023-04-22. URL: https://www.zaproxy.org/.

[203] Burp suite - the leading software for web security testing. Online, 2006. Accessed: 2023-04-22. URL: https://portswigger.net/burp.

[204] Mandiant. M-trends 2022 insights into today's top cyber security trends and attacks. Technical report, Mandiant, 2022. Accessed: 2023-04-22. URL: https://www.mandiant.com/m-trends.

[205] Ben Buchanan. A national security research agenda for cybersecurity and artificial intelligence. Technical report, Center for Security and Emerging Technology, 2020. Accessed: 2023-04-22. URL: https://cset.georgetown.edu/publication/a-national-security-research-agenda-for-cybersecurity-and-artificial-intelligence/.

[206] Mitre att&ck. Online, 2013. Accessed: 2023-04-22. URL: https://attack.mitre.org/.

[207] Gwanghyun Ahn, Kookjin Kim, Wonhyung Park, and Dongkyoo Shin. Malicious file detection method using machine learning and interworking with mitre att&ck framework. *Applied Sciences*, 12(21), 2022. doi:10.3390/app122110761.

[208] Yashovardhan Sharma, Simon Birnbach, and Ivan Martinovic. Radar: Effective network-based malware detection based on the mitre att&ck framework. *arXiv*, 2022. doi:10.48550/arXiv.2212.03793.

[209] Yuvraj Sanjayrao Takey, Sai Gopal Tatikayala, Mahesh Uttam Patil, Lakshmi Eswari P. R, and Satyanadha Sarma Samavedam. Real time multistage attack detection leveraging machine learning and mitre framework. In *2022 11th International Conference on System Modeling*

& Advancement in Research Trends (SMART), pages 1226–1230, 2022. doi:10.1109/SMART55829.2022.10047248.

[210] Yi-Ting Huang, Chi Yu Lin, Ying-Ren Guo, Kai-Chieh Lo, Yeali S. Sun, and Meng Chang Chen. Open source intelligence for malicious behavior discovery and interpretation. *IEEE Transactions on Dependable and Secure Computing*, 19(2):776–789, 2022. doi:10.1109/TDSC.2021.3119008.

[211] Firdevs Sevde Toker, Kevser Ovaz Akpinar, and İbrahim Özçelik. Mitre ics attack simulation and detection on ethercat based drinking water system. In *2021 9th International Symposium on Digital Forensics and Security (ISDFS)*, pages 1–6, 2021. doi:10.1109/ISDFS52919.2021.9486331.

[212] Yuvraj Sanjayrao Takey, Sai Gopal Tatikayala, Satyanadha Sarma Samavedam, P R Lakshmi Eswari, and Mahesh Uttam Patil. Real time early multi stage attack detection. In *2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS)*, volume 1, pages 283–290, 2021. doi:10.1109/ICACCS51430.2021.9441956.

[213] Seungoh Choi, Jongwon Choi, Jeong-Han Yun, Byung-Gil Min, and HyoungChun Kim. Expansion of ICS testbed for security validation based on MITRE ATT&CK techniques. In *13th USENIX Workshop on Cyber Security Experimentation and Test (CSET 20)*. USENIX Association, 2020. URL: https://www.usenix.org/conference/cset20/presentation/choi.

[214] Anca Georgiana Butnar. Análisis automático de técnicas, tácticas y procedimientos empleados por malware mediante deep learning. Master's

thesis, Universidad Rey Juan Carlos, Madrid, Spain, 2019. Dirigido por Marta Beltrán.

[215] Cuckoo sandbox. Online, 2014. Accessed: 2023-04-22. URL: https://cuckoosandbox.org/.

[216] Mitre att&ck - chopstick. Online, 2017. Accessed: 2023-04-22. URL: https://attack.mitre.org/software/S0023/.

[217] Microsoft defender for endpoint. Online, 2023. Accessed: 2023-04-22. URL: https://learn.microsoft.com/en-us/microsoft-365/security/defender-endpoint/microsoft-defender-endpoint?view=o365-worldwide.

[218] Powershell documentation. Online, 2023. Accessed: 2023-04-22. URL: https://docs.microsoft.com/en-us/powershell/.

[219] Miguel Calvo and Marta Beltrán. A model for risk-based adaptive security controls. *Computers & Security*, 115, 2022. doi:10.1016/j.cose.2022.102612.

[220] Miguel Calvo and Marta Beltrán. An adaptive web application firewall. In *Proceedings of the 19th International Conference on Security and Cryptography - SECRYPT*, pages 96–107. INSTICC, SciTePress, 2022. URL: https://www.scitepress.org/PublishedPapers/2022/111469/pdf/index.html, doi:10.5220/0011146900003283.

[221] Miguel Calvo and Marta Beltrán. Remote attestation as a service for edge-enabled iot. In *2021 IEEE International Conference on Services Computing (SCC)*, pages 329–339, 2021. doi:10.1109/SCC53864.2021.00046.

[222] Miguel Calvo and Marta Beltrán. Verificación remota de controles de seguridad en contextos de seguridad adaptativa. In *Actas del VI*

Jornadas Nacionales de Investigación en Ciberseguridad (JNIC 2021 LIVE), pages 189–196. Ediciones de la Universidad de Castilla-La Mancha, 2021. URL: https://ruidera.uclm.es/xmlui/handle/10578/28652, doi:10.18239/jornadas_2021.34.42.

[223] Javier Sánchez García-Ochoa. Seguridad adaptativa para aplicaciones en la nube. Trabajo fin de grado, Universidad Rey Juan Carlos, Madrid, Spain, 2022. Co-dirigido por Miguel Calvo y Marta Beltrán.

Atribuciones

Todos los iconos utilizados en las figuras de esta tesis han sido extraídos de flaticon.com y están sujetos a las condiciones de uso de la plataforma.