ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA

GRADO EN INGENIERIA DE LA CIBERSEGURIDAD

Curso Académico 2023/2024

Trabajo Fin de Grado

DESARROLLO DE UNA PLATAFORMA CTF EDUCATIVA PARA PRINCIPIANTES

Autor: Iván Domínguez Romero

Directores: César Cáceres Taladriz

RESUMEN

Este Trabajo de Fin de Grado se centra en el desarrollo de una plataforma de Captura la Bandera (CTF), concebida especialmente para introducir a los principiantes en el mundo de la ciberseguridad. El proyecto ha sido diseñado para abordar la falta de recursos educativos accesibles que combinan teoría y práctica en un formato amigable, entretenido y estructurado. Se han creado diversos retos, diseñados para escalar en dificultad y abarcar un amplio espectro de habilidades y técnicas, permitiendo de este modo comprender varios conceptos a nivel superficial en lugar de profundizar sobre uno único.

Cada reto tiene asignado un vídeo de resolución mostrando cómo se resuelven y explicando detalladamente las vulnerabilidades explotadas. Todos los recursos están disponibles en una plataforma web específicamente diseñada para este proyecto.

Los retos implementados abordan desde el reconocimiento y análisis básico hasta técnicas más avanzadas entre las cuales podemos mencionar la inyección SQL, el manejo de scripts maliciosos y la explotación de configuraciones inseguras; incluso aparece un reto ligeramente diferente para trabajar con conceptos de redes y sus ataques. Cada reto ha sido meticulosamente diseñado para asegurar que los participantes no solo entendieran 'qué' herramientas usar, sino también el 'por qué' y el 'cómo', fomentando así un aprendizaje profundo y significativo.

PALABRAS CLAVE

CTF; ciberseguridad; aprendizaje; hacking ético; vulnerabilidad.

ÍNDICE

1. INTRODUCCIÓN

En el siguiente capítulo se mostrará el contexto donde se implementa el TFG, las causas que han fomentado su desarrollo, algunos datos y se presentará finalmente la idea sobre la que se fundamenta este proyecto. A continuación, en el apartado de estado del arte, se presentan algunas de las diferentes soluciones que conforman el ecosistema CTF.

1.1. Contexto

En el vasto y complejo dominio de la ciberseguridad, en conjunto con la naturaleza siempre cambiante de las amenazas digitales ha surgido una necesidad ineludible de educación continua y perfeccionamiento de habilidades en la materia.

Contrario a la noción ya obsoleta de que la seguridad por oscuridad es una estrategia suficientemente resolutiva, se ha hecho evidente que una comprensión detallada y profunda de las tácticas, técnicas y procedimientos (TTP) utilizados por los adversarios es fundamental para fortalecer las capacidades defensivas de cualquier organización o individuo. En este contexto, ha crecido exponencialmente la demanda de profesionales altamente capacitados que sean capaces de identificar, entender y contrarrestar vulnerabilidades en sistemas IT. Los datos indican que la ciberseguridad es un sector en pleno crecimiento, “El número de profesionales necesarios en ciberseguridad se elevaba a 63.191 empleos, mientras que en 2024 superará los 83.000” [REF1]

Esta transformación ha promocionado la ciberseguridad de un campo de interés nicho, a una disciplina de importancia crítica en el ámbito global.

Frente a este panorama, los retos de hacking y las competiciones de Capture The Flag (CTF) han emergido como herramientas educativas innovadoras, diseñadas para simular entornos vulnerables donde los participantes deben analizar y comprometer un sistema para mejorar sus habilidades prácticas. En el siguiente documento [REF2] se explora la efectividad de utilizar desafíos CTF como metodología didáctica para realizar pruebas de penetración y pentesting. La investigación concluye que los CTF son una herramienta educativa valiosa, permitiendo a los participantes identificar y corregir configuraciones incorrectas en sistemas expuestos a redes públicas, y proporcionando una comprensión práctica de las posibles acciones de un atacante y las medidas preventivas necesarias.

Con el constante descubrimiento de nuevas vulnerabilidades, estas competiciones CTF se han adaptado y expandido, ofreciendo desafíos que van desde la resolución de problemas básicos hasta la participación en simulaciones complejas y altamente realistas.

Estas actividades no solo han cultivado un creciente interés en el arte del hacking ético, sino que también han proporcionado una plataforma para el entrenamiento y una herramienta esencial para el desarrollo de las futuras generaciones de expertos en seguridad.

Este Trabajo de Fin de Grado (TFG) nace de la identificación de un vacío significativo entre los recursos disponibles para aquellos que buscan adentrarse en el estimulante mundo de los CTFs.

A menudo, los principiantes se enfrentan a desafiantes barreras de entrada, incluyendo la ausencia de guías integrales o un punto de partida definido que les oriente a través de su

aprendizaje. Aunque existen numerosas plataformas dedicadas a los CTFs, muchas de ellas funcionan esencialmente como colecciones de desafíos aislados, centradas sobre todo en la competitividad mediante un sistema de recompensas, pero sin ofrecer una estructura de aprendizaje progresivo o mecanismos que permitan mantener el interés y la motivación del usuario más novato.

Para llenar este vacío, este proyecto presenta el desarrollo de un CTF diseñado específicamente para un público objetivo, aquellos usuarios principiantes que pretendan iniciarse en este apasionante mundo. Este proyecto pretende proporcionar una serie de retos que aumentan gradualmente en dificultad y que abarcan un amplio espectro de temas relevantes en términos de hacking.

Para este fin, se ha diseñado y desarrollado una plataforma web personalizada que no solo facilita el acceso a los desafíos, sino que también integra recursos didácticos adicionales, como tutoriales en vídeo que sirven como guías paso a paso, para acompañar a los usuarios en su proceso de aprendizaje. A través de esta iniciativa, se aspira a ofrecer una experiencia educativa completa, interactiva y atractiva que inspire y motive a los principiantes a explorar con mayor profundidad el campo de la ciberseguridad.

En resumen, este TFG busca completar un hueco presente en el ecosistema, proporcionando un entorno completo y enriquecedor para todo aquel usuario que tenga intención de iniciarse en el apasionante mundo del hacking. A través de la diversidad de retos y el enfoque en la enseñanza, se espera que esta iniciativa inspire a futuros profesionales de la seguridad informática y contribuya a fortalecer la comunidad de ciberseguridad.

1.2. Estado del arte

Las plataformas de CTF han adquirido un papel fundamental como herramientas pedagógicas en el ámbito de la formación en ciberseguridad, proveyendo un medio interactivo y desafiante para que los entusiastas y profesionales pongan a prueba, practiquen y mejoren sus habilidades. La necesidad de recursos educativos efectivos, accesibles y adaptados a diferentes niveles de experiencia ha crecido de manera significativa y se han identificado algunas de las diferentes soluciones que se han desarrollado y publicado para hacer frente a esta demanda.

1.2.1. Hack The Box (HTB):

Hack The Box (HTB) es la plataforma por excelencia en la industria que ofrece una amplia gama de desafíos de ciberseguridad y laboratorios virtuales. Con una gran comunidad de usuarios, HTB se ha convertido en un punto de referencia para los entusiastas de la ciberseguridad. Proporciona máquinas virtuales que permiten a los participantes hacer uso de sus habilidades en entornos controlados. Cuenta con una enorme infraestructura que facilita a los usuarios con bajos recursos su participación.

En Hack The Box, existe un ranking según las puntuaciones obtenidas en los diferentes retos. Es un punto central para todos aquellos que se quieran hacer un nombre en el mundo del hacking.

1.2.2. TryHackMe:

TryHackMe se destaca por su enfoque en la enseñanza práctica de la ciberseguridad. Ofrece habitaciones temáticas y rutas de aprendizaje que guían a los principiantes a través de desafíos específicos. La plataforma también se centra en la accesibilidad y la comunidad, facilitando que los usuarios compartan conocimientos y experiencias. Si bien ofrece una enseñanza más guiada, el atractivo de los retos no llega a tener suficiente influencia como para que un principiante sin apenas conocimientos termine interesado.

1.2.3. PentesterLab:

PentesterLab se especializa en la seguridad web y las pruebas de penetración. Proporciona ejercicios prácticos y lecciones detalladas que ayudan a los usuarios a comprender y explotar vulnerabilidades web comunes. Es una excelente opción para quienes deseen profundizar en la seguridad de aplicaciones web. Al igual que en la anterior, la enseñanza es su objetivo primordial y sin duda ha sido uno de los grandes referentes para el desarrollo de esta plataforma, sin embargo, del mismo modo que la anterior, el entretenimiento está relegado a un segundo plano.

1.2.4. Root Me:

Root Me ofrece una amplia variedad de retos que abarcan múltiples categorías, incluyendo hacking web, criptografía, esteganografía y más. La plataforma permite a los usuarios mejorar sus habilidades en una variedad de disciplinas de seguridad informática. Su punto fuerte es la organización de los retos. Si bien los retos no suelen estar conexos, es una gran herramienta para aquellos usuarios que pretendan mejorar en un sector específico de la ciberseguridad.

1.2.5. VulnHub:

VulnHub es conocida por proporcionar máquinas virtuales vulnerables que los usuarios pueden descargar y explotar. Este sistema me pareció especialmente interesante para esta plataforma, a pesar de que requiere recursos por parte del usuario, es excelente a la hora de ayudar a comprender la vulnerabilidad explotada. La posibilidad de acceder a la misma teniendo la OVA permite tener una visibilidad mucho mayor sobre el reto enfrentado. Por otro lado, VulnHub ofrece una experiencia realista al simular escenarios del mundo real. Los retos varían en dificultad y abordan una amplia gama de temas de seguridad.

Estas plataformas representan solo una pequeña parte del ecosistema CTF. Su popularidad y diversidad reflejan la creciente necesidad de recursos accesibles para principiantes en el campo de la ciberseguridad. Cada plataforma tiene sus propias características únicas y enfoques pedagógicos que atraen a diferentes tipos de estudiantes y entusiastas. El presente Trabajo de Fin de Grado (TFG) se desarrolla en respuesta a esa brecha identificada en la oferta de CTF para principiantes con un enfoque en el entretenimiento y aprendizaje. Se busca proporcionar una plataforma personalizada que ofrece un camino claro y accesible para principiantes, respaldado por una experiencia envolvente, tematizada y educativa.

2. OBJETIVOS

En el siguiente capítulo se presentará el objetivo principal que motiva el desarrollo de este Trabajo de Fin de Grado. A continuación, se detallan los diferentes objetivos secundarios que se derivan del principal, explicando su relevancia y cómo contribuyen al éxito del proyecto.

2.1. Objetivo principal

El objetivo principal es desarrollar y validar una plataforma educativa interactiva tipo Capture The Flag (CTF) dirigida principalmente a principiantes en el campo de la ciberseguridad y el mundo del hacking ético. La plataforma deberá facilitar el aprendizaje y la práctica de habilidades básicas de ciberseguridad mediante retos que escalan en dificultad, ofreciendo a su vez recursos didácticos en formato de vídeo que guíen al usuario en la resolución de dichos retos.

2.2. Objetivos secundarios

El objetivo principal, presentado en el apartado anterior, se desglosa en múltiples objetivos secundarios que se detallan a continuación:

1. **Diseño de retos escalados en ciberseguridad**

El objetivo es diseñar retos que abarquen algunos de los fundamentos básicos hasta habilidades algo más avanzadas en ciberseguridad, sin olvidar cual es el público al que va dirigido, que es un usuario principiante. Cada desafío debe construirse sobre el anterior, implementando una ligera escalabilidad en su dificultad y permitiendo que los usuarios desarrollen sus habilidades de manera progresiva.

2. **Evaluación de la efectividad de la plataforma**

Se llevará a cabo un estudio para medir la efectividad de la plataforma en la mejora de las habilidades de ciberseguridad de los usuarios. Un grupo de usuarios pondrá a prueba la plataforma, entre ellos se encontrarán perfiles variados, desde un principiante hasta un experto. Este estudio incluirá preguntas para conocer el antes y después del uso de la plataforma, permitiendo valorar de este modo la efectividad de esta. Se pretende demostrar que, considerando que el público objetivo es un usuario principiante, el nivel de dificultad será suficientemente bajo para que estos usuarios puedan solucionarlo, mientras que un perfil de usuario más avanzado no le suponga un gran reto.

3. **Promoción de la conciencia en ciberseguridad**

Este objetivo busca aumentar la conciencia y el interés en la ciberseguridad a través de una plataforma accesible y atractiva para un público diverso. Se realizarán esfuerzos para garantizar que la plataforma sea fácil de usar y los retos sean accesibles para todos aquellos que deseen participar. España fue uno de los países con mayor afectación por ciberataques según [REF3], por lo que concienciar a un sector más amplio de la población ayudaría a crear políticas y procedimientos para combatir estos malos datos.

4. **Aplicar los conocimientos obtenidos en el grado de ingeniería de la ciberseguridad**

A lo largo de los aproximadamente cuatro años de estudios en el grado de Ingeniería de la Ciberseguridad, se han adquirido una amplia gama de conocimientos teóricos y prácticos. Este proyecto pretende poner en práctica esos conocimientos, permitiendo aplicar lo aprendido durante la carrera. Se espera que este proyecto no solo demuestre la competencia técnica del estudiante, sino que también refleje la capacidad para desarrollar soluciones innovadoras y efectivas en el campo de la ciberseguridad y la capacidad individual del alumno a la hora de implementar una plataforma tan compleja como la que nos concierne.

5. **Creación de una narrativa inmersiva**

Uno de los problemas principales identificados en otras plataformas CTF es la incapacidad de estas para desarrollar una historia envolvente que sirva como hilo conductor a través de los diferentes retos de la plataforma. Esta narrativa no solo captará la atención de los usuarios, sino que también incrementará su interacción con el contenido educativo. La historia del ladrón de arte puede ser utilizada para introducir conceptos de ciberseguridad de manera contextualizada, haciendo que los aprendizajes sean más significativos y memorables.

6. **Integración de elementos interactivos y multimedia**

Implementar una variedad de recursos multimedia que complementen y enriquezcan la experiencia educativa. Los vídeos no solo mostrarán la resolución de retos, sino que también incluirán elementos narrativos que aporten contexto y profundidad a la historia. Está demostrado, según [REF4] que la experiencia audiovisual ayuda a los usuarios a aprender. “Mediante la audición el 20% de la información que llega al receptor se convierte en conocimiento, un 70% lo hace a través de la visión y el 10% restante se lleva a cabo por los demás sentidos, de ahí que, al combinar varios de los sentidos el aprendizaje se realice de forma más rápida y efectiva”

7. **Implantación de la plataforma en contextos educativos formales**

Explorar la integración de la plataforma en contextos educativos formales, como escuelas o universidades. Esto podría incluir la adaptación de los retos y recursos para que sean utilizados como complementos en cursos de ciberseguridad. La idea de este proyecto es servir de inspiración o como referencia a otros futuros desarrollos en la materia.

3. METODOLOGÍA Y PLAN DE TRABAJO

En el siguiente capítulo se presentará la metodología utilizada para desarrollar la plataforma y se profundizará en las fases del desarrollo, explicando las actividades realizadas en cada una.

3.1. Metodología empleada

El desarrollo de la plataforma CTF para principiantes se estructuró utilizando una metodología de desarrollo en cascada, con ligeras variaciones. Esta metodología resulta particularmente efectiva para proyectos con requerimientos claramente definidos y etapas de desarrollo bien delineadas. Este enfoque metodológico permitió manejar de manera eficiente las dependencias entre las diversas etapas del desarrollo. Por ejemplo, es inviable grabar los vídeos de resolución sin haber implementado los retos o desarrollar la plataforma web sin conocer el contenido y formato de las flags.

Los problemas encontrados se añadieron en una bitácora; así, si en una fase posterior del proyecto apareciera un problema similar, la resolución sería mucho más sencilla, pues estaría debidamente documentada.

3.2. Fases del desarrollo del proyecto

El proyecto se ha desarrollado en cinco fases que se describen a continuación.

3.2.1. Fase I: Investigación y análisis

Inicialmente, el proyecto comenzó con una fase de investigación y análisis, en la que se realizó un estudio detallado de plataformas CTF existentes y se profundizó en el conocimiento de vulnerabilidades de seguridad clave. Se pretendía conocer en detalle una parte del ecosistema CTF para identificar debilidades y fortalezas que mejorar o implementar. En este punto, se detectó la inexistencia de recursos debidamente adaptados a principiantes. Todas estas labores de investigación fueron cruciales para recopilar ideas sobre cómo deberían ser los desafíos, prestando especial atención a que estos fueran tanto instructivos, como entretenidos y técnicamente realistas.

Esta etapa inicial fue clave para establecer las bases sólidas del proyecto. La investigación incluyó todo tipo de soluciones, desde las más populares a proyectos individuales presentes en GitHub. Se fueron tomando apuntes sobre las vulnerabilidades explotadas y como solían los retos ya existentes afrontar el factor entretenimiento.

Simultáneamente, personalmente realicé algunos de los retos para asegurar una comprensión profunda de las vulnerabilidades más comunes y su explotación, además de comprender cuál sería el mejor enfoque para la enseñanza de estos conceptos.

También sirvió de guía un repositorio [REF5] que contaba con una gran cantidad de write-ups detallados. También se obtuvo formación en la materia mediante el contenido audiovisual de algunos creadores de contenido del sector en YouTube. Sus videos me permitieron formarme en distintas áreas de ciberseguridad. Este conocimiento fue esencial para seleccionar las vulnerabilidades que se implementarían posteriormente en el proyecto y también ayudó a conocer como diseñar los desafíos.

3.2.2. Fase II: Desarrollo de los retos

Después de esta primera fase, el proyecto avanzó hacia el diseño y desarrollo de las máquinas virtuales, cada una configurada para simular diferentes entornos con problemas de seguridad.

Cada máquina virtual se diseñó para representar un nivel diferente de dificultad. Todas las maquinas compartirían Sistema Operativo, pero se configuraron distintas vulnerabilidades en cada una de ellas. Finalmente se repartieron de tal manera que termino quedando así:

- Reto 1: reconocimiento sin acceso a la máquina. Se explotarían puertos expuestos.
- Reto 2: acceso a la maquina sin escalado de privilegios. Alguna vulnerabilidad web y también se explotaría el SSH para lograr dicho acceso.
- Reto 3: acceso a la maquina con escalado de privilegios. Vulnerabilidades web, pero acceso distinto al SSH, por lo que se decidió implementar un LFI to RCE para obtener una Reverse Shell y posterior escalado de privilegios.
- Reto 4: escalado de privilegios. Un reto distinto al resto, el usuario ya tendrá acceso a la máquina y simplemente tendrá que escalar privilegios. Este reto está completamente centrado en ataques de red.

Esta fase fue especialmente compleja pues no solo debía conocer como explotar la máquina, sino que la debía habilitar para su explotación por parte de los participantes. Tuve que comprender mucho mejor el origen de la vulnerabilidad e investigar cómo podría implementarla en mis retos.

3.2.3. Fase III: Desarrollo de la web

Cuando las máquinas ya estaban en una etapa avanzada se diseñó y desarrolló una primera versión del sitio web para la verificación de flags, en este punto el desarrollo de los retos y la página web coincidieron y se retroalimentaron. Mientras desarrollaba el código web, hacía ciertas pruebas y modificaciones en las máquinas ajustando según que parámetros.

Una vez existía una primera versión estable de las máquinas virtuales, el desarrollo se centró en terminar de configurar el sitio web con toda su funcionalidad. El desarrollo del sitio web fue un proceso iterativo que involucró la creación de un frontend amigable y accesible a partir de una plantilla de Boostrap, así como un backend que aportase la funcionalidad necesaria. El diseño se centró en la usabilidad y en mantener una estética agradable a la vista.

Quizás la parte más relevante de este apartado fue la creación del Centro de Control. Se integró un panel de control donde los usuarios pueden ver su progreso en la recuperación de las obras de arte de una forma muy visual, incentivando así su continuo aprendizaje. Añadí posteriormente un mapa mediante la integración con una API para aportar a ese apartado gráfico.

3.2.4. Fase IV: Contenido multimedia

Los contenidos multimedia fueron diseñados no solo para instruir, sino para contar una historia que mantuviera a los usuarios comprometidos y emocionados por avanzar. El guion base sobre cuatro retos identificados con las distintas edades del hombre fue una idea que estuvo ahí desde el principio, sin embargo, faltaba algo que proporcionase al participante un objetivo claro. De esta necesidad surgió la idea del robo que sirvió como hilo conductor y estaba perfectamente alineado con la idea original.

Además de todo el contenido multimedia elaborado para generar narrativa y atractivo al proyecto, también se crearon los videos de resolución para dar cobertura a la parte educacional. Cada vídeo tutorial fue meticulosamente guionizado para asegurar que se comunicara los conceptos clave de manera clara y eficaz.

La creación de estos vídeos ha sido un apartado que ha llevado bastante trabajo, muchas horas dedicadas a la creación del guion, a la grabación y posteriormente a la edición.

3.2.5. Fase V: Pruebas

Tras este punto, el proyecto entró en una fase crítica de verificación y pruebas. Durante esta etapa, se realizaron pruebas exhaustivas para identificar y corregir fallos. Este proceso fue fundamental para garantizar que la plataforma no solo fuera operativa y segura, sino también que ofreciera una experiencia de usuario coherente y libre de errores

Tras llevar a cabo pruebas personalmente se contactó con un compañero con amplia experiencia en la resolución de CTFs para que comprobase en primera instancia que todo funcionaba como se esperaba. Tras dicha verificación, se llevaron a cabo algunas modificaciones según el feedback recibido y se estableció una segunda versión estable de las maquinas.

Tras esta primera fase de pruebas iniciales tanto por parte del desarrollador como por el tester, se envió la plataforma a un grupo selecto de usuarios para que la evaluaran en condiciones reales de uso. El objetivo era que estos usuarios valorasen la plataforma centrándose en tres principales capacidades: dificultad, aprendizaje y entretenimiento. Se solicitó que los participantes rellenasen un formulario que sirviese como estudio para sacar conclusiones sobre los objetivos fijados y conseguidos de este proyecto.

El feedback obtenido de esta prueba reveló varias áreas de mejora, pero lo más importante fue que indicó que la plataforma era funcional y su valoración fue más que notable. Siguiendo los principios de la metodología en cascada, algunos errores detectados se corrigieron sistemáticamente, generando así una tercera y definitiva versión de las máquinas. Esta etapa de ajuste post-pruebas subraya la importancia de la retroalimentación en el desarrollo de software y la necesidad de iteraciones dentro de un enfoque en cascada, aunque cada iteración se maneje de manera secuencial y controlada.

Como conclusión, este feedback no solo ayudó a mejorar la usabilidad y el contenido educativo, sino que también aseguró que la experiencia general fuera coherente y enriquecedora.

3.3. Cronograma

El desarrollo de todo este proyecto se ha extendido durante más de un año, dada la disponibilidad reducida y la compatibilidad con la vida laboral.

Por estas razones, se adjunta, en la figura 1, un diagrama de Gantt basado en la experiencia, pero simulando una jornada laboral de 40 horas semanales. Con esto se busca tener una fotografía más realista de lo que supondría desarrollar esta plataforma. El objetivo es representar mediante una estimación aproximada el proceso de desarrollo en una situación real dentro de una organización.

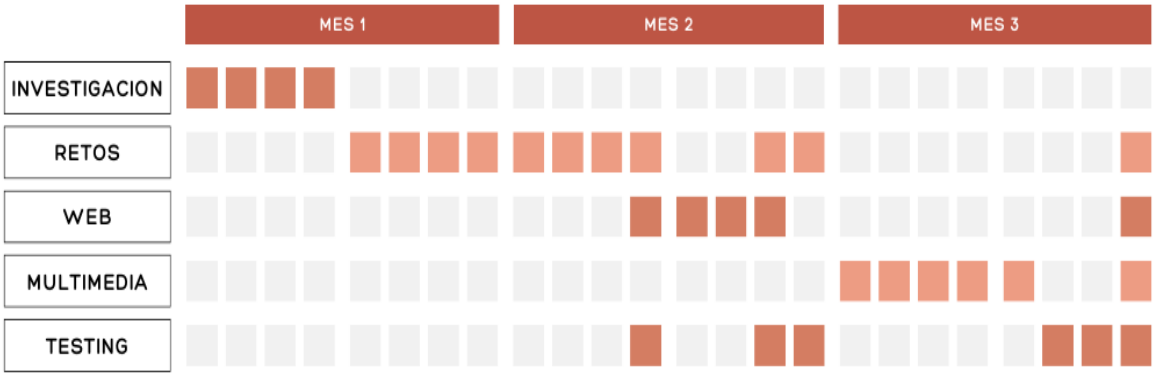

Figura 1 – Diagrama de Gantt

4. ANÁLISIS Y DISEÑO

En el siguiente capítulo se presentarán los distintos requisitos que se identificaron como indispensables para el desarrollo del proyecto. También se profundizará en cómo se ha diseñado la solución.

4.1. Análisis de requisitos

Previo al desarrollo de este TFG, fue necesario identificar una serie de requisitos. Aquí están listados todos los requisitos identificados.

4.1.1. Requisitos funcionales

En primer lugar, se listarán los requisitos funcionales los cuales definen las acciones o características principales que debe poseer un sistema de software para cumplir con el propósito previsto.

ID	Requisito	Descripción	Justificación
RF1	Plataforma Web Central	Debe existir una plataforma web que actúe como el punto central para acceder a la descarga de máquinas, vídeos y permita la verificación de flags. También existirá un centro de control.	Simplifica el acceso a los recursos, mejora la experiencia del usuario y otorga la funcionalidad de validación y consulta de una forma accesible para cualquier tipo de público.
RF2	Función de verificación de Flags	Debe existir una plataforma web un sistema que permita al usuario comprobar si la Flag obtenida es la correcta mostrando el resultado mediante un pop-up, actualizando el estado del reto a" completado" y actualizando el mapa del centro de control	Proporciona retroalimentación inmediata y visualización del progreso, mejorando la experiencia educativa y de juego. Se podrá consultar el porcentaje de progreso en todo momento en el centro de control.
RF3	Visualización de vídeos	La plataforma cuenta con una serie de vídeos presentes en YouTube, estos tienen que estar integrados en la plataforma web para ser accesibles por el usuario.	Facilita el acceso a los vídeos de narrativa y de resolución, haciendo que estos sean mucho más accesibles y utilizados por los usuarios
RF4	Botón de descarga de retos	Debe existir una plataforma web un sistema que permita al usuario descargar los retos mediante un botón	Facilita la descarga de los retos integrándolo en la propia plataforma y haciendo que sea un simple click
RF5	Centro de Control	La página web debe incluir un centro de control que	Proporciona un resumen visual e interactivo del progreso del

ID	Requisito	Descripción	Justificación
		muestre en tiempo real los retos completados, el porcentaje de atrapar al ladrón, un mapa con ubicaciones de las obras recuperadas y un botón para resolver el caso.	usuario, mejorando la inmersión y motivación durante el juego.
RF6	Listado de retos con estado actualizado	Los usuarios deben poder consultar el estado de los retos en el apartado del centro de control	Permite verificar que retos se han completado ya, con un simple vistazo
RF7	Mapa	Los usuarios deben poder consultar donde se han encontrado las obras de arte. Además, obtendrán una pista para el cuarto reto en el propio mapa	Permite tener una visión más gráfica de la situación geográfica además de ser una manera creativa de ofrecer al usuario una nueva pista
RF8	Botón de finalización del CTF	Los usuarios deben poder finalizar el CTF en cualquier momento. Se determinará si se ha conseguido superar o no usando un sistema de porcentaje de éxito que aumenta con cada flag resuelta y un mecanismo aleatorio que decide el resultado al presionar "resolver caso".	Permite una experiencia de juego flexible y autodeterminada, adecuada para usuarios con diversos niveles de habilidad y disponibilidad de tiempo.

4.1.2. Requisitos no funcionales

En segundo lugar, se listarán los requisitos no funcionales los cuales definen las cualidades, características y limitaciones del sistema

ID	Requisito	Descripción	Justificación
RNF1	Retos en formato OVA	Los usuarios deben poder descargar retos encapsulados en formato OVA que contengan todas las configuraciones y herramientas necesarias para resolverlos localmente.	Facilita el acceso a la plataforma en entornos con conectividad limitada y proporciona flexibilidad en el uso.
RNF2	Variedad en los retos	Cada reto debe presentar una ambientación y vulnerabilidad diferente, alineados con el hilo	Mantiene el interés y proporciona una educación más completa al explorar diversos aspectos de la

ID	Requisito	Descripción	Justificación
		narrativo central sobre el robo de obras de arte. El objetivo es que el participante pueda aprender y probar habilidades de ciberseguridad variadas.	ciberseguridad en contextos prácticos y atractivos.
RNF3	Complejidad adecuada para principiantes	Los retos deben ser accesibles para principiantes, enfocándose en vulnerabilidades y técnicas básicas. El reto debe resolverse sin requerir conocimientos muy avanzados o inasequibles para un usuario novato.	Facilita una introducción efectiva a la ciberseguridad para principiantes, promoviendo la acumulación gradual de habilidades.
RNF4	Escalabilidad de dificultad	Los retos deben incrementar su dificultad gradualmente, comenzando con conceptos básicos y culminando en un desafío más complejo en el cuarto reto.	Asegura una curva de aprendizaje adecuada que promueve la acumulación gradual de habilidades y mantiene la motivación del usuario. El objetivo principal es que el usuario pueda finalizar el reto y no desista, aunque tienen la opción.
RNF5	Uso de distintas herramientas de hacking	Los retos deben requerir el uso de herramientas de hacking habituales pero variadas tales como: Wireshark, Burp Suite, Nmap, John the Ripper, Hashcat, Metasploit, Hydra…	Asegura que los participantes se familiaricen con distintas herramientas y técnicas utilizadas en el ámbito de la ciberseguridad, ofreciendo una formación práctica y relevante para su futuro en la resolución de retos de hacking.
RNF6	No almacenamiento de datos personales	La plataforma debe operar de manera anónima, sin almacenar datos personales de los usuarios y sin requerir un sistema de login; el progreso debe almacenarse sólo en la caché local del navegador.	Aumenta la privacidad y seguridad de los usuarios, evitando riesgos de violaciones de datos y simplificando el acceso a la plataforma.
RNF8	Guías de resolución en vídeo	Debe haber vídeos disponibles que expliquen cómo resolver cada reto y las vulnerabilidades explotadas en cada uno de ellos. Los vídeos deben	Facilita el aprendizaje autodidacta y mejora la accesibilidad de la plataforma para principiantes, proporcionando una herramienta educativa clara y

ID	Requisito	Descripción	Justificación
		ser accesibles desde la página web.	efectiva para el público objetivo
RNF9	Narrativa común y contextualización de retos	Todos los retos deben estar integrados dentro de una narrativa común. Se ha decidido que todo girará en torno al robo de unas obras de arte y su recuperación. Cada reto contará con introducciones que contextualicen cada obra relacionada con el reto.	Aumenta el compromiso y el interés al proporcionar un contexto adicional y un propósito claro para resolver los retos. No solo se estarán aprendiendo sino también se estarán entreteniendo

4.1.3. Vulnerabilidades o ataques por incluir en la plataforma

Por último, se describen las distintas vulnerabilidades o ataques que se pretenden incluir en los retos. Su selección ha sido influenciada a su vez por algunos de los requisitos previamente comentados.

ID	Vulnerabilidad/ataque	Descripción	Justificación
V1	Puertos abiertos con login anónimo	Retos que incluyen puertos abiertos permitiendo login anónimo.	Enseña la identificación y las implicaciones de configuraciones de red inseguras.
R2	SQL Inyección (SQLi)	Retos que presentan vulnerabilidades de inyección SQL en aplicaciones web.	Capacita a los usuarios en el reconocimiento y explotación de vulnerabilidades comunes en aplicaciones web.
R3	Fuerza bruta	Retos que requieren ataques de fuerza bruta para descifrar contraseñas.	Ayuda a entender la importancia de usar contraseñas seguras y cómo proteger sistemas contra ataques de fuerza bruta.
R4	Local File Inclusion (LFI)	Retos con vulnerabilidades de inclusión de archivos a la maquina víctima.	Enseña cómo los atacantes pueden explotar aplicaciones web para introducir archivos potencialmente peligrosos
R5	Remote Code Execution (RCE)	Retos que permiten la ejecución de código remoto en servidores vulnerables.	Demuestra la severidad de las vulnerabilidades de ejecución de código, en sincronía con el LFI previamente explicado.
R6	Path hijacking	Retos relacionados con la manipulación de rutas de archivos o variables.	Enseña la importancia del manejo de permisos sobre archivos y cómo las aplicaciones pueden ser

ID	Vulnerabilidad/ataque	Descripción	Justificación
			manipuladas para comprometer la seguridad.
R7	Sniffing	Retos que implican la interceptación y análisis de tráfico de red.	Fomenta la comprensión de la seguridad de la red y las técnicas para proteger la transmisión de datos.
R8	Spoofing	Retos que involucran engañar a sistemas o usuarios con tráfico desde una fuente falsa.	Enseña las tácticas de engaño utilizadas por atacantes y cómo detectar y prevenir estos ataques.

4.2. Diseño

En el siguiente capítulo se define la arquitectura del sistema y se especifican los componentes que lo conformarán, así como la manera en que interactuarán entre sí.

4.2.1. Diagrama arquitectura de la plataforma

En la figura 2 se adjunta un pequeño diagrama que representa el diseño previo sobre cómo sería la plataforma.

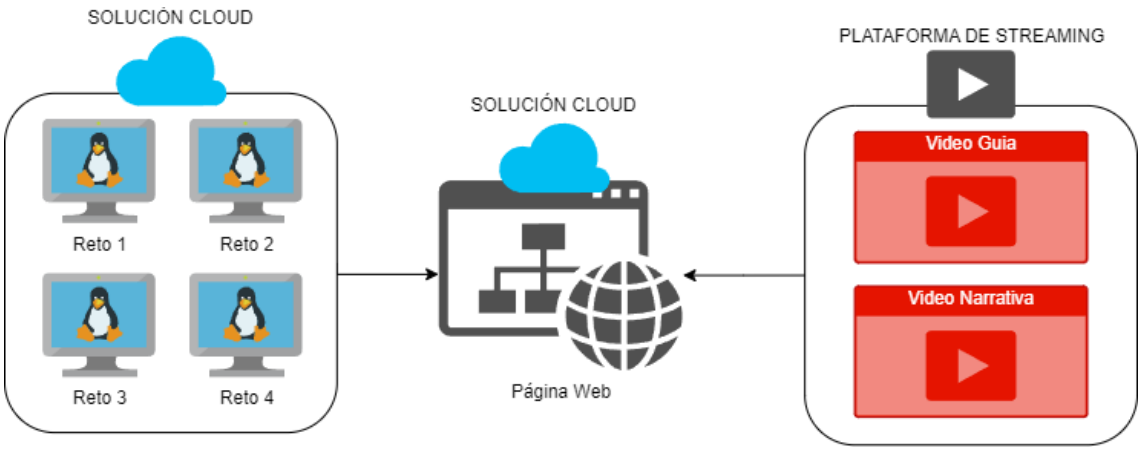

Figura 2 – Diagrama diseño de la plataforma

Como se puede apreciar, se compone de tres elementos principales:

- La página web servirá como punto central de la plataforma. De este modo, el usuario únicamente necesitará tener acceso a la misma para obtener todo lo necesario para realizar los retos. Se utilizará una solución cloud para su alojamiento.
- Las máquinas virtuales que contienen los retos estarán almacenadas en la nube. Debería existir un acceso directo para la descarga de estas, integrado en la propia página web.
- Los videos de resolución estarán subidos a alguna plataforma de streaming. A su vez, estarán integrados en la página web, haciéndolos más accesibles para el usuario.

4.2.2. Diseño de la interfaz web

La plataforma web está diseñada para ser el único punto de interacción para los usuarios, ofreciendo un acceso directo y sencillo a todas las funcionalidades necesarias. Su diseño persigue presentar una interfaz unificada que incluye secciones claramente definidas para cada reto.

La página web no implementará ningún sistema de login o registro. Los usuarios podrán acceder y participar en los retos sin proporcionar información personal, asegurando su anonimato. De este modo, se evita que aquellas personas reticentes a entregar sus datos se abstengan de participar en el reto.

Todo el progreso del usuario en los retos, incluidas las flags obtenidas y los estados de los retos, se almacenará únicamente en la caché local del navegador. Esto significa que la información solo estará disponible durante la sesión activa y se borrará automáticamente cuando el usuario cierre el navegador.

El contenido de la página web será el siguiente:

- Se integrarán en la plataforma dos tipos de vídeos: los narrativos estarán integrados dentro de la propia plataforma, mientras que los de resolución serán redirecciones a YouTube.
- La sección de descargas está diseñada para permitir a los usuarios obtener fácilmente las máquinas virtuales necesarias para cada reto. Estos archivos OVA estarán almacenados en la nube y serán accesibles mediante enlaces directos para su descarga.
- La plataforma incluye un sistema de verificación de flags interactivo, donde los usuarios pueden ingresar las flags obtenidas durante los retos. Este sistema verifica inmediatamente la corrección de las entradas y proporciona feedback instantáneo según sea acierto o error.
- Existirá además una sección aparte denominada centro de control, que mostrará el progreso del usuario en tiempo real. En dicha sección, habrá un listado con los retos, indicando cuáles han sido resueltos y cuáles aún están pendientes, una barra de progreso y un mapa. Cada reto completado actualizará automáticamente su etiqueta, la barra de progreso y el mapa con la ubicación donde se ha encontrado la obra de arte. Finalmente, incluirá un botón para "resolver" el caso, permitiendo a los usuarios finalizar el CTF cuando lo deseen, jugando con el azar si se decide finalizar el CTF sin completar todos los retos.

En las figuras 3 y 4 se muestra un esquema representativo de la idea sobre cómo será la interfaz web.

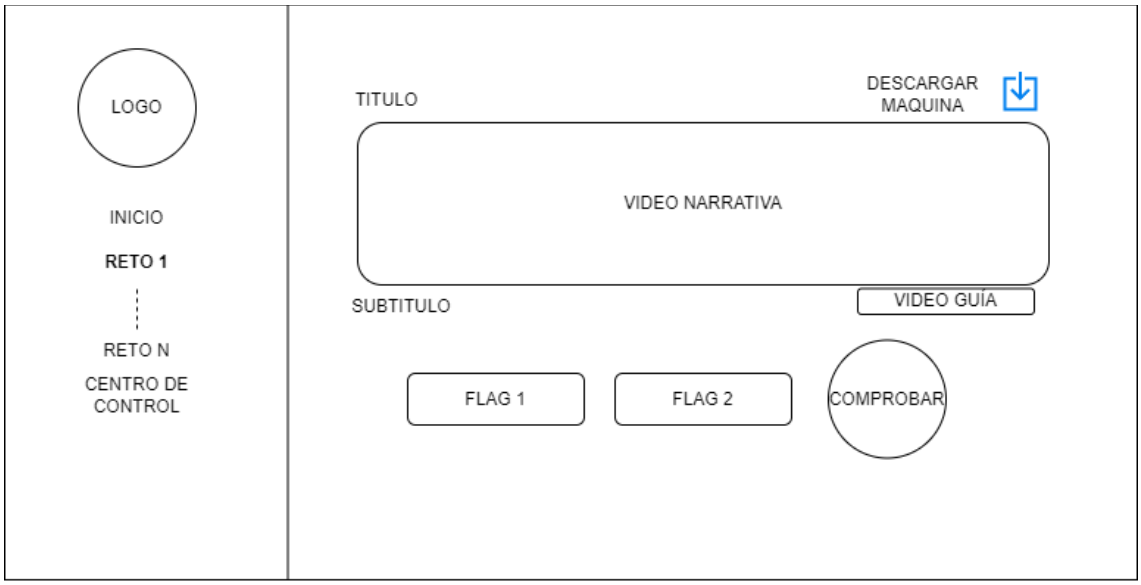

Figura 3 – Esquema interfaz de reto

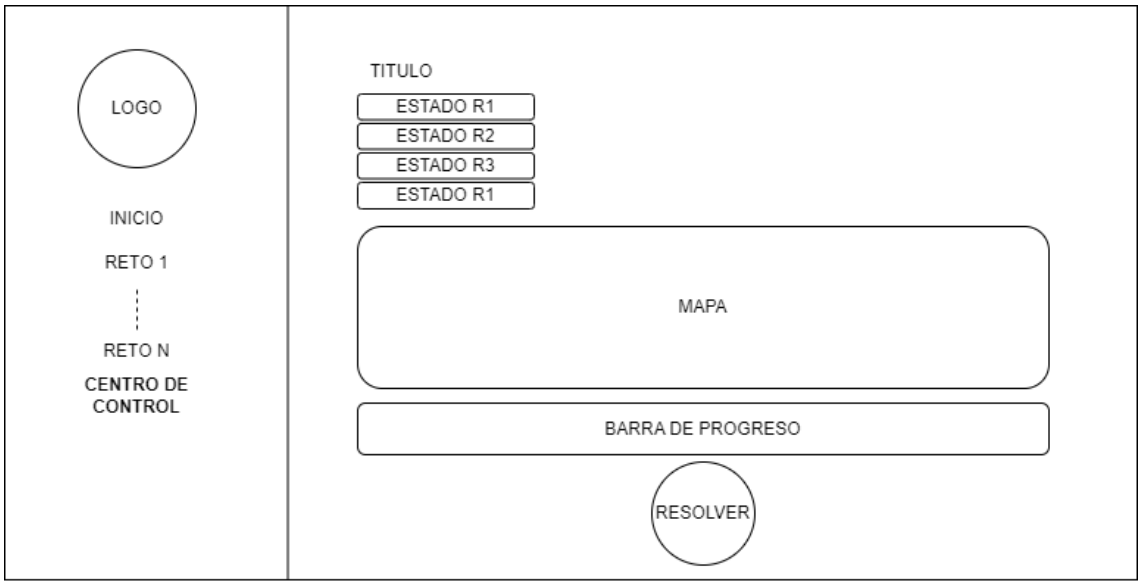

Figura 4 – Esquema interfaz del centro del control

4.2.3. Diseño de la narrativa

La narrativa de este CTF se construye en torno al robo de cuatro obras de arte: la venus de Willendorf, el ojo de Guiza, las cartas náuticas de Colón y la Mona Lisa. Existirá un primer vídeo introductorio en el que se presenta el reto, el cual trataría de lo siguiente: todos los noticieros del mundo abren con la noticia de última hora informando sobre el robo, pero la señal se ve interrumpida por lo que parece ser el ladrón de las obras de arte. Este propone a todos los oyentes que intenten resolver cuatro retos de hacking para poder recuperar las obras de arte y, finalmente, capturarle.

Cada reto estará temáticamente vinculado a una obra de arte específica. Además de ese primer vídeo introductorio, cada reto contará con un vídeo introductorio sobre la obra de arte robada.

Siempre que ha sido posible se han intentado incluir imágenes y referencias culturales o históricas pertenecientes a la época de la obra o relacionadas con la historia del robo dentro del propio reto. Esto no solo enriquece la experiencia visual y educativa, sino que también ayuda a los usuarios a sentir una conexión más profunda con el reto, aumentando su interés y compromiso con el mismo.

4.2.4. Diseño educativo de los retos

Otro aspecto importante que considerar en el diseño de los retos son los vídeos de resolución. Cada reto tiene en la plataforma un botón que redirige al usuario a un vídeo de YouTube de resolución del reto. En estos vídeos, no solo se resuelve el reto, sino que en ellos se explica la vulnerabilidad explotada, la técnica de hacking empleada y por supuesto las herramientas utilizadas para la explotación. Los usuarios tienen libertad de utilizar las herramientas que prefieran, pero cada reto está diseñado para resolverse con las recomendadas en las guías, garantizando que los usuarios sigan las instrucciones paso a paso para alcanzar la solución.

Es importante recalcar que en estos vídeos se explican las técnicas y el razonamiento detrás de cada paso, pero en ningún momento se revelan las flags reales para la resolución del reto, forzando de este modo a que los usuarios tengan aplicar lo aprendido para descubrir las soluciones por sí mismos. Se pretende de este modo, que la resolución de los retos sirva como educación más allá de la satisfacción de resolver un desafío.

4.2.5. Diseño técnico de los retos

Se realizó una investigación exhaustiva previa al desarrollo para identificar vulnerabilidades fundamentales en el aprendizaje de la ciberseguridad pero que no requieran un nivel avanzado de conocimientos técnicos para explotarlas.

Cada reto ha sido diseñado para enseñar una lección específica o una habilidad de hacking ético, siempre orientado a ser algo entendible para un principiante. Esto incluye proporcionar pistas y guías dentro del reto, que pueden ayudar a los usuarios a comprender y aprender a resolverlos sin llegar a un punto de frustración excesiva.

En la siguiente tabla se muestran cómo se reparten las distintas vulnerabilidades o ataques entre los cuatro retos diseñados, así como las herramientas propuestas para su resolución.

Reto	**Vulnerabilidad o ataque**	**Herramienta**
Reto 1	Puertos Abiertos con Login Anónimo	Nmap
Reto 2	SQL Injection (SQLi)	Dirbuster
	Fuerza Bruta	Hydra
Reto 3	Local File Inclusion (LFI)	Burpsuite
	Remote Code Execution (RCE)	PHP, Bash
	Path Hijacking	Bash, C
Reto 4	Sniffing	Wireshark
	Spoofing	Ettercap

A continuación, se presenta el diseño de cada uno de los retos describiendo su objetivo principal:

- **Objetivo del Reto 1**

El primer reto, "La prehistoria", está diseñado como una introducción para los usuarios más novatos, buscando enseñarles la importancia de la fase de reconocimiento en la resolución de un reto de hacking. Los participantes deben utilizar NMAP, una herramienta esencial para el escaneo de red, para identificar servicios vulnerables y recolectar las primeras flags.

- **Objetivo del Reto 2**

El objetivo de este reto es triple: primero, familiarizar al usuario con herramientas de automatización y técnicas de reconocimiento en entornos web; segundo, enseñar al usuario cómo identificar y explotar vulnerabilidades de inyección SQL en aplicaciones web; tercero y último, aprender a usar herramientas de fuerza bruta con diccionarios de contraseñas, para lo que se propone el uso de hydra. Este reto es esencial para entender cómo identificar y explotar las vulnerabilidades en escenarios reales, proporcionando a los usuarios habilidades prácticas que pueden aplicar en futuros retos e incluso medidas de seguridad personales como el uso correcto de contraseñas.

- **Objetivo del Reto 3**

El objetivo de este reto es desafiar al usuario a utilizar técnicas de scripting y explotación web para engañar al servidor mediante la subida de un archivo malicioso. Una vez que el archivo está subido, el usuario debe utilizarlo para obtener acceso no autorizado a la máquina víctima, en otras palabras, el archivo malicioso devolverá al atacante una Reverse Shell y podrá así obtener la primera flag. Finalmente, una vez que el usuario haya conseguido acceso este tendrá que escalar sus privilegios para recuperar una segunda flag escondida en el sistema con privilegios superiores.

- **Objetivo del Reto 4**

El cuarto reto desafía a los participantes a analizar y manipular activamente un script, usando habilidades de sniffing para comprender su comportamiento y spoofing para engañarlo. Además, este reto incorpora un componente de esteganografía y criptografía, donde los participantes deben recuperar el secreto escondido en una imagen y posteriormente descifrar dicho secreto para recuperar la flag oculta. Este reto se ha pensado para evaluar habilidades muy diversas tales como: análisis de tráfico, técnicas de spoofing y esteganografía junto a criptografía.

5. IMPLEMENTACIÓN

En el siguiente capítulo se explicará al detalle todo lo que se ha implementado, haciendo referencia a las herramientas o tecnologías utilizadas para el desarrollo y explicando cada aspecto técnico al detalle.

5.1. Tecnologías empleadas para el desarrollo de la web

El principal framework de desarrollo fue **Visual Studio Code**. VSC es una herramienta gratuita muy utilizada por desarrolladores, dada su capacidad de adaptación a distintos lenguajes y sistemas operativos. En este caso, se ha utilizado VSC para el desarrollo de todo el código que conformará la página web. Uno de los motivos por los que se eligió esta herramienta es debido a su reconocido prestigio, está muy extendida en el ámbito de la programación y cuenta con la confianza de un gran número de desabolladores en todo el mundo. Además, los lenguajes sobre los que se programará la página web están debidamente integrados en la herramienta, lo que facilita considerablemente la creación del código.

Para el desarrollo del frontend de la plataforma de CTF, se seleccionó una plantilla de **Bootstrap** [REF6] para el código **CSS** y **HTML**. Dado que el objetivo principal de este TFG no es el desarrollo web, pero es uno de los pilares sobre el que se sustenta, se decidió que el uso de una plantilla de Bootstrap sería idóneo como punto de partida para, a continuación, personalizarla y añadir la funcionalidad necesaria. Se eligió Bootstrap y no otras soluciones, por su flexibilidad y adaptabilidad, esencial para crear interfaces de usuario dinámicas y atractivas de forma sencilla. La elección de Bootstrap también se apoyó en asegurar que la plataforma sea accesible y funcional en una amplia variedad de dispositivos, desde teléfonos móviles hasta ordenadores. La posibilidad de usar dispositivos móviles se ha tenido en cuenta para el desarrollo, a pesar de no haber sido considerada como requisito.

En el backend, se optó por utilizar **JavaScript** para lograr una integración fluida con las tecnologías del frontend. Sobre un script de JS se ha desplegado toda la lógica de aplicación web. Al no requerir almacenamiento de datos persistente, JavaScript permitió el manejo efectivo del estado de la aplicación directamente en la caché, evitando la necesidad de bases de datos, el manejo de datos personales de los participantes y simplificando la arquitectura del sistema. Este script de JS ha sido ofuscado mediante el uso de la herramienta online obfuscator.io [REF7] para evitar que los participantes tuviesen acceso de una forma sencilla a la lógica de aplicación.

La plataforma está alojada en una instancia de **Google Cloud Platform**, aunque inicialmente comenzó en **Oracle Cloud Infrastructure** (OCI). La transición a GCP estuvo influenciada principalmente por mi experiencia con la herramienta, lo que permitió aprovechar las capacidades y herramientas robustas de Google para el hosting además de facilitar la gestión con una interfaz más intuitiva que la ofrecida por Oracle. La instancia ejecuta un Ubuntu Server con Apache2, configurado para servir la aplicación web, lo que proporciona una base sólida y segura para el alojamiento de la plataforma.

A día de hoy, la escalabilidad no es un problema, pues la plataforma no ha sido promocionada y su acceso ha estado muy acotado. Únicamente los miembros de testing

han podido hacer uso de ella. En un proyecto real las capacidades que te ofrece una solución cloud como la de Google permitirían que la página web no sufriese problemas de disponibilidad. Las maquinas que contienen los retos aún se almacenan en OCI, pues el coste de su almacenaje supone un ahorro respecto a su homónimo en Google Cloud.

5.2. Implementación de la web

5.2.1. Frontend

A pesar de utilizar una plantilla, se tuvieron que realizar muchas modificaciones sustanciales sobre el código frontend, para adaptar el diseño a los requisitos específicos del CTF. Si bien la plantilla sirvió como base, esta modificación era indispensable para crear un proyecto más personal y adaptado a las necesidades de este. A continuación, se presentarán las diferentes secciones que componen el sitio web.

Sección de inicio: La sección inicio cuenta con el título de la plataforma de CTFs, "BlockedCTF", y el nombre del reto concreto que se va a realizar, "Cronos". Además, existe una redirección hacía el LinkedIn del creador. Esta sección inicio sirve como punto de partida, por lo que aquí se encuentra el video de introducción al reto donde se presenta la narrativa y el objetivo a cumplir. En la figura 5 se puede ver una captura de esta sección.

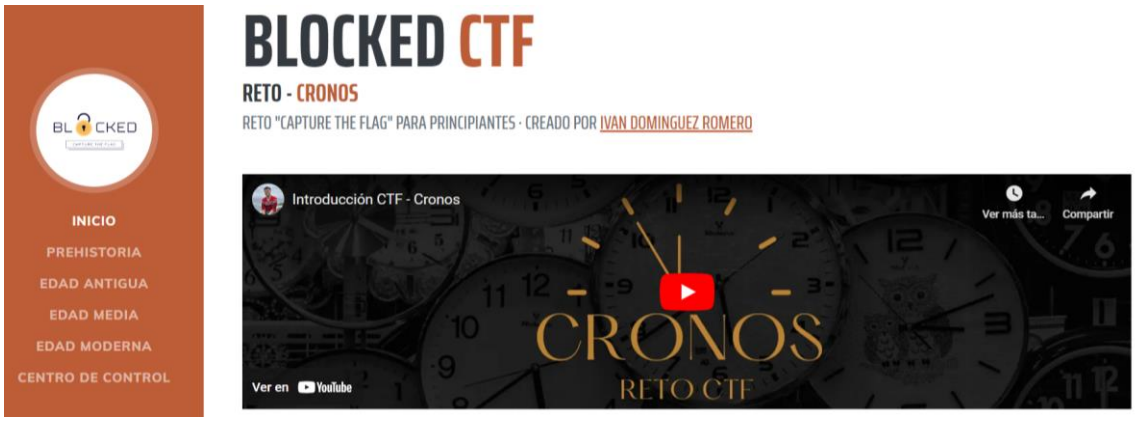

Figura 5 – Captura sección inicio

Sección de reto: Cada sección de reto está diseñada para albergar todo lo necesario para realizar el reto: vídeos, descarga de maquina y sistema de comprobación. En la figura 6 se muestra como se ve en la web una de las secciones de reto.

Figura 6 – Captura sección reto

En cada sección de reto, existe un botón de descarga que funciona como un enlace directo para la descarga de la OVA. Esto facilita a los usuarios el proceso de obtener y comenzar a trabajar en los retos, proporcionando un método de acceso simple, directo y centralizado.

En segundo lugar, como se ha explicado en la sección de requisitos, existe la necesidad de adaptabilidad a recursos audiovisuales. Estos vídeos están alojados en YouTube y han sido estratégicamente ubicados dentro de la interfaz de cada reto de la siguiente forma:

1. Los vídeos narrativos quedan completamente integrados en la propia página web. Se añadieron directamente a través de iframes de YouTube.
2. Por otro lado, los vídeos de resolución del reto se tomó la decisión de ofrecerlos como una redirección al propio YouTube. La razón es evitar que estos se utilicen como primer recurso, sino solo en caso de necesidad.

Por último, se encuentra el sistema de comprobación, este espera recibir dos parámetros que serán las flags del reto para llamar a la función de comprobación de coordenadas que se explicará en el siguiente apartado.

Centro de control: En este punto se buscó introducir elementos propios muy visuales tales como el mapa o la barra de progreso. Prácticamente toda la funcionalidad del centro de control se configura en el backend por lo que se explicará en el siguiente apartado. En la figura 7, se muestra cómo se vería el centro de control tras haber completado los tres primeros retos a falta del cuarto.

Figura 7 - Sección centro de control

5.2.2. Backend

A continuación, se van a explicar las diferentes funcionalidades que conforman la lógica de la web.

Comprobación de Coordenadas

Esta funcionalidad es la base del reto, pues es fundamental para verificar si las coordenadas ingresadas por el usuario son correctas. A continuación, en la figura 8 se muestra un diagrama de flujo de dicha funcionalidad:

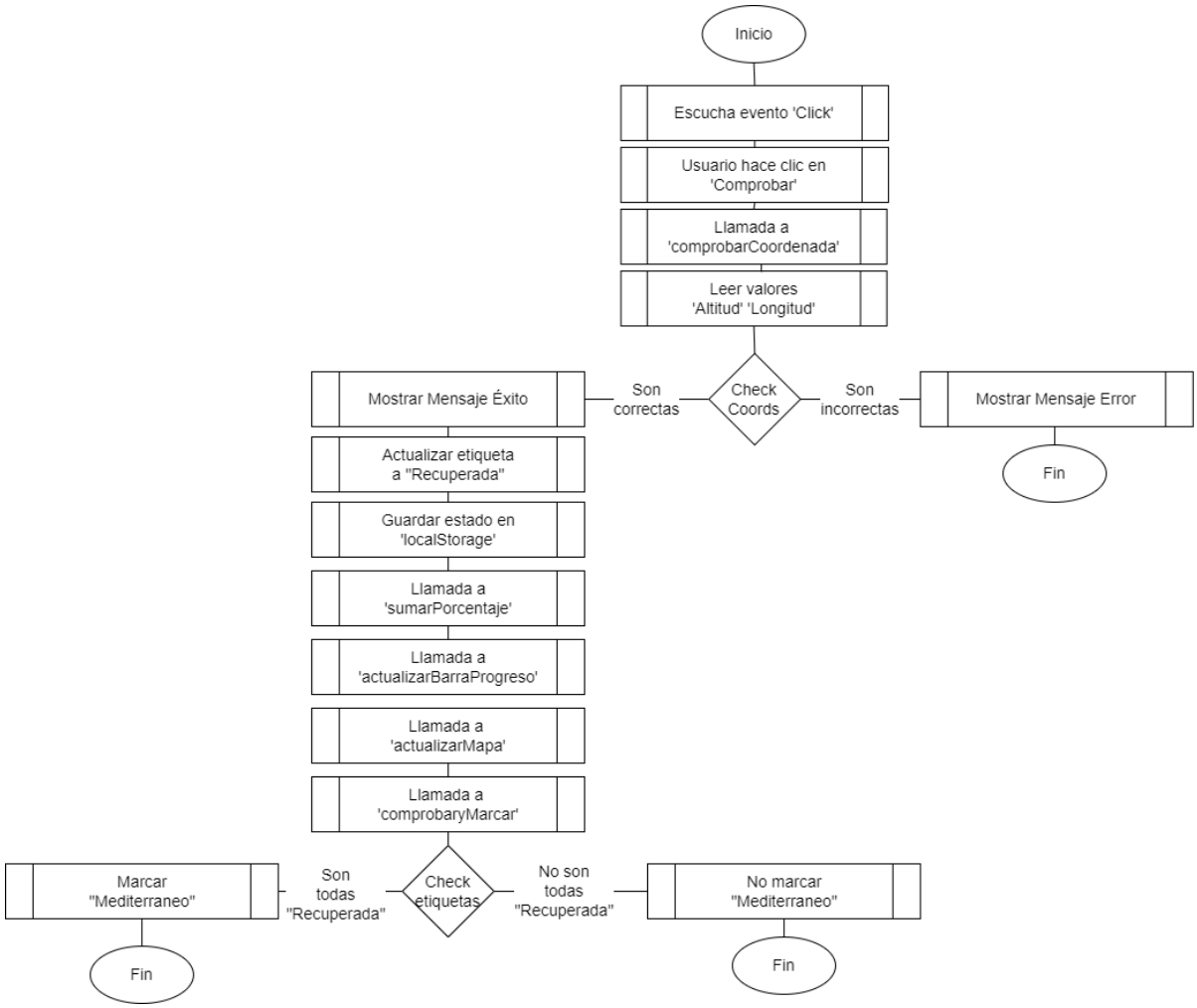

Figura 8 – Diagrama de flujo comprobación de coordenadas

A continuación, se van a mostrar y explicar las diferentes funciones que la componen y que están representadas en este diagrama flujo

```
////////////////////////////////////////// FLAG CHECK ////////////////////////////////////////////////////

function comprobarCoordenada(latitudInputId, latitudCorrecta, LongitudInputId, LongitudCorrecta, elementId) {
    var latitud = parseFloat(document.getElementById(latitudInputId).value);
    var Longitud = parseFloat(document.getElementById(LongitudInputId).value);

    if (!isNaN(latitud) && !isNaN(Longitud) && latitud === latitudCorrecta && Longitud === LongitudCorrecta) {
        alert(`¡Coordenadas correctas (Latitud: ${latitud}, Longitud: ${Longitud})!`);

        var elemento = document.getElementById(elementId);
        if (elemento) {
            elemento.innerHTML = `<span class="badge badge-recuperada">Recuperada</span> ${elementId}`;
            localStorage.setItem(`${elementId}Status`, 'Recuperada');
            sumarPorcentaje(20);
        }
        actualizarMapa();
    } else if (!isNaN(latitud) && !isNaN(Longitud) && latitud === latitudCorrecta) {
        alert(`¡incorrecta! (Latitud: ${latitud}, Longitud: ${Longitud})!`);
    } else if (!isNaN(latitud) && !isNaN(Longitud) && Longitud === LongitudCorrecta) {
        alert(`¡incorrecta! (Latitud: ${latitud}, Longitud: ${Longitud})!`);
    } else {
        alert(`incorrectas (Latitud: ${latitud}, Longitud: ${Longitud}). Inténtalo de nuevo.`);
    }

}
```

Figura 9 – Función Comprobar Coordenadas

En la figura 9, se visualiza la función “comporbarCoordenada” que como su propio nombre indica, es la encargada de comprobar si los dos valores de coordenadas ingresados por el usuario son correctos. Los dos valores que espera recibir corresponden a las flags de cada reto, que los usuarios obtendrán tras completarlos.

```
////////////////////////////// BARRA DE PROGRESO ////////////////////////////////////
  var porcentajeTotal = 0;

  function sumarPorcentaje(porcentaje) {
      porcentajeTotal += porcentaje;
      actualizarBarraProgreso(porcentajeTotal);
  }

  function actualizarBarraProgreso(porcentajeTotal) {
      var barraProgreso = document.getElementById('barraProgreso');
      if (barraProgreso) {
          barraProgreso.style.width = porcentajeTotal + '%';
          barraProgreso.innerHTML = porcentajeTotal.toFixed(2) + '%';

          localStorage.setItem('FormularioPorcentaje', porcentajeTotal.toFixed(2));
      }
  }
```

Figura 10 - Función Actualizar Barra de Progreso

La figura 10 muestra la función que va a actualizar la barra de progreso, almacenando el nuevo porcentaje total en el almacenamiento local. Esto sirve para asegurar la persistencia del progreso a lo largo de la sesión del usuario a nivel de cache. De esta manera la plataforma gestiona el progreso del usuario en el juego

```
////////////////////////////// MAPA ////////////////////////////////////////////

function actualizarMapa() {
    for (let obra in coordenadas) {
        let estado = localStorage.getItem(`${obra}Status`);
        if (localStorage.getItem('VenusStatus') === 'Recuperada') {
            L.marker(coordenadas['Venus'], {icon: venusIcon}).addTo(map)
                .bindPopup('Venus recuperada');
        }
        if (localStorage.getItem('GuizaStatus') === 'Recuperada') {
            L.marker(coordenadas['Guiza'], {icon: guizaIcon}).addTo(map)
                .bindPopup('Guiza recuperada');
        }
        if (localStorage.getItem('ColonStatus') === 'Recuperada') {
            L.marker(coordenadas['Colon'], {icon: colonIcon}).addTo(map)
                .bindPopup('Colón recuperada');
        }
    }
    comprobaryMarcar();
}
```

Figura 11 - Función Actualizar mapa

Como se ilustra en la figura 11, "actualizarMapa" es la función que gestiona la visualización de los estados de las obras de arte en el mapa interactivo, marcando las obras recuperadas y actualizando en tiempo real según el progreso del usuario. Esta función utiliza Leaflet, una biblioteca de JavaScript para mapas interactivos. Cada obra de arte tiene, además, su propio icono personalizado.

```
////////////////////////////// PISTA ////////////////////////////////////////////

function comprobaryMarcar() {
    if (localStorage.getItem('VenusStatus') === 'Recuperada' &&
        localStorage.getItem('GuizaStatus') === 'Recuperada' &&
        localStorage.getItem('ColonStatus') === 'Recuperada') {


        const latitudMedia = (coordenadas.Venus[0] + coordenadas.Guiza[0] + coordenadas.Colon[0]) / 3;
        const longitudMedia = (coordenadas.Venus[1] + coordenadas.Guiza[1] + coordenadas.Colon[1]) / 3;


        L.marker([latitudMedia, longitudMedia]).addTo(map)
            .bindPopup('Mediterraneo')
            .openPopup();
    }
}
```

Figura 12 - Función Comprobar y Marcar

Basándose en el estado de las obras de arte, si el reto correspondiente ha sido resuelto correctamente, el estado pasa de "Desaparecida" que es el estado por defecto a "Recuperada". Esta función comprueba que los tres primeros retos han sido resueltos, es decir, si están con la etiqueta "Recuperada". Esto le otorgará al usuario una pista para la resolución del último reto a través del mapa previamente explicado.

Comprobación del secreto

Similar a la funcionalidad anterior, pero se utiliza para verificar el secreto correspondiente al cuarto reto. El cuarto reto es ligeramente diferente a los anteriores pues solo se trata de una única flag y su valor porcentual para el cálculo de la probabilidad de éxito es mayor. En la figura 13 se puede observar un diagrama de flujo que representa dicha funcionalidad:

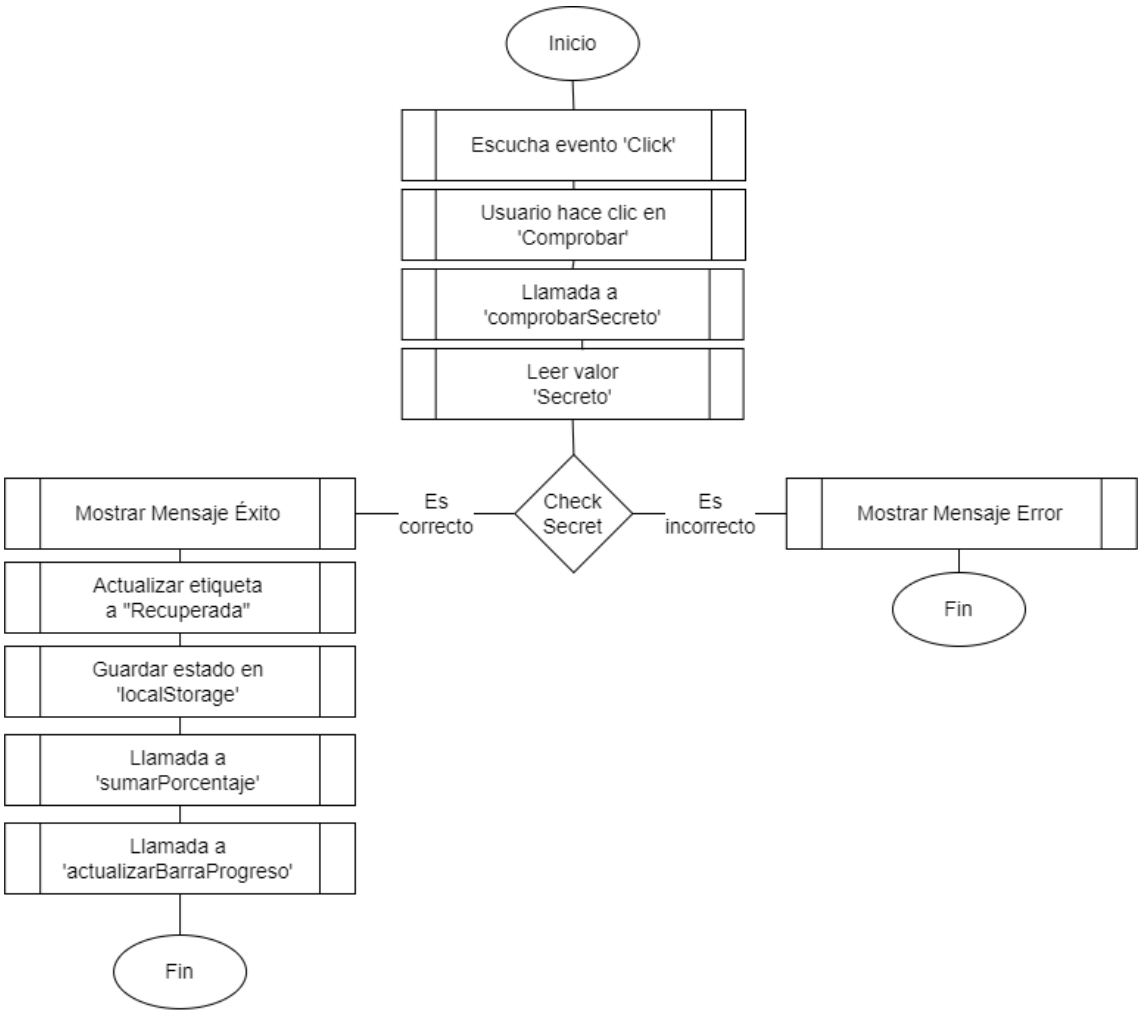

Figura 13 - Diagrama de flujo comprobación del secreto

A continuación, se van a mostrar y explicar las diferentes funciones que la componen y que están representadas en este diagrama de flujo. La función “actualizarBarraProgreso” también será utilizada aquí por lo que no se volverá a incluir.

```
////////////////////////////////////// SECRET CHECK //////////////////////////////////////

function comprobarSecreto(secretInput, secretocorrecto, elementId) {
    var secreto = document.getElementById(secretInput).value;

    if (secreto === secretocorrecto) {
        alert(`¡Secreto correcto (Secreto: ${secreto})!`);

        var elemento = document.getElementById(elementId);
        if (elemento) {
            elemento.innerHTML = `<span class="badge badge-recuperada">Recuperada</span> ${elementId}`;
            localStorage.setItem(`${elementId}Status`, 'Recuperada');
            localStorage.setItem(`${elementId}Acierto`, 'true');
            sumarPorcentaje(40);
        }
        actualizarMapa();
    } else {
        alert(`¡Secreto incorrecto (Secreto: ${secreto})!`);
    }
}
```

Figura 14 - Función Comprobar Secreto

La función representada en la figura 14 comprueba el valor secreto ingresado por el usuario. Este valor deberá corresponder a la última flag del CTF, obtenida tras resolver el reto de "La Edad Moderna". Si la flag es correcta, la función actualiza la etiqueta del reto y suma un 40% al porcentaje de resolución del caso, completando de este modo el 100%.

Resolución del Caso

Desde el inicio del CTF, los usuarios tienen acceso a un botón en el Centro de control que les permite finalizar el CTF en cualquier momento. Cada reto completado incrementa el porcentaje de éxito del usuario en la plataforma. Los tres primeros retos aumentan este porcentaje en un 20%, mientras que el reto final, que es más complejo, contribuye con el 40% adicional. Su funcionalidad está representada en el diagrama de la figura 15.

Figura 15 - Diagrama de flujo resolución del caso

A continuación, se van a mostrar y explicar las diferentes funciones que la componen y que están representadas en este diagrama de flujo.

```
//////////////////////////// RESOLVER CASO //////////////////////////////////////////

function comprobaryMarcar() {
    var porcentajeTotal = parseFloat(localStorage.getItem('FormularioPorcentaje'));

    porcentajeTotal = isNaN(porcentajeTotal) ? 0 : porcentajeTotal;

    var valorAlAzar = Math.random() * 100;

    if (porcentajeTotal > 0 && valorAlAzar <= porcentajeTotal) {
        window.location.href = 'assets/img/COMPLETADO.mp4';
    } else {
        window.location.href = 'assets/img/GAMEOVER.mp4';
    }
};
```

Figura 16 - Función Resolver Caso

Al presionar el botón para finalizar el reto se activa un mecanismo que genera un número aleatorio. Tal y como se muestra en la figura 16, a continuación, determina si el usuario ha acumulado suficiente porcentaje para resolver el caso con éxito, comparando el número aleatorio con el porcentaje acumulado. Dependiendo del resultado, redirige al usuario a un vídeo de éxito "COMPLETADO.mp4" o de fracaso "GAMOVER.mp4". Este sistema está diseñado para incentivar a los usuarios a completar todos los retos para maximizar sus probabilidades de éxito y su resolución ordenada, pero siempre ofreciéndoles la opción de retirarse cuando deseen.

5.3. Tecnologías empleadas para el desarrollo de las máquinas

Cada reto se ha montado sobre un SO **Ubuntu Server 20.04**, y ha sido configurado específicamente para incluir los desafíos y vulnerabilidades necesarios para el reto. Esto proporciona un entorno controlado y seguro para aplicar conocimientos de hacking sin comprometer la seguridad de sistemas externos. Una vez configuradas y probadas, estas máquinas virtuales se han exportado en formato **OVA** para que los usuarios puedan descargarlas e importarlas. El formato OVA es perfecto para este tipo de retos ya que encapsula toda la configuración de la máquina virtual, incluidos discos, memoria, redes y todas las configuraciones del SO, en un solo archivo fácil de manejar. Como se ha mencionado anteriormente, tras su exportación estos archivos se han almacenado en un bucket en la nube, proporcionado por Oracle, y se han establecido sus permisos para que sean públicos y accesibles por los participantes.

Las tecnologías que conforman el propio reto se explican a continuación.

5.4. Implementación de las máquinas

Se realizó una investigación exhaustiva previa al desarrollo para identificar vulnerabilidades fundamentales y algunas de las fuentes más influyentes están aquí referenciadas [REF8] [REF9] [REF10] [REF11] [REF12] y [REF13]

Cada reto incorpora una o varias vulnerabilidades específicas. Esto garantiza que los participantes deban emplear diferentes habilidades y herramientas de hacking para superar cada desafío. Cada reto está diseñado para ser resuelto con herramientas adecuadas a la vulnerabilidad presentada.

5.4.1. La Prehistoria

- **Definición del reto**

El desafío está centrado en la fase de reconocimiento donde, en primer lugar, el participante identifica la dirección IP de la máquina objetivo. Utilizando la herramienta NMAP, como se muestra en la figura 17, se lleva a cabo un escaneo que revela los puertos 21 (FTP) y 139 | 445 (Samba) abiertos. Además, el propio nmap destaca la capacidad de realizar un inicio de sesión anónimo en el servicio de FTP.

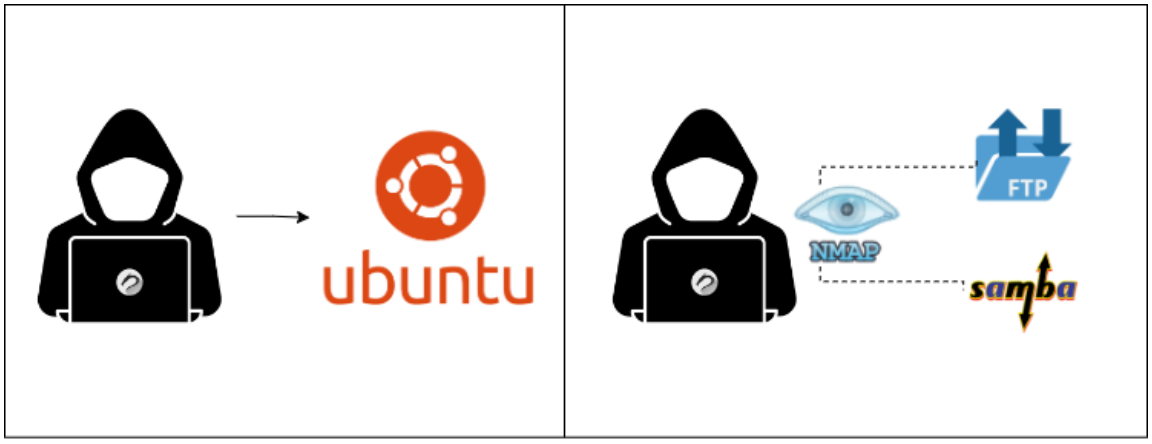

Figura 17 – Esquema fase de reconocimiento

Para la explotación del FTP, el participante tendrá que usar el cliente FTP que le permite conectarse al puerto 21. Cuando se le soliciten las credenciales, el participante tendrá que insertar el usuario "Anonymous", accediendo al sistema y localizando la primera flag. Esto se ha esquematizado en la figura 18. Tras descubrir la flag, al intentar acceder a otra carpeta, el acceso está bloqueado pues requiere permisos superiores, se tendrá que buscar otra solución.

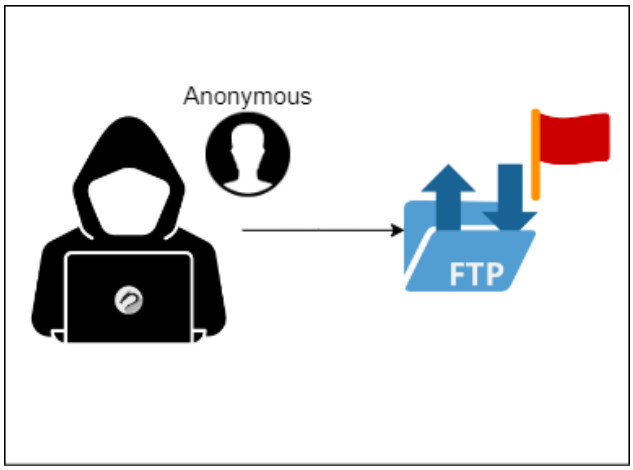

Figura 18 – Esquema explotación FTP

A continuación, el foco se traslada al servicio de Samba. Utilizando credenciales "guest", el participante se conecta usando smbclient y realiza una enumeración de las carpetas compartidas. Esta exploración le permite identificar y acceder a una carpeta específica donde se encuentra la segunda flag. Esta fase se muestra en la figura 19.

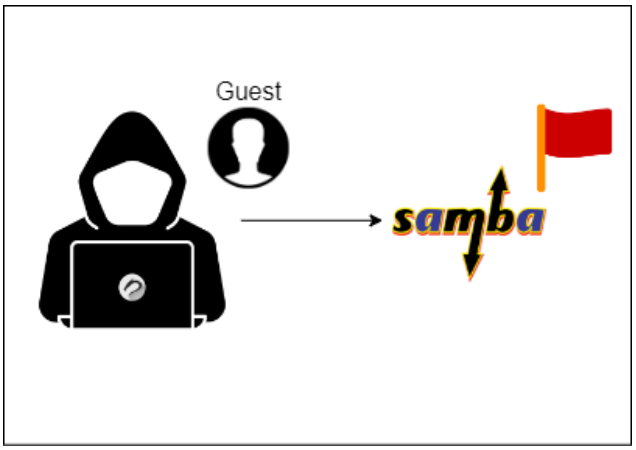

Figura 19 – Esquema explotación Samba

- **Vulnerabilidades**

Acceso anónimo a FTP: El puerto 21 está abierto y configurado para permitir el acceso anónimo. Esta vulnerabilidad es crítica porque permite a cualquier usuario acceder sin autenticación, lo que puede exponer archivos sensibles o permitir la carga de archivos maliciosos a la máquina víctima. Se pretende mostrar al participante la importancia de restringir el acceso anónimo, y se enseña cómo una mala configuración del servidor FTP puede ser un punto de entrada para ataques más severos o perdidas de datos sensibles. Esta configuración se implementa en el archivo "/etc/vsftpd.conf" como se muestra en la figura 20

```
  GNU nano 6.2                                  /etc/vsftpd.conf
# Allow anonymous FTP? (Disabled by default).
anonymous_enable=YES
anon_upload_enable=NO
anon_mkdir_write_enable=NO
anon_root=/home/myth
```

Figura 20 – Configuración FTP

Acceso SMB sin restricciones: Los puertos 139 y 445, asociados con el protocolo SMB, están configurados para permitir que cualquier usuario inicie sesión utilizando credenciales predeterminadas. Esta configuración expone a los sistemas a riesgos significativos. Los participantes exploran cómo se pueden explotar estas configuraciones para acceder a recursos de red compartidos y la importancia de implementar políticas de seguridad robustas en los servicios SMB. En este caso, la configuración insegura se encuentra en el archivo "/etc/samba/smb.conf" y se muestra en figura 21

```
  GNU nano 6.2                                  /etc/samba/smb.conf
[Compartido]
   comment = Carpeta Compartida
   path = /home/myth/FlagCoordenadaX
   browsable = yes
   guest ok = yes
   read only = no
```

Figura 21 – Configuración SMB

- **Herramientas Utilizadas en la resolución**

Netdiscover: se trata de una herramienta muy habitual en este tipo de retos, la cual permite el escaneo de redes en búsqueda de direcciones IP activas. Con los resultados de esta herramienta podremos identificar los hosts activos en una red determinada.

Nmap: Utilizado de manera intensiva para el reconocimiento inicial, nmap es una herramienta esencial no solo por su capacidad de detectar puertos abiertos, sino también por su potencial para identificar los servicios que se ejecutan detrás de esos puertos y sus versiones específicas. Los participantes aprenden a utilizar opciones y parámetros como -sV para la detección de versiones de servicio, -p para especificar rangos de puertos o la ejecución de scripts prestablecidos que pueden revelar vulnerabilidades de manera automática.

Ftp: es el acrónimo de "File Transfer Protocol", se trata de un protocolo que se utiliza para transferir todo tipo de archivos entre equipos conectados a una red. En este reto se usará el cliente de ftp para establecer la conexión al servicio de FTP de la maquina víctima. Es muy utilizado en este tipo de retos y sus comandos son muy similares a los de Bash.

Smbclient: smbclient permite acceder a los recursos compartidos de un servidor SMB, de forma similar al cliente FTP explicado anteriormente. Para la resolución del reto, se usará la herramienta smbclient para la conexión al servicio de Samba de la maquina víctima. SMB si tiene alguna variación respecto a los comandos de Bash.

5.4.2. La Edad Antigua

- **Definición del reto**

El proceso de resolución del desafío comienza con una fase de reconocimiento inicial, representada en la figura 22, donde el participante identifica la máquina objetivo y los posibles vectores de ataque, como se ha aprendido en el reto anterior. En este proceso, se utiliza la herramienta NMAP para escanear la máquina, revelando dos puertos abiertos: el puerto 80, operado por Apache, y el puerto 22, correspondiente a SSH.

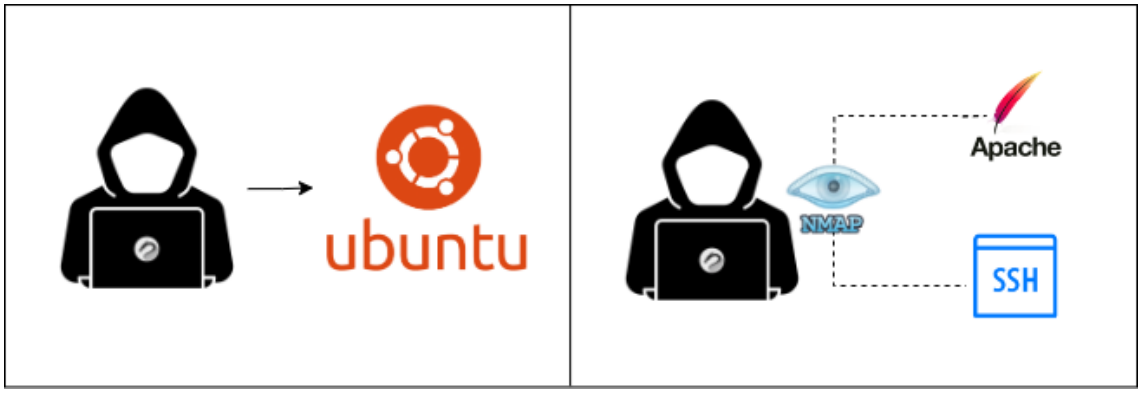

Figura 22 – Esquema fase de reconocimiento

El análisis inicial se centra en el puerto 80. Utilizando un navegador, el participante ingresa la dirección IP de la máquina objetivo y accede a una interfaz que, aunque no revela soluciones directas al desafío, proporciona pistas útiles para avanzar. Para explorar más a fondo, se recomienda el uso de herramientas de automatización como ffuz, wfuzz o dirbuster, representada en la figura 23, que permiten descubrir los directorios disponibles en el servidor web.

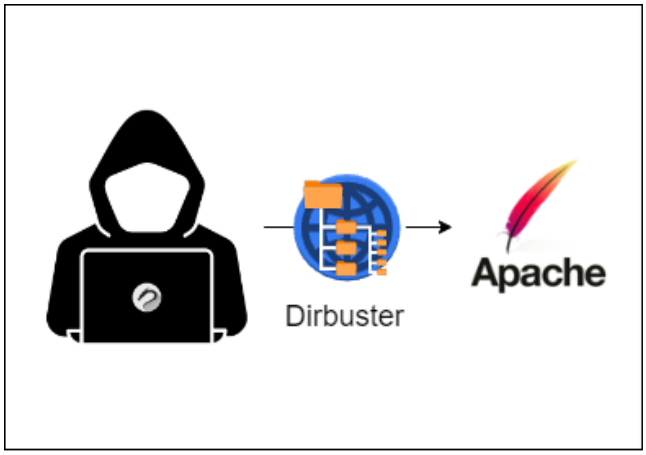

Figura 23 – Esquema uso de Dirbuster

A través de estas dirbuster (o similares), se identifica el archivo “access.php”, que contiene una interfaz de inicio de sesión. Utilizando las pistas halladas en la página principal, el participante intentará realizar un ataque de inyección SQL (SQLi) en el formulario de inicio de sesión. Como se puede observar en la captura de pantalla de la figura 24, una inyección SQL básica es suficiente para obtener acceso.

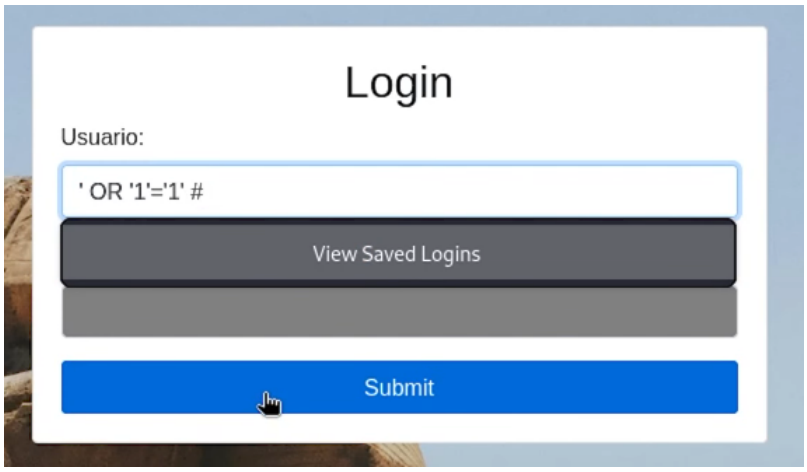

Figura 24 – Sentencia SQLi simple

Tras lograr la inyección SQL se nos redirige a “bienvenido.php”, donde parece que se encuentra la primera flag del desafío. Se trata de un engaño, pues al hacer click sobre el botón, se revela que está temporalmente deshabilitado, por lo que se tendrá que inspeccionar la página para encontrar, ahora sí, la primera flag.

Para la obtención de la segunda flag, el archivo “robots.txt” juega un papel crucial. Dicho archivo, comúnmente revisado durante las fases iniciales de los retos CTF para comprender la estructura del sitio y las tecnologías utilizadas, puede contener información sobre directorios o archivos que el administrador prefiere mantener fuera del alcance de los motores de búsqueda. En este desafío, el archivo robots.txt no da detalles estructurales, sino que su función se ha readaptado como pista, sugiriendo un nombre de usuario que puede emplearse en un ataque de diccionario. En la figura 25 se muestra el contenido de dicho archivo.

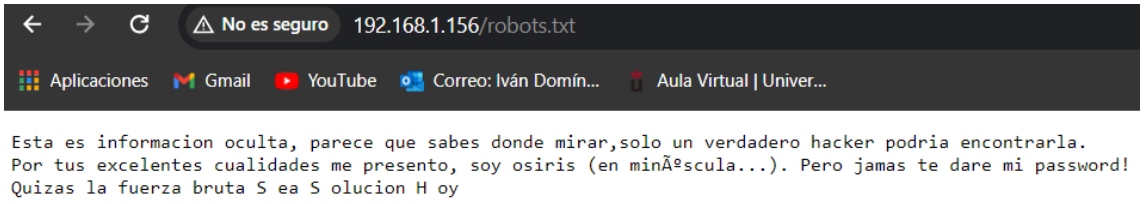

Figura 25 – Contenido del archivo “robots.txt”

Con esta información y las pistas obtenidas, el participante emplea la herramienta Hydra para lanzar un ataque de diccionario utilizando el nombre de usuario "osiris" y el diccionario “rockyou” contra el SSH de la máquina víctima. La figura 26 muestra una captura del comando y la salida.

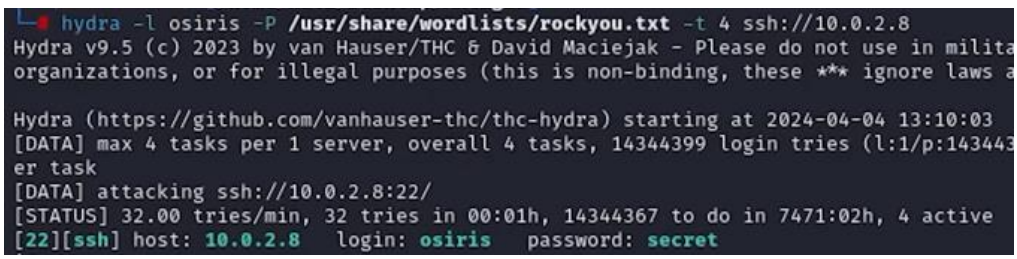

Figura 26 – Uso de Hydra para ataque de fuerza bruta

Este método representa un intento estratégico para superar las medidas de seguridad e infiltrarse en la máquina objetivo, culminando así la resolución del desafío de hacking al obtener la segunda flag.

- **Vulnerabilidades Exploradas**

SQL Injection en formulario de login: La explotación de la vulnerabilidad de inyección SQL en access.php permite al usuario simular o inyectar comandos SQL que el servidor ejecuta como si de una solicitud real y legítima se tratase. Esto puede resultar en la elusión de autenticaciones normales y la exposición de datos sensibles. El origen de dicha vulnerabilidad es una mala configuración a la hora de hacer el parseo anterior a la consulta a la base de datos SQL como se muestra en la figura 27

```php
<?php
$conexion = new mysqli('localhost', 'Myth', 'antigua', 'usuarios');


if ($conexion->connect_error) {
    die("Error de conexión: " . $conexion->connect_error);
}


if ($_SERVER["REQUEST_METHOD"] == "POST") {

  $usuario = $_POST['usuario'];
  $contrasena = $_POST['contrasena'];


  $consulta = "SELECT * FROM usuarios WHERE nombre='$usuario' AND contrasena='$contrasena'";


  $resultado = $conexion->query($consulta);


  if ($resultado->num_rows > 0) {
      session_start();
      $_SESSION['usuario'] = $usuario;
      header("Location: bienvenido.php");
      exit();
  } else {
      header("Location: inicio.php?error=1");
      exit();
  }


  mysqli_close($conexion);
}
?>
```

Figura 27 – Código vulnerable a SQLi

Esto permite que la consulta engañe a la base de datos SQL y el atacante se autorice como si fuese un usuario legítimo. De este modo podrá burlar el parámetro de seguridad del archivo “bienvenido.php”.

Fuzzing de directorios: El uso de herramientas como dirbuster para descubrir directorios ocultos ayuda a los participantes a entender cómo los administradores web a veces dejan información sensible accesible sin la seguridad adecuada. En este caso, el robots.txt no contiene la información habitual, pero sí que nos da una pista para el ataque de fuerza bruta.

Ataque de diccionario por SSH: usando Hydra y las pistas de la página web y robots.txt, los usuarios aplican un ataque de fuerza bruta haciendo uso del reconocido diccionario “rockyou.txt” para acceder al servidor por SSH, lo que subraya la importancia de usar contraseñas fuertes y políticas de seguridad robustas.

- **Herramientas Utilizadas en la resolución**

Nmap: Utilizado para realizar un escaneo inicial e identificar puertos abiertos. El output en este caso es el puerto 80 para HTTP y el puerto 22 para SSH.

Herramientas de automatización(ffuf/wfuzz/dirbuster): herramientas de fuzzing web empleadas para descubrir directorios y archivos presentes en el servidor, esta herramienta es crucial para encontrar el archivo access.php que contiene un formulario de login vulnerable a SQLi. Si bien lo habitual es usar un login.php, se quería complicar un poco más y forzar al usuario a usar estas herramientas para ir familiarizándose con ellas.

Inspección de la Página: usando herramientas de inspección manual del código fuente de la página. Al llegar a la página de bienvenido se encuentra un botón que, al hacer clic, revela que está temporalmente deshabilitado. Esto resulta ser un engaño, por lo que se requiere la herramienta inspeccionar del navegador para analizar la página y encontrar la verdadera primera flag. Su uso puede también revelar el contenido del JS para comprender la lógica de la aplicación si este no está debidamente protegido.

Herramientas de SQL Injection: Se pueden utilizar herramientas de automatización para la inyección SQL tales como sqlmap, sin embargo, en esta ocasión es una SQL inyection simple por lo que no fue necesario hacer uso de dicha herramienta, pero si se menciona en los videos de resolución.

Hydra: Herramienta de fuerza bruta utilizada para lanzar un ataque de diccionario contra el servicio SSH. Con la información obtenida el participante emplea la herramienta Hydra para lanzar un ataque de diccionario utilizando el nombre de usuario "osiris" y el diccionario de contraseñas “rockyou.txt” contra el servicio SSH en el puerto 22 de la máquina víctima. Hydra es una herramienta de fuerza bruta que intenta múltiples combinaciones de contraseñas hasta encontrar la correcta.

5.4.3. La Edad Media

- **Definición del reto**

En el tercer reto, el participante inicia con una fase de reconocimiento donde identifica dos servicios web: un NGINX en el puerto 80 y un Apache en el puerto 8080. En primer lugar, el usuario intentará acceder a la web usando su navegador pero este devolverá un error 404. El servicio de NGINX que está en el puerto 80 es un señuelo. Esto se ve representado en el esquema de la figura 28

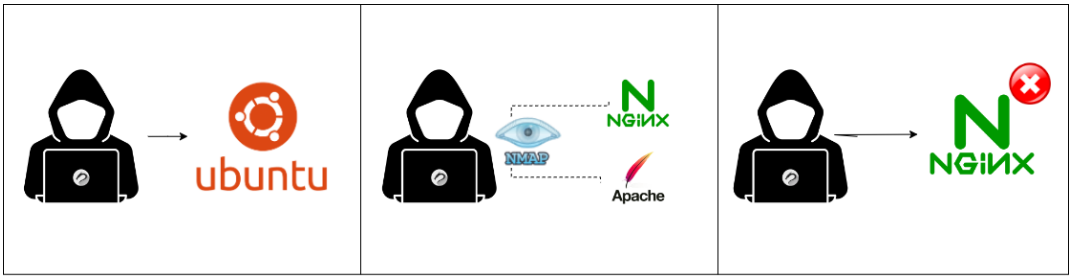

Figura 28 – Esquema reconocimiento y señuelo

Tras determinar que el NGINX es un señuelo, la atención se dirige hacia el Apache, por lo que accederá a este haciendo uso del navegador y forzando el uso del puerto 8080. En este punto se encuentra una página que permite la carga de archivos, pero únicamente en formatos jpg o jpeg. En la figura 29 se muestra una captura de la interfaz que ofrece el Apache

Figura 29 – Interfaz subida de archivos

El atacante, buscando una manera de ejecutar código en el servidor, decide crear un archivo PHP malicioso que, al ser lanzado, establece una reverse shell utilizando Netcat hacia la IP de la máquina atacante. Hay muchas maneras de establecer dicha reverse shell [REF14], aunque en esta ocasión se ha usado bash y netcat. Este archivo es inicialmente guardado con una extensión .jpg para satisfacer las restricciones de la página de carga.

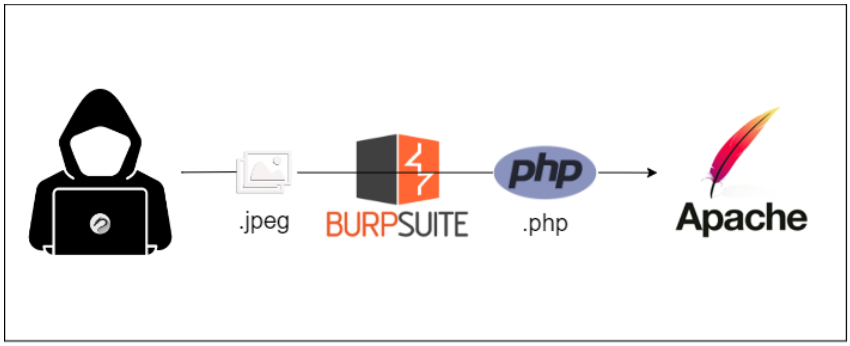

Figura 30 – Esquema uso de burpsuite

Posteriormente, el archivo es enviado al servidor no sin antes activar Burpsuite y Foxyproxy, esto esta esquematizado en la figura 30, permitiendo manipular la extensión del archivo. El atacante hace uso de Burp Suite, configurado como proxy, para interceptar la solicitud de carga del archivo. Durante esta intercepción, cambia la extensión del archivo de .jpg a .php. Se muestra en la figura 31 la traza capturada y modificada en Burpsuite.

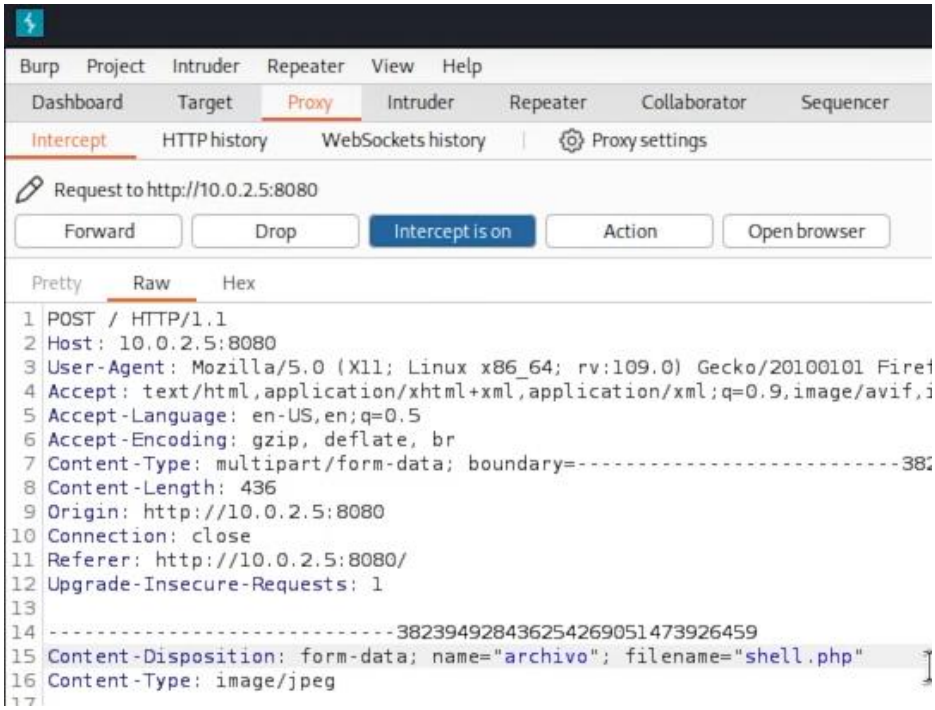

Figura 31 – Captura de traza con Burpsuite

Luego de modificar la solicitud, establece una conexión en escucha con netcat desde la maquina atacante y accede al archivo PHP a través del navegador, lo que activa la reverse shell y permite al atacante obtener acceso a la máquina como el usuario www-data y recuperar de esta forma la primera flag. Se puede ver visualmente en el esquema de la figura 32.

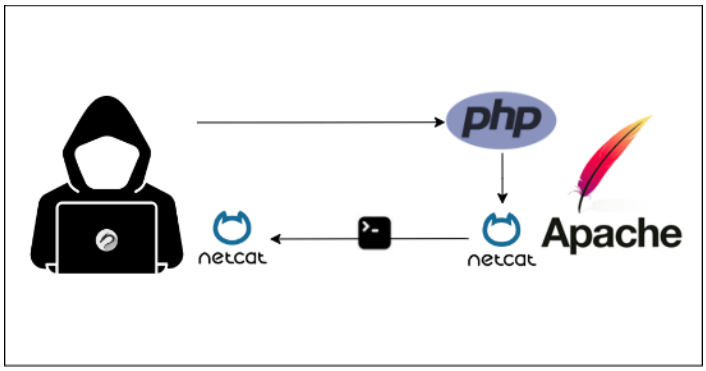

Figura 32 – Esquema reverse Shell.

La exploración del sistema revela junto a la flag un archivo .c que indica el desarrollo de una variante de la función 'head'. Interpretando su código se descubre que el binario resultante posee permisos sudo y ejecuta la función ‘head’. El atacante identifica la ubicación de este binario y aplica la técnica de path hijacking. Para esto, ajusta las

variables de entorno para que cuando se ejecute la función 'head' se active un script controlado por el atacante en lugar del binario original. Este script devuelve una shell con privilegios de root debido a los permisos del binario.

Se incluye una captura en la figura 33 que muestra el contenido del archivo ".c" y el comando export para modificar el PATH.

```
www-data@media:/home/www-data$ cat head_personalizado.c
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>

int main() {
    setuid(0);
    printf("\nEjecutando head:\n");
    system("head /etc/passwd");
}

//Este código es muy peligroso y tu eres muy despistado. ¡Por favor, acuerdate de eliminar el binario generado!
www-data@media:/home/www-data$ chmod 777 head
www-data@media:/home/www-data$ export PATH=.:$PATH
```

Figura 33 – Contenido del archivo .c y comando para modificar el PATH

Finalmente, ejecutando el binario modificado con permisos sudo, el atacante obtiene una Shell de root, lo que le permite acceder y recuperar la segunda flag, completando así el desafío. El escalado de privilegios está representado con el esquema de la figura 34.

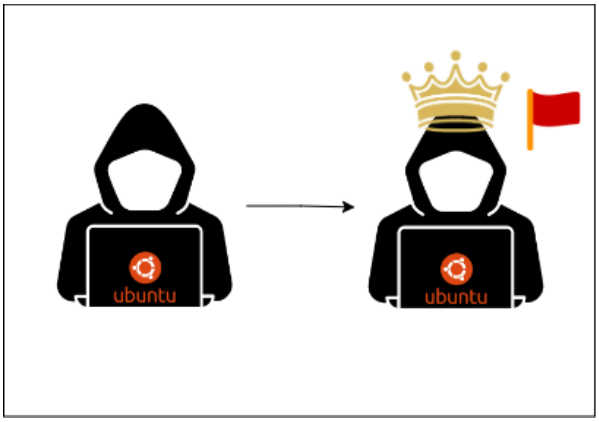

Figura 34 – Esquema escalado de privilegios

- **Vulnerabilidades Exploradas**

Vulnerabilidad de Subida de Archivos: El servidor Apache en el puerto 8080 permite la subida de archivos sin el debido filtrado. La vulnerabilidad en el código se debe a que permite a los usuarios subir archivos sin verificar bien el contenido del archivo, la verificación del archivo debería hacerse del lado del servidor, nunca del navegador. Esto permite que alguien con malas intenciones podría subir un archivo con la extensión esperada, como una supuesta imagen .jpg, modificar su transacción a .php u otras extensiones que le permitan posteriormente la ejecución remota de comandos.

Ejecución Remota de Comandos: A través del archivo malicioso subido, el usuario puede ejecutar comandos en el servidor, lo cual es un paso crítico para lograr el acceso a la máquina. En este caso el archivo maligno se ha introducido intencionalmente por parte del atacante, pero es posible que existan archivos propios e inseguros del lado del servidor que le permitan al atacante ejecutar comandos sobre la maquina víctima.

Path Hijacking para el Escalado de Privilegios: El path hijacking se basá en una modificación de la variable path, la cual indica los distintos directorios ordenados, donde buscar un binario al ser llamado. Para este ataque, una vez dentro del sistema, el usuario debe utilizar el path hijacking junto al binario inseguro para obtener mayores permisos y acceder a la flag protegida.

- **Herramientas Utilizadas para la resolución**

Nmap: en este caso, se descubren dos servicios web: un NGINX en el puerto 80 y un Apache en el puerto 8080.

Burp Suite: Herramienta esencial para interceptar y modificar solicitudes HTTP. Esta herramienta tiene cientos de funcionalidades aplicables en otros retos, pero en este es fundamental para cambiar la extensión del archivo cargado en el servidor. Se configura Burp Suite como proxy, utilizando Foxyproxy para redirigir el tráfico del navegador a través de este.

Foxyproxy: Complemento de navegador utilizado para redirigir el tráfico. Se emplea para facilitar la configuración de proxies, permitiendo redirigir el tráfico del navegador a través de Burp Suite. Es esencial para la manipulación de solicitudes HTTP de manera eficiente durante la fase de carga de archivos.

Netcat: Para establecer una reverse shell, se utiliza Netcat. El archivo PHP malicioso creado contiene un comando de Netcat que, al ejecutarse, establece una conexión de vuelta (reverse shell) hacia la máquina del atacante. Netcat se configura en la máquina atacante para escuchar en un puerto específico, esperando la conexión entrante desde el servidor comprometido.

Shell de Root: Finalmente, al ejecutar el binario modificado con permisos sudo, el atacante obtiene una shell de root. Esta herramienta le permite acceder a todos los archivos del sistema y recuperar la segunda flag, completando así el desafío.

5.4.4. La Edad Moderna

- **Definición del reto**

En el tercer desafío, el atacante comienza desde el inicio con acceso limitado a la máquina víctima a través de una cuenta de usuario invitado. Aunque tiene permisos para ejecutar un script en la máquina, no tiene visibilidad directa de su contenido. Para analizar y entender las operaciones que realiza el script, el atacante configura Wireshark en su propia máquina para monitorear la actividad de red que resulta de la ejecución del script, tal y como se muestra en la figura 35.

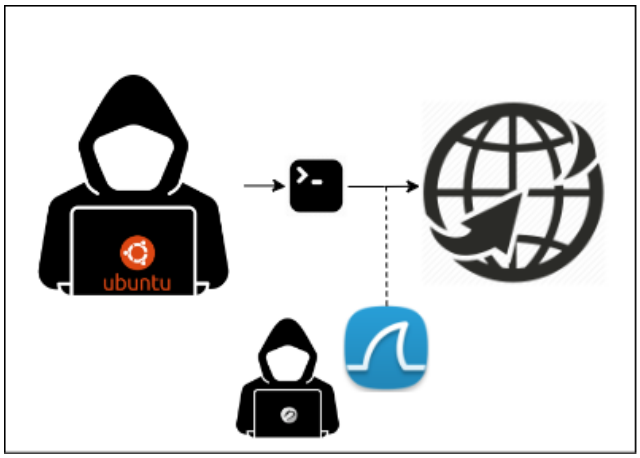

Figura 35 – Esquema uso de wireshark para la escucha de tráfico

Durante la monitorización con Wireshark, el atacante captura varios datos clave.

- Descubre credenciales de usuario y contraseña que iban sin cifrar mediante HTTP hacía un login
- Aparece tráfico hacia dos urls significativas: una lleva a un blog sobre el cifrado Vigenère y otra a una página dedicada al dios del tiempo, Cronos.
- La captura más importante revela que la función “Comprobación de seguridad” realiza solicitudes a una URL específica http://dguezprojects.es/securityservice Este sitio espera recibir ciertos parámetros; si se cumplen correctamente, elimina la contraseña que protege al usuario “Myth”.

Con estos datos, el atacante procede al siguiente paso de su estrategia. Configura un servidor Apache en su propia máquina y replica el código obtenido de la URL mencionada. Modifica los parámetros del código basándose en la información obtenida a través de Wireshark, incluyendo las credenciales de usuario y contraseña. Además, existe una clave por ubicación. Esta clave había sido revelada en el mapa del sitio web del reto tras resolver los desafíos anteriores, no hay otra manera de obtenerla y su valor es "Mediterráneo". En la captura de la figura 36 se muestra la página real y los valores que participante deberá cambiar.

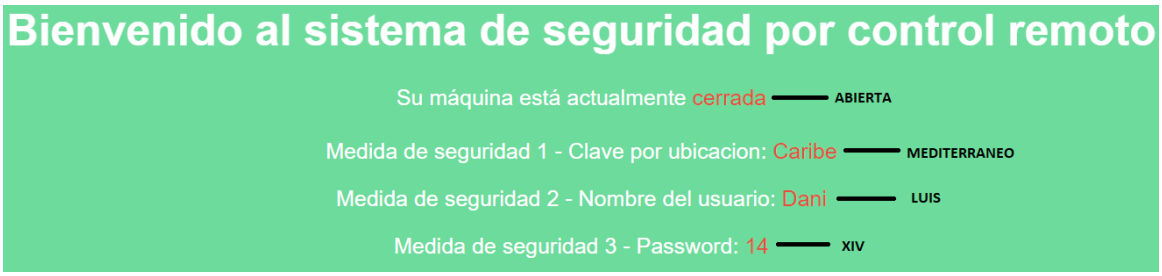

Figura 36 – Captura del sistema de seguridad

El ataque avanza con la implementación de DNS spoofing utilizando Ettercap. El atacante empieza con un ARP poisoning, manipulando la tabla ARP de la máquina víctima para hacerse pasar por el Gateway. Cuando se ejecuta el script desde la máquina víctima, Ettercap intercepta las solicitudes DNS y responde con direcciones maliciosas que redirigen al servidor Apache controlado por el atacante en lugar del dominio legítimo "dguezprojects.es", como se representa en la figura 37.

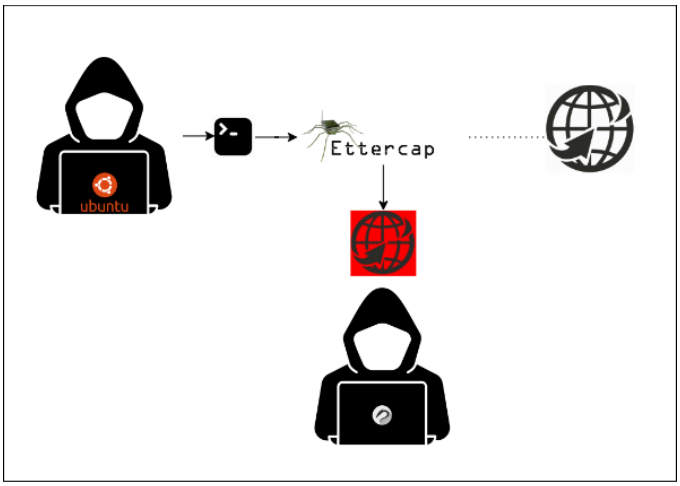

Figura 37 – Esquema uso de ettercap para DNS spoofing

Esta maniobra engaña al script para que crea que está interactuando con el servicio legítimo y, como resultado, deshabilita la contraseña del usuario "Myth", representado en la figura 38.

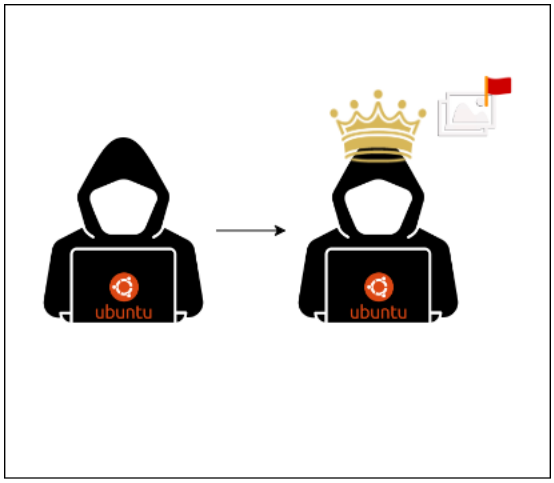

Figura 38 – Esquema escalado de privilegios.

Una vez dentro del entorno del usuario “Myth”, el atacante no encuentra una flag directamente, sino una imagen que esconde un mensaje secreto oculto mediante esteganografía. Al extraer y analizar el contenido oculto, descubre que el secreto no era el final del reto, sino que hay dos acertijos y la flag está cifrada. Utilizando la información sobre Vigenère obtenida previamente y la referencia a Cronos en conjunto a los acertijos, el participante determina que el cifrado utilizado es Vigenère y que la clave para descifrar la flag es "cronos". Este proceso se ha esquematizado en la figura 39

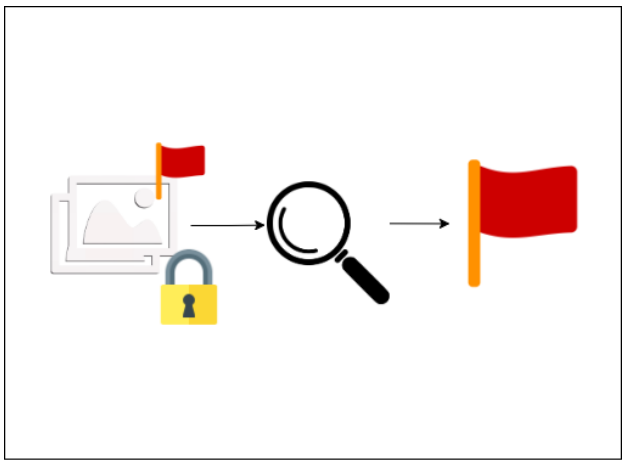

Figura 39 – Esquema esteganografía y criptografía

Finalmente, el atacante aplica esta clave para descifrar la flag, completando con éxito el desafío más complejo, dando por finalizado el CTF.

- **Vulnerabilidades Exploradas**

Sniffing: los usuarios tienen acceso a un segmento de la red donde se encuentra la máquina víctima y por donde genera cierto tráfico. Utilizando herramientas de captura de paquetes como Wireshark, los participantes observan cómo los datos viajan a través de la red, identificando información clave como usuarios y contraseñas, que dominios han sido visitados y su interpretación. Parte de como se generá el tráfico se muestra en la figura 40.

```
# Generar trafico
echo "Ejecutando generacion de trafico..."

urlvigenere="http://www.dcode.fr/cifrado-vigenere"
curl -s "$urlvigenere" > /dev/null 2>&1

urldios="http://www.worldhistory.org/trans/es/1-21126/cronos/"
curl -s "$urldios" > /dev/null 2>&1



URLLOGIN="http://testphp.vulnweb.com/login.php"
USERNAME="Luis"
PASSWORD="XIV"


curl -s -X POST -d "username=$USERNAME&password=$PASSWORD" $URLLOGIN >/dev/null 2>&1

echo "Finalizado"
;;
```

Figura 40 – Función generar tráfico del script

DNS Spoofing: implica hacerse pasar por un dominio legítimo para engañar a la maquina víctima y que esta crea que está interactuando con una fuente confiable. Se explotará utilizando Ettercap o similares, una herramienta conocida por su capacidad para realizar ataques de man-in-the-middle (MitM) y suplantación a través de la red. En este caso el objetivo es engañar a un script para que ejecute la función de verificación de seguridad, permitiendo el acceso a una cuenta privilegiada. La función que va a ser engañada se encuentra en la figura 41.

```
# Comprobacion de seguridad
echo "Ejecutando comprobacion de seguridad..."

response=$(curl -s http://dguezprojects.es/securityservice.html)
status=$(echo "$response" | grep -o 'class="status">.*<' | sed 's/class="status">//;s/<.*//')
location=$(echo "$response" | grep -o 'Medida de seguridad 1 - Clave por ubicacion: <span class="location">[^<]*</spa
s="location">//;s/<.*//')
username=$(echo "$response" | grep -o 'Medida de seguridad 2 - Nombre del usuario: <span class="username">[^<]*</span
="username">//;s/<.*//')
pass=$(echo "$response" | grep -o 'Medida de seguridad 3 - Password: <span class="password">[^<]*</span>' | sed 's/Me


echo $status
echo $location
echo $username
echo $pass

if [[ "$status" == "abierta" && "$location" == "Mediterraneo" && "$username" == "Luis" && "$pass" == "XIV" ]]; then
    echo "Eliminando contraseña para el usuario myth..."
    # Comando para eliminar la contraseña del usuario "myth"
    sudo passwd -d myth
    echo "Contraseña del usuario myth eliminada correctamente."
else
    echo "No se cumplieron las condiciones de seguridad."
fi
;;
```

Figura 41 – Función comprobación de seguridad del script

Criptografía y Esteganografía: La flag final está cifrada y escondida dentro de una imagen. El usuario tendrá que aplicar esteganografía para sacar el secreto, que a su vez está cifrado con Vigenere. Esto lo puede averiguar interpretando las capturas de tráfico o resolviendo el acertijo. Finalmente usando una herramienta de descifrado como cyberchef, podrá obtener la flag.

- **Herramientas Utilizadas**

Herramientas de Sniffing: herramientas como Wireshark se utilizan para capturar y analizar el tráfico generado por el script, permitiendo a los usuarios observar y entender las operaciones que se realizan en segundo plano.

Ettercap: herramienta de software utilizada para realizar ataques de red, específicamente ataques man-in-the-middle. Tiene múltiples funcionalidades, en este caso se utilizará para la interceptación y modificación de datos en tránsito. Será la herramienta que permitirá el ARP poisoning y posteriormente el DNS spoofing.

Steghide: permite ocultar datos dentro de archivos multimedia como imágenes, audio y video de forma imperceptible, soportando formatos como BMP, WAV y JPEG, y ofreciendo opciones de compresión y encriptación de datos. También se puede usar al contrario, para obtener el secreto oculto en alguno de estos archivo.

CyberChef: es una plataforma web para realizar cifrado, decodificación, compresión y análisis de datos, todo a través de una interfaz muy sencilla de entender. En esta ocasión nos permitirá obtener la flag cifrada por Vigenere usando la clave “Cronos”

6. EXPERIMENTOS Y VALIDACIÓN

En el siguiente capítulo se explicarán las diferentes fases de pruebas a las que se ha sometido el proyecto. A continuación, se explicará que procedimiento se ha seguido para validar la plataforma además de sacar ciertas conclusiones.

6.1. Experimentación

Se llevaron a cabo 3 fases de experimentación a lo largo del desarrollo de este CTF.

6.1.1. Fase de pruebas iniciales – Realizadas por el desarrollador

El proceso de testing sobre el desarrollo de la plataforma CTF comenzó con pruebas exhaustivas realizadas por mí, el desarrollador, inmediatamente después de configurar cada reto y tener la página web funcional. En estas pruebas iniciales, se identificaron errores técnicos que se rectificaron de inmediato antes de continuar con las siguientes etapas del desarrollo. Esta etapa fue crucial pues aseguró que los fundamentos de cada reto estuvieran sólidamente establecidos y funcionaran correctamente al menos en un entorno local, preparando el terreno para pruebas más extensivas y rigurosas en fases posteriores. Además, se puso a prueba que toda la funcionalidad web estaba completa y correcta.

Una vez se comprobó que estos dos elementos eran funcionales se pasó a una siguiente fase.

6.1.2. Pruebas externas – Realizadas por experto

Se invitó a un compañero experto en ciberseguridad para someter la plataforma a una prueba crítica. Este paso fue esencial para obtener una perspectiva externa y experta, crucial para identificar problemas que no fueron evidentes durante las pruebas iniciales.

Durante esta fase, se descubrieron configuraciones que, aunque funcionaban bien en local, presentaban complicaciones cuando se operaban en remoto. Estos problemas incluyeron, por ejemplo, configuraciones de red y problemas de compatibilidad del software utilizado en la virtualización y que por consiguiente afectaban la funcionalidad de la plataforma en un entorno realista. La colaboración con el experto proporcionó insights valiosos que condujeron a ajustes técnicos significativos y mejoras en la configuración general de la plataforma. Algunos de los problemas se pudieron solventar, otros se tuvieron eliminar o ser modificados

Se puso a prueba una última vez por parte del desarrollador, mientras se grababan los vídeos de resolución y tras su posterior edición y subida, se pudo continuar con la etapa final.

6.1.3. Pruebas de Usuario – Realizadas por un grupo de usuarios selecto

La fase final de la validación comenzó una vez que todos los retos fueron ajustados y la plataforma estuvo operativa y sin errores.

En esta etapa, se compartió la plataforma con un grupo seleccionado de usuarios, que representaban en mayor o menor medida a la audiencia objetivo de la plataforma. Los usuarios fueron invitados a interactuar con los retos, explorar las funcionalidades de la

plataforma y proporcionar feedback sobre su experiencia. Este proceso no solo sirvió para evaluar la usabilidad y la efectividad educativa de la plataforma, sino que también permitió recopilar sugerencias y recomendaciones para futuras mejoras.

Este feedback proporcionó una confirmación crucial de que la plataforma cumplía con sus objetivos educativos, al tiempo que destacaba ciertos puntos débiles sobre los que trabajar en futuras versiones de la plataforma. Estas interacciones demostraron la importancia de la iteración entre usuario y desarrollador.

6.2. Validación

Como se ha mencionado anteriormente, un conjunto de 5 usuarios fue seleccionados para poner a prueba la plataforma y conseguir de este modo su feedback. Todos ellos tuvieron que realizar los 4 retos, interactuar con la página y a continuación, rellenar una encuesta donde se valoraban varios puntos clave de este desarrollo. Los resultados obtenidos de este formulario, incluido como anexo de este proyecto, están interpretados a continuación:

6.2.1. Experiencia General en la Plataforma:

Los participantes evaluaron su experiencia general con un promedio de 8 en una escala del 1 al 10, lo que destaca el éxito de la plataforma en proporcionar un entorno atractivo y funcional. Este alto nivel de satisfacción general subraya el acierto del diseño interactivo y la estructura de los retos que mantienen a los usuarios comprometidos y motivados.

6.2.2. Dificultad del CTF:

La dificultad se calificó de media con un 7. El grupo de usuarios era diverso, algunos no tenían ninguna experiencia en este tipo de retos, otros habían participado en algún reto y también contábamos con participantes con amplia experiencia en el mundo del hacking ético. Esta puntuación refleja un equilibrio cuidadoso entre desafío y accesibilidad, crucial para mantener a los usuarios interesados sin sobrepasar su capacidad de resolución. Esta medición ayuda a comprender cómo los usuarios perciben el escalonamiento de los retos y la progresión dentro de la plataforma.

6.2.3. Aprendizaje Derivado:

Los usuarios reportaron un alto nivel de aprendizaje, con una puntuación promedio de 8. Este resultado es indicativo de la efectividad de los materiales educativos y los desafíos prácticos ofrecidos. Destaca la importancia de los métodos empleados para la transmisión de conocimientos técnicos y prácticos, mediante el contenido audiovisual, en un campo tan complejo como la ciberseguridad. Muchos de los comentarios recibidos elogiaban los vídeos tanto de narrativa como de resolución.

6.2.4. Ambientación y Entretenimiento:

Las puntuaciones para ambientación y entretenimiento también fueron altas, con un promedio de 8. Estas métricas son vitales porque, como se ha presentado a lo largo de todo este documento, una experiencia entretenida es fundamental para el aprendizaje experimental. Este promedio refleja el éxito en la creación de un entorno inmersivo que no solo educa, sino que también engancha a los usuarios.

6.2.5. Reto 1 - La Prehistoria:

La complejidad de este reto se valoró como baja, con un promedio de 3 sobre 5, este reto se considera muy accesible, ideal para un primer contacto con los retos CTF. Por otro lado, la ambientación fue calificada con 4 sobre 5, lo que indica que los elementos narrativos y visuales fueron efectivos para establecer el contexto histórico y mejorar la inmersión en el problema.

6.2.6. Reto 2 - La Edad Antigua:

En cuanto a complejidad, este reto recibió una valoración media de 4 sobre 5. Este promedio nos indica que los participantes encontraron un incremento adecuado en el desafío en comparación con el primer reto, lo cual es esencial para mantener la curva de aprendizaje progresiva. Sobre la ambientación, también fue calificado con un 4 sobre 5, lo que refleja una recepción positiva de la integración temática que ayuda a sumergir a los usuarios en el contexto del CTF. Este incremento está justificado pues al tener una interfaz gráfica, fue posible incluir más elementos visuales.

6.2.7. Reto 3 - La Edad Media:

El tercer reto, tuvo una evaluación de complejidad de 4.5 sobre 5, muestra una evolución en la dificultad percibida, adecuada para usuarios que han superado los retos anteriores y están buscando desafíos aún mayores. La ambientación recibió una calificación de 4 sobre 5. Del mismo modo que en el reto anterior, la capacidad de incluir narrativa a través de imágenes y textos para dar contexto al reto ha posibilitado este incremento.

6.2.8. Reto 4 - La Edad Moderna:

Este último reto alcanzó una calificación de 4.5 sobre 5 en complejidad, indicando que se logró el objetivo de proporcionar un desafío significativo y avanzado adecuado para el cierre de la serie de retos. La ambientación consiguió un pleno en esta ocasión, con una puntuación de 5 sobre 5, este reto fue especialmente elogiado en este aspecto, obteniendo la valoración la más alta entre todos los retos. Los usuarios apreciaron cómo los elementos visuales y temáticos se alinearon efectivamente con los desafíos técnicos, proporcionando una conclusión inmersiva y memorable de la serie.

7. CONCLUSIONES, DESAFIOS ENFRENTADOS y TRABAJOS FUTUROS

En el siguiente capítulo se presentarán las conclusiones que se han sacado después del desarrollo del proyecto y también se presentarán posibles mejoras o trabajos futuros para el desarrollo de la plataforma.

7.1. Conclusión

El objetivo principal que era desarrollar y validar una plataforma educativa interactiva tipo Capture The Flag (CTF) dirigida principalmente a principiantes en el campo de la ciberseguridad y el mundo del hacking ético se ha cumplido.

En cuanto al objetivo "Diseño de retos escalados en ciberseguridad" también se ha cumplido correctamente. La plataforma logra establecer una gradualidad en la dificultad de los retos, haciendo que cada reto sea ligeramente más desafiante que el anterior. Esta afirmación también se ve apoyada por el feedback recibido por parte de los participantes, pues la valoración en la dificultad ha ido casi siempre incrementándose paulatinamente en cada uno de los retos.

Por otro lado, también se ha logrado completar el objetivo "Evaluación de la efectividad de la plataforma". Tras finalizar el desarrollo, la plataforma fue juzgada por el grupo de usuarios tester, recibiendo por parte de estos sus valoraciones a través del formulario. La plataforma pretende extenderse a nuevos entornos para ser evaluada, por lo que siempre hay cabida para nuevas propuestas y críticas por parte de los futuros participantes.

El objetivo "Promoción de la conciencia en ciberseguridad" es uno de los retos que a lo largo de este desarrollo no se ha logrado superar. Al ser una plataforma en preproducción, muy acotada a un grupo de usuarios reducido y controlado, es imposible considerar que se haya obtenido una mejora real sobre la conciencia en ciberseguridad. Por otro lado, sí que se ha observado que los usuarios no solo mejoran en sus habilidades técnicas, sino que también desarrollan una comprensión más robusta de los principios de la seguridad informática. Aunque estos resultados demuestran que en un futuro la plataforma puede ayudar en materia de concienciación y educación, por ahora no se puede hacer esta afirmación.

Respecto al objetivo "Aplicar los conocimientos obtenidos en el grado de ingeniería de la ciberseguridad" no hay duda de que se ha superado considerablemente. Durante el desarrollo de la web se han puesto en práctica los conocimientos aprendidos en la asignatura de "Desarrollo web seguro". A la hora de configurar los retos asignaturas como "técnicas de hacking", "pentesting" o "seguridad en redes" han servido como base para el desarrollo de los retos. Complementando todo esto, "ingeniería del software" ha guiado todo el desarrollo de la plataforma. La selección de plataformas cloud para el alojamiento de los recursos ha estado profundamente influenciada por la asignatura de "Redes avanzadas y computación en la nube". Prácticamente todas y cada una de las asignaturas del grado han servido de ayuda en algún punto del desarrollo.

La narrativa es sin duda uno de los pilares fundamentales del proyecto, por lo que podemos afirmar que ambos objetivos "Creación de una narrativa inmersiva" e "Integración de elementos interactivos y multimedia" se han cumplido con creces. La historia detrás del contexto del CTF sirve como hilo conductor durante toda la resolución del reto, acompañando efectivamente al usuario en este juego educativo. Los vídeos se integran a la perfección con el contexto y sirven como apoyo para aumentar aún más la implicación del usuario con la plataforma. El feedback recibido sobre esta parte es más que notorio. La estructura del curso ha sido particularmente elogiada por su capacidad de guiar a los usuarios a través de conceptos hilados entre ellos y en un formato que les ha resultado entretenido, incluso a los más experimentados. Considerado uno de los grandes puntos fuertes de todo el reto tanto por mi parte como por los testers.

Un objetivo marcado que no se ha llegado a cumplir ha sido la "Implantación de la plataforma en contextos educativos formales". Este objetivo fue preconcebido para un futuro en el que la plataforma estuviese capacitada para extenderse. Los comentarios recibidos por parte de los usuarios indican un alto grado de satisfacción, mencionando cómo la plataforma ha facilitado la posibilidad de aprender e introducirse en el basto y complejo mundo de la ciberseguridad. Profesores y profesionales del sector podrían utilizar la plataforma como una herramienta de enseñanza a modo de complemento para aquellos alumnos que tengan interés en ir más allá del programa formativo establecido.

Por otro lado, la importancia que se le ha dado a la narrativa ha ayudado a mantener a los usuarios motivados y comprometidos con el material del curso, un factor crucial en el aprendizaje autónomo por el que aboga este TFG. La motivación es un factor esencial en la enseñanza, tal y como se indica en [REF15] por eso se le ha dado tanta atención a desarrollar algo que vaya más allá del conocimiento teórico, sino que motive al participante a seguir aprendiendo.

También se puede concluir que las vulnerabilidades y las tecnologías empleadas para los retos se escogieron debidamente. Se optó por un paquete tecnológico basado en herramientas gratuitas para garantizar la accesibilidad a los participantes y la facilidad para integrarlos en la creación de retos. Cada herramienta fue escogida por su relevancia en el mundo real de la ciberseguridad, permitiendo a los usuarios desarrollar habilidades transferibles al entorno laboral o para futuros retos.

7.2. Desafíos enfrentados

Desarrollar una plataforma CTF educativa para principiantes en ciberseguridad conllevó numerosos desafíos, tanto técnicos como pedagógicos, que requirieron soluciones creativas y muchas modificaciones durante el desarrollo. A continuación, se detallan algunos de los principales desafíos enfrentados durante este proyecto.

El primer y más complejo desafío enfrentado durante este TFG fue comenzar a desarrollar una plataforma completa y funcional con un único desarrollador. La plataforma tenía un objetivo demasiado ambicioso para un único estudiante. La experiencia laboral obtenida junto al desarrollo de este TFG ha sido crucial para poder hacer frente a tan desafiante reto.

En segundo lugar, otro de los desafíos más complejos encontrados ha sido encontrar el cómo montar los retos. Si bien existen un sinfín de documentos y artículos sobre vulnerabilidades, los recursos que explican como implementar la vulnerabilidad son mucho más escasos o inexistentes. Fue necesario comprender completamente en que consiste el ataque, para readaptar la idea y forzar una vulnerabilidad que lo permita.

Otro de los desafíos más significativos fue asegurar que la plataforma web y las máquinas virtuales funcionaran de manera fluida y sin errores. Durante la fase de pruebas, surgieron varios problemas técnicos relacionados con la configuración de red o la compatibilidad de software en entornos virtuales distintos. Algunos retos supusieron muchas horas de trabajo para conseguir que funcionasen. Estos problemas requirieron una revisión constante acompañada de ajustes para garantizar que los participantes pudieran acceder a los retos sin interrupciones en el momento en el que la plataforma se expandiese.

Hay que mencionar explícitamente la complejidad que conllevo la creación del cuarto y último reto. Este reto fue una idea completamente original, nunca había realizado ningún reto de este estilo, pero quería incluir material relacionado con redes. En primer lugar, intente comprender como funcionan las herramientas para este tipo de ataques, consulte varios artículos [REF16] y vídeos [REF17] para lograrlo. A continuación, tuve que extrapolar el conocimiento obtenido para poder habilitarlo. Tras un largo prueba y error, finalmente terminé desarrollando el que, para mí, es el mejor reto de este CTF.

Otro desafío importante fue la creación de contenido educativo, era necesario que fuera técnico pero accesible. Traducir conceptos complejos de ciberseguridad en lecciones digeribles para los principiantes requirió una revisión constante además de algunos ajustes basados en el feedback de los usuarios. Todos los vídeos fueron retocados y redefinidos en algún punto tras recibir el feedback del grupo de testers, asegurando que cada lección fuera clara y comprensible.

Mantener a los usuarios motivados y comprometidos con el reto fue otro desafío clave. La incorporación de una narrativa envolvente era un requisito indispensable para la idea que tenía sobre el reto. La creación y edición de los vídeos no supusieron un gran reto técnico pues ya contaba con cierta experiencia en la materia, pero si conllevaron un gran consumo de tiempo y muchas correcciones tanto en el guion como en la edición.

La fase de pruebas y validación también presentó desafíos significativos, el primero fue conseguir que los usuarios aceptasen compartir su tiempo de manera altruista para llevar a cabo las pruebas pertinentes. El feedback de los usuarios fue invaluable, pero implementar sus las sugerencias y resolver los problemas detectados requirió un esfuerzo considerable.

7.3. Propuestas de mejora para la plataforma CTF

En el siguiente capítulo se presentarán las mejoras técnicas propuestas para la optimización y evolución de la plataforma. A continuación, se detallan las distintas mejoras en contenido y funcionalidad que se han identificado, explicando su relevancia y cómo contribuirán a mejorar el rendimiento y la funcionalidad general del proyecto.

7.3.1. Mejoras Técnicas

En este apartado, se listan una serie de mejoras técnicas que se podrían implementar para optimizar el actual funcionamiento de la plataforma

Optimización del rendimiento: aunque la plataforma funciona eficientemente, siempre hay espacio para mejorar la velocidad de carga y la respuesta de la interfaz de usuario sobre todo cuando la plataforma pase a un entorno de producción real y el tráfico sobre la misma se incremente exponencialmente. Esta es una de las razones por las que se ha decidido trabajar en un entorno cloud.

Readaptación de los retos a un formato más portable: si bien trabajar en herramientas como VirtualBox es parte de la educación que se quería impartir con este CTF, la descarga de retos tan pesados y su importación podría verse beneficiada si se tratase de un contenedor de Docker en lugar de una máquina virtual.

Aleatorización de flags: las flags por el momento son completamente estáticas, por lo que sería interesante crear un paquete de flags aleatorio que se vaya intercalando y que, de este modo la pista del reto 4 también sea variante.

7.3.2. Expansión de Contenido y Funcionalidades

En este apartado, se explorarán las propuestas para expandir el contenido y las funcionalidades de la plataforma, con el fin de enriquecer la experiencia de los usuarios.

Niveles avanzados: actualmente, la plataforma está diseñada para principiantes, pero sería conveniente continuar con su formación e incluir nuevos y más desafiantes retos. Agregar niveles más avanzados con retos más complejos ayudaría a mantener el interés de los usuarios a medida que desarrollan sus habilidades y podría proporcionarles una experiencia más rica además de prepararlos mejor para los desafíos del mundo real.

Nuevos tipos de reto: esto también podría incluir la creación de rutas de aprendizaje especializadas para diferentes áreas de interés dentro de la ciberseguridad incluyendo retos sobre tecnologías emergentes como inteligencia artificial, Blockchain, Web3…

Ataque-defensa: sería realmente interesante en un futuro expandir la plataforma con retos completamente distintos al que se ha desarrollado en este proyecto. Un reto en tiempo real de ataque-defensa por equipos sería muy interesante de desarrollar.

Análisis forense post-evento: incorporar retos que involucren análisis forense después de un ciberataque. Crear retos sobre análisis de datos, identificación de vectores de ataque sería un punto muy interesante, además todo estaría guionizado como en este mismo proyecto. Los usuarios pueden aprender también otro sector de la ciberseguridad como es el análisis forense, aprendiendo a investigar y responder a incidentes de seguridad de manera efectiva.

Cuestionario final: introducir un cuestionario en la propia página web justo al finalizar el reto, para que el usuario pueda valorar la plataforma del mismo modo que lo han hecho los usuarios que han probado el proyecto.

Implementación en entornos educativos: Establecer asociaciones con instituciones educativas o empresas centradas en la docencia del sector para lograr conseguir algunos de los objetivos que han quedado pendientes de este proyecto. Estas colaboraciones también pueden proporcionar recursos adicionales para el desarrollo y la expansión de la plataforma.

Implementación de Chatbots Inteligentes: Desarrollar chatbots con inteligencia artificial fue una de las ideas originales que tuvo que quedar fuera del proyecto. La idea es que estos chatbots puedan responder preguntas frecuentes a modo de pista para guiar a los usuarios a través de los retos y proporcionar asistencia en tiempo real. Esto puede mejorar la experiencia del usuario al ofrecer soporte inmediato y personalizado.

8. BIBLIOGRAFIA

REF1	Análisis y diagnóstico del talento de ciberseguridad en España | ED2026 | INCIBE. (s. f.). https://www.incibe.es/ed2026/talento-hacker/publicaciones/diagnostico-talento-ciberseguridad
REF2	Serra-Ruiz, J., & Pablo, G. P. (2023, 10 enero). Pentesting & Hacking Ético mediante resolución de un Capture The Flag (CTF). http://hdl.handle.net/10609/147381
REF3	Acronis Cyberthreats Report H2 2023 | Acronis Resource Center. (s. f.). Acronis. https://www.acronis.com/en-us/resource-center/resource/acronis-cyberthreats-report-h2-2023/
REF4	Guamán-Gómez, V. J., Chapa-Argudo, C. E., & Marín-Reyes, I. P. (2021). Importancia de los medios audiovisuales para la enseñanza y el aprendizaje. Revista Transdisciplinaria de Estudios Sociales y Tecnológicos, 1(2), 48-56.
REF5	CTF challenges - Hacking articles. (2024, 9 mayo). Hacking Articles. https://www.hackingarticles.in/ctf-challenges-walkthrough/
REF6	StartBootstrap. (s. f.). GitHub - StartBootstrap/startbootstrap-resume: A Bootstrap 4 resume/CV theme created by Start Bootstrap. GitHub. https://github.com/startbootstrap/startbootstrap-resume
REF7	Obfuscator.io. (s. f.). JavaScript Obfuscator Tool. https://obfuscator.io/
REF8	Samsar. (s. f.). GitHub - Samsar4/Ethical-Hacking-Labs: Practical Ethical Hacking Labs . GitHub. https://github.com/Samsar4/Ethical-Hacking-Labs/tree/master
REF9	StackHawk. (2021, 30 abril). Preventing Command Injection in PHP. StackHawk. https://www.stackhawk.com/blog/php-command-injection/
REF10	Moeinfatehi. (s. f.). GitHub - moeinfatehi/lfi-to-rce-scenario: This repository is a Dockerized php application containing a LFI (Local File Inclusion) vulnerability which can lead to RCE (Remote Code Execution). GitHub. https://github.com/moeinfatehi/lfi-to-rce-scenario
REF11	CS 4740/6740: Network Security. (s. f.-b). https://www.khoury.northeastern.edu/home/amirali/teaching/Summer14/lab/mitm.html
REF12	Orangetw. (s. f.). GitHub - orangetw/My-CTF-Web-Challenges: Collection of CTF Web challenges I made. GitHub. https://github.com/orangetw/My-CTF-Web-Challenges
REF13	GeoannyCode. (2022, 18 diciembre). ♡Vulnerabilidad en login con PHP🚀Ataque SQL INJECTION. DEV Community. https://dev.to/geoannycode/ejemplo-basico-de-sql-injection-9e1
REF14	Reverse Shell Cheat Sheet | pentestmonkey. (s. f.). https://pentestmonkey.net/cheat-sheet/shells/reverse-shell-cheat-sheet
REF15	Fernández, L. M. L., Cueto, E. G., & Álvaro, P. G. (2000). Relación entre motivación y aprendizaje. Psicothema, 12(Su2), 344-347.
REF16	Sharma, V. (2022b, enero 7). DNS Spoofing using BetterCap - Vikas Sharma - Medium. Medium. https://psychovik.medium.com/dns-spoofing-using-bettercap-24a8435f7a03
REF17	BePractical. (2022, 12 junio). DNS SPOOFING ATTACK USING ETTERCAP (2022) | BePractical [Vídeo]. YouTube. https://www.youtube.com/watch?v=4i7kc8cY654

9. APÉNDICE

La página web del proyecto está disponible en http://www.dguezprojects.es/

10. ANEXOS


web.rar


Formulario.zip