Universidad Internacional de La Rioja (UNIR)

ESIT

Máster universitario en Seguridad Informática

Herramienta para auditorías de seguridad en entornos industriales y SCADA

Trabajo Fin de Máster

presentado por: Lorenzo González, David

Director/a: Suárez Touceda, Diego

Ciudad: Logroño
Fecha: 06/02/2020

Agradecimientos

Quería agradecer especialmente a mi familia por haber llegado hasta aquí. Desde hace unos años llevo interesado mucho por la seguridad informática y a lo largo de todo este máster he aprendido muchísimas cosas que me apasionan y me motivan a seguir adquiriendo más conocimientos. Y es gracias a ellos por lo que he podido realizarlo.

También dar las gracias a mi pareja que es uno de mis pilares fundamentales y me ha apoyado y animado a lo largo de esta etapa.

Y por último, quería dar gracias a todos los profesores que he tenido a lo largo de este máster que, sin duda, son todos unos grandes profesionales y expertos en la materia. Llevan años trabajando en el sector y denotan grandes conocimientos que son capaces de transmitirnos a los alumnos. Y especialmente, agradecer a mi director por sus comentarios y correcciones en las distintas entregas ya que me han ayudado y motivado a terminarlo.

Gracias a todos.

Resumen

Realizar auditorías de seguridad o test de penetración es cada día más importante para las empresas, sobre todo para aquellas que tratan información confidencial o que disponen de infraestructuras críticas y sistemas industriales. Con el avance de las tecnologías, los sistemas de control industrial se encuentran conectados a Internet y pueden ser controlados remotamente, lo que puede poner en peligro a toda una población si estos no son protegidos adecuadamente, por ejemplo, el caso Stuxnet.

Con el desarrollo de este proyecto, se busca la implementación de una herramienta que ayude al auditor en las auditorías de seguridad informática en entornos industriales, ya que es un campo muy concreto y muchas de las pruebas típicas de Sistemas de Información no son aplicables aquí.

Para ello, se tratan dos de los protocolos más utilizados como son Modbus y Siemens S7 y se incorporan pruebas reconocidas para este tipo de test de penetración.

Palabras Clave

Pentesting, Sistemas de Control Industrial (ICS), auditoría de seguridad, Modbus, Siemens

Abstract

Performing vulnerability analysis and penetration tests is becoming more and more important for the companies, especially that ones which treats confidential information and offers industrial control systems. Nowadays, with new technologies, the Industrial Control Systems are connected to Internet and are accessible from everywhere. This could be an issue if they are not secure enough and could trigger a fatal disaster, for example, the case of Stuxnet.

The main purpose of the project is to implement a tool to help the auditor on his security audits at industrial systems. This is such a huge field that typical proofs in Information Technology cannot be applied here.

For this, the tool covers two of the protocols most used in the industry which are Modbus and Siemens S7 and it includes some recognized tests in penetration tests.

Keywords

Pentesting, Industrial Control System (ICS), security audit, Modbus, Siemens

Índice General

Índice de Figuras

Capítulo 1. Introducción

1.1. Justificación del Proyecto

Con la cuarta revolución industrial cada vez encontramos más dispositivos conectados a Internet, y más aún con el Internet de las Cosas (IoT), siendo capaz de controlar actuadores y otros elementos remotamente. Esto supone una posible brecha de seguridad, ya que cualquier atacante externo podría tomar el control del sistema sin importar la situación geográfica del mismo. No sirven de nada los controles previamente utilizados, ya que era el operario quien físicamente manejaba la máquina y no había forma de recibir ataques a través de Internet.

Cada vez se producen más ataques y la mayoría de este tipo de empresas no están preparadas para ello. Como se muestra en el siguiente gráfico, en 2011 se produjo un punto de inflexión y las vulnerabilidades han aumentado un 85% de un año para otro. Además, no dejan de subir más aún en la actualidad con el IoT.

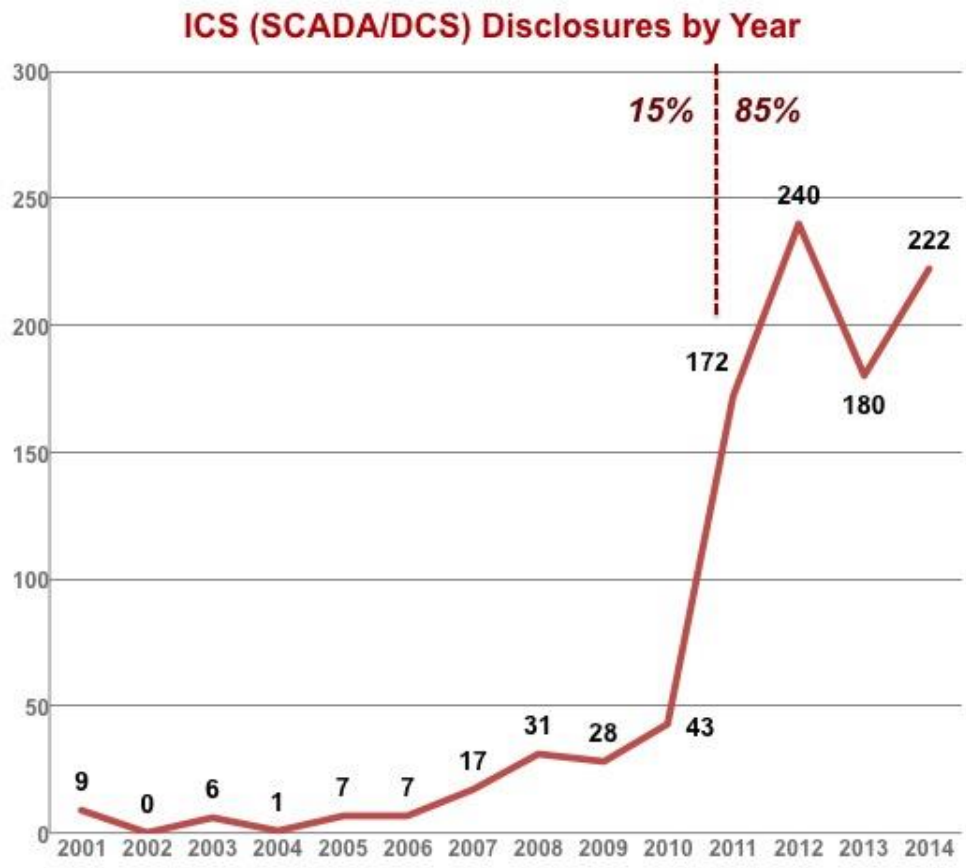

Figura 1.1. Vulnerabilidades ICS publicadas por año.

Basta con observar el número de dispositivos de control industrial conectados en la actualidad para darnos cuenta de la exposición a la que están sometidos, siendo de vital importancia la protección y securización de estos debido a que algunos son controles utilizados en centrales nucleares

llegando a darse desastres naturales. En la siguiente imagen, se muestran los dispositivos que utilizan el protocolo SIEMENS S7 (uno de los más usados en sistemas SCADA).

Figura 1.2. Dispositivos que utilizan Siemens S7 obtenido de Shodan

Por todo esto, es indispensable una herramienta que contenga un compendio de pruebas para realizar las distintas fases de una auditoría de seguridad y permita mejorar la seguridad de estos entornos industriales, desde la recolección de información hasta la explotación de las potenciales vulnerabilidades encontradas y su solución para mitigar el riesgo que puedan ocasionar.
En la actualidad, no existe una solución que permita automatizar las distintas fases y pruebas que se realizan en un pentesting o auditoría a este tipo de sistemas, únicamente alguna parte de esta.

1.2. Objetivos del Proyecto

El principal objetivo es crear un software que se ejecute por línea de comandos (similar de la herramienta SET, Social Engineer Toolkit, en cuanto a su interfaz, pero diferente a su funcionamiento), que facilite la tarea al auditor de seguridad en su trabajo y solamente tenga que seleccionar de un menú de opciones lo que desee testear.

No necesitará un grandísimo conocimiento de los elementos industriales (PLC, RTU, DCS, SCADA…), sino que bastará con tener alguna experiencia auditando otros sistemas TI típicos e interpretar la salida y los resultados de la herramienta. La salida será una descripción de los fallos de seguridad/vulnerabilidades encontradas, así como su explotación y su solución, al igual que se haría al finalizar una auditoría con la entrega del informe.

1.3. Estudio de la Situación Actual

1.3.1. Evaluación de Alternativas

1.3.1.1. Nessus

La herramienta dispone de plugins como *ICS/SCADA Smart Scanning* o incluso una solución completa que la llaman Tenable Industrial Security. Ambas posibilidades permiten escanear la red industrial como si se tratase de una estructura de TI normal y corriente y de esta forma obtener los fallos de seguridad y las vulnerabilidades existentes en cada uno de los dispositivos como habitualmente funciona Nessus. Soporta diversos protocolos, algunos son:

- Siemens S7
- Modbus
- BACnet
- Omron FINS
- Ethernet CIP
- 7T IGSS
- ICCP COTP

Proporciona un inventario actualizado de los sistemas y las aplicaciones, y de sus vulnerabilidades, para ayudar a las organizaciones a comprender la exposición cibernética de su tecnología operativa (OT) y proteger el rendimiento operativo. La solución, que fue diseñada especialmente para los sistemas de OT, utiliza el monitoreo pasivo para proporcionar información segura y confiable, de modo que sepa lo que tiene y lo que es vulnerable. Industrial Security, que cubre un amplio rango de ICS, SCADA y sistemas de TI tradicionales, ayuda a los equipos de seguridad de TI y OT, de operaciones de planta y de cumplimiento a mejorar la seguridad y la protección de los activos, y a reforzar el cumplimiento normativo.

Este escáner es muy completo también para el mundo OT, pero al ser de pago, no es accesible por todos los usuarios. También destacar que solamente realizaría la parte de análisis de vulnerabilidades dentro de una auditoría típica de seguridad y no llegaría a comprobar la explotación de estas para verificar si son realmente vulnerables o no.

1.3.1.2. Metasploit

Si de la explotación de cualquier sistema se trata, Metasploit es una de las mejores soluciones que existen hoy en día, por no decir la mejor. Sin embargo, en ICS no existen tantos exploits y módulos instalados por defecto en la herramienta y habría que añadir los que nos interesasen para el objetivo concreto. Sin embargo, sigue siendo una buena opción para la explotación de sistemas OT y podría ser un elemento adicional tras el análisis de vulnerabilidades y el resto de pruebas realizadas previamente en la auditoría.

1.3.1.3. Modbus-scanner

Es una utilidad gratuita que permite la comunicación con los dispositivos Modbus, tanto TCP/IP como en serie, y realizar diversas funciones, las dos más importantes son las siguientes:

- Leer datos de registros, bobinas, entradas…
- Descubrir dispositivos en la red

La herramienta tiene soporte desde Windows XP/Windows 2000, por lo que puede correr en casi cualquier dispositivo Windows. Dispone de una sencilla interfaz.

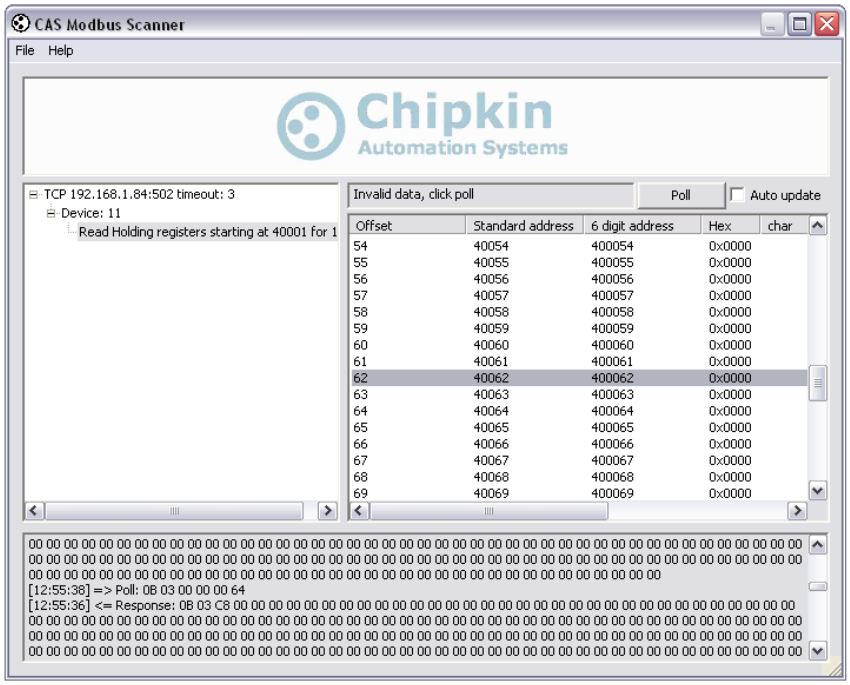

Figura 1.3. Interfaz Modbus-scanner

Es una herramienta útil para dispositivos Modbus pero algo limitada para realizar auditorías completas.

1.3.1.4. SMOD

Se trata de un framework que sirve para interactuar con sistemas Modbus y explotar algunas características de forma ofensiva. En cuanto a funcionamiento e interfaz es similar a Metasploit y tiene funciones interesantes. Es una buena herramienta para la parte de explotación a este tipo de sistemas, además de ser gratuita. Sin embargo, actualmente no está en desarrollo y no se ha añadido ningún cambio o mejora a su última versión desde hace unos años.

1.3.1.5. Modscan

Es una herramienta diseñada para descubrir la topología de una red basada en SCADA Modbus TCP. Está escrita en Python lo que le da portabilidad y que pueda ser ejecutada en cualquier plataforma.

1.3.2. Análisis de la normativa de la seguridad referente a sistemas de control industrial

La normativa de la seguridad referente a los sistemas de control se ha centrado principalmente en aspectos propios del funcionamiento de los dispositivos o a la especificación de protocolos de comunicación. Estas normativas tenían, además, la característica de ser abiertas en cuanto a la interpretación que podía realizarse de los mismas, por ejemplo, la especificación del protocolo IEC 61870-5-101 permite la implementación de diferentes perfiles incompatibles entre ellos y todos ellos cumpliendo el estándar.

1.3.2.1. ISA99

El grupo de normas ISA99 se crea pretendiendo dar un nuevo empuje a la seguridad de los sistemas de control industrial y creando para ello un conjunto de documentos que ayudasen al incremento de la protección de estos sistemas frente a ataques informáticos. Engloba una serie de guías e informes para definir los conceptos, términos y modelos que se han de utilizar, así como la descripción de los elementos necesarios para la implantación de un sistema de gestión de la ciberseguridad. Los documentos desarrollados son los siguientes:

- ANSI/ISA-99.01.01-2007 “Security for Industrial Automation and Control Systems: Concepts, Terminology and Models”.
- ANSI/ISA-TR99.01.02-2007 “Security Technologies for Manufacturing and Control Systems”.
- ANSI/ISA-99.02.01-2009 “Establishing an Industrial Automation and Control Systems Security Program”.
- ANSI/ISA-99.02.02 “Operating an industrial automation and control system security program”.
- ANSI/ISA–99.03.xx “Technical security requirements for industrial automation and control systems”.

1.3.2.2. IEC 62443

La norma **IEC 62443** es un conjunto de estándares que se basa en los conceptos definidos por la norma **ISA99**, previamente explicada. Es elaborada por el grupo TC65 de la IEC con la intención de mejorar y ampliar las capacidades de actuación y objetivos de ISA99. Está compuesta por 13 documentos:

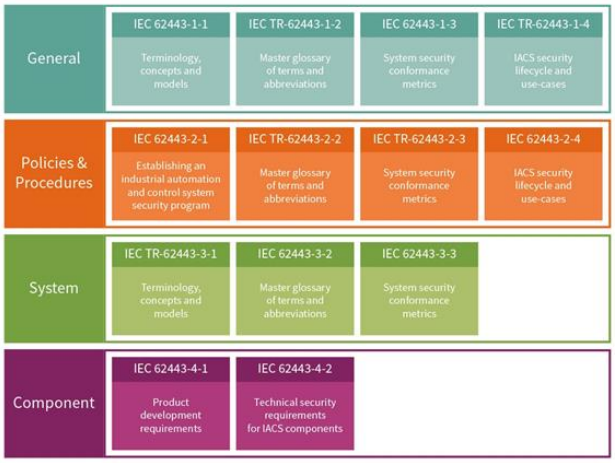

Figura 1.4. Partes de la norma IEC 62443

- **IEC/TS 62443-1-1:2009**: Especificación técnica que define la terminología, conceptos y modelos para los sistemas de control y automatización industrial. Es la actualización del documento ANSI/ISA-99.01.01-2007 de la ISA99.
- **IEC 62443-2-1:2010**: Este documento se corresponde con el documento ANSI/ISA-99.02.01-2009 publicado por la ISA 99, describiendo el sistema de gestión de la ciberseguridad para sistemas de control.
- **IEC/TR 62443-3-1:2009**: Renueva el informe técnico SI/ISA-TR99.01.02-2007 con nuevas herramientas de seguridad para los sistemas de control y automatización industrial (IACS).
- **IEC/PAS 62443-3-3:2008**: Especificación disponible para todo el público cuyo objetivo es la creación de un marco que asegure las tecnologías de comunicación y la información referentes a los procesos industriales.

1.3.2.3. NIST SP 800-82

Este documento proporciona una guía para implantar un sistema de control industrial (ICS) seguro, incluyendo sistemas SCADA, sistemas de control distribuido (DCS) u otros elementos tales como PLCs. Define topologías de sistemas típicos en control industrial e identifica amenazas y vulnerabilidades comunes en este tipo de sistemas, donde también proporciona la solución a estos y formas de mitigar su impacto.

1.3.2.4. NIST SP 800-53

El objetivo de esta publicación es proporcionar una guía sobre la seguridad y los controles de privacidad para los sistemas de información federales y organizaciones para mitigar diversos tipos de ciberataques, desastres naturales, fallos estructurales y errores humanos ante los que están expuestos. Los controles son personalizables e implementados como parte del proceso de una organización que maneja información crítica.

En unos de los apéndices del documento, se recogen una serie de controles de seguridad, diseñados para facilitar el cumplimiento con diversas leyes. El propio NIST publicó una serie de guías adicionales, ICS Supplemental Guidance, ICS Enhancements (one or more) e ICS Enhancement Supplemental Guidance, que ayudan e indican como han de aplicarse los controles publicados en NIST SP 800-53 sobre los sistemas de control industrial, así como información de cuáles aplican y cuáles no.

1.3.3. Investigación del estado de los sistemas de control industrial

Para comprobar el estado actual de algunos sistemas de control industrial basta con realizar alguna búsqueda especial utilizando filtros en el buscador *Shodan*. Por ejemplo, si buscamos los dispositivos que utilicen el protocolo Modbus, nos aparecen más de 20.000. Pero si queremos encontrar algún dispositivo vulnerable, podemos buscar versiones de sistemas operativos antiguos que no tengan soporte a día de hoy y haya vulnerabilidades conocidas como puede ser el caso de Windows XP. En este caso, encontramos los siguientes:

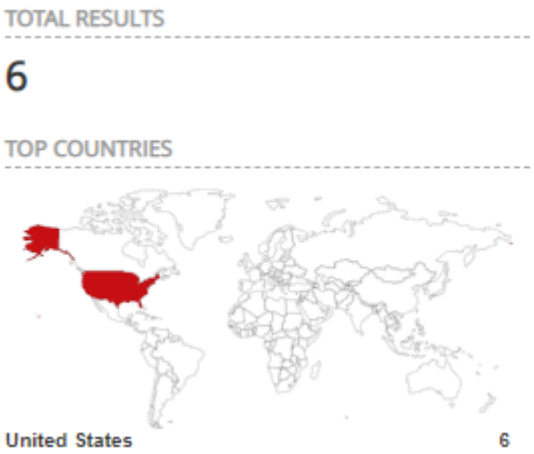

Figura 1.5. Dispositivos Modbus con Windows XP

Existen 6 dispositivos Modbus que utilizan Windows XP y que son fácilmente vulnerables, lo que hace que estén expuestos a posibles ataques.

O por ejemplo, si buscamos el protocolo DNP3 que también es ampliamente usado en entornos industriales, obtenemos que existen más de 30.000 dispositivos utilizándolo. Sin embargo, si filtramos por sistemas operativos de versiones antiguas, en este caso versiones de Linux 2.4 (la última actualización es de 2009) también vemos como existen algunas máquinas con este SO.

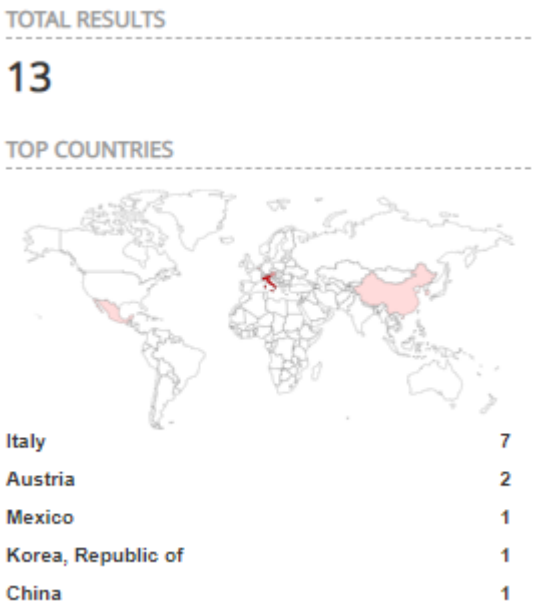

Figura 1.6. Dispositivos DNP3 con Linux 2.4

Esta versión de Linux, según la base de datos de CVEs del Mitre, tiene 57 vulnerabilidades conocidas. Por ello, es un objetivo vulnerable para cualquier atacante malicioso que quiera perpetrar el sistema y acceder a cualquier dato disponible en él o incluso si se trata de alguna máquina que controle mecanismos de centrales nucleares o eléctricas, puede llegar a suponer una catástrofe.

Por ello, es necesario disponer de una herramienta que pueda conocer las posibles vulnerabilidades a las que está expuesta y dar la mayor información posible acerca de la solución a todas ellas.

Capítulo 2. Aspectos Teóricos

2.1. ICS

ICS, por sus siglas en inglés Industrial Control Systems, es decir, Sistemas de Control Industrial, se trata de un término asociado a diversos sistemas de control utilizados en los procesos industriales. Estos sistemas pueden abarcar desde unos pocos controladores hasta enormes sistemas de control distribuidos con miles de elementos interconectados. A su vez, existen diversos tipos de Sistemas de Control Industrial como HMI, PLC o SCADA que se describirán en los siguientes puntos.

2.2. Clasificación de los Sistemas Industriales

ISA 95 define 5 niveles para la clasificación de los componentes en este tipo de sistemas:

Figura 2.1. Clasificación de los Sistemas Industriales

- **Nivel 0**: Proceso de producción físico.
- **Nivel 1**: Sensores, los cuales manipulan y actúan sobre el proceso de producción.
- **Nivel 2**: Monitorización, supervisión y control de los procesos.
- **Nivel 3**: Control de flujos u órdenes para crear los productos finales, mantenimiento de registros y optimización del proceso de producción.
- **Nivel 4**: Destinado a la alta gestión con procesos como logística y económicos facilitando información a la dirección ejecutiva.

2.2.1. PLC

Programmable Logic Controller, en español autómatas programables, realizan acciones definidas en el programa que contiene su memoria basada en las entradas que recibe. Estas entradas pueden ser de varios tipos:

- Sensores de temperatura, presión, humedad…
- Detectores de posición, movimiento, luz…

2.2.2. RTU

Remote Terminal Unit se trata de un dispositivo compuesto por microprocesadores que controla los interfaces con instrumentos de proceso y actuadores. Incluye la recolección de datos de telemetría en los sistemas. Una de sus principales características es que transmiten la información a lo largo de grandes distancias hasta una unidad central. Por ello, suelen ser utilizados en infraestructuras con gran dispersión geográfica.

2.2.3. MTU

Master Terminal Unit. En este grupo entran los servidores y el software responsable de comunicarse con los equipos de campo (RTU, PLC…). Este terminal ejecuta acciones programadas en función de los valores recibidos de los sensores. Esta se realiza en lenguajes de alto nivel como C o Basic, además de almacenar datos para ser accesibles por otras aplicaciones.

Entre las funciones más importantes de los MTU se encuentran:

- Adquisición de datos de los RTU.
- Salvaguarda de datos y puesta a disposición de los operadores.
- Procesamiento de alarmas.
- Control.
- Visualización de datos de campo.
- Creación de informes.
- Gestión de interfaces con otros sistemas.
- Administración de la red.

2.2.4. HMI

Human Machine Interface. Es el entorno visual que permite la interacción del ser humano con los medios tecnológicos implementados. Es una manera sencilla y estandarizada de supervisar los múltiples elementos: RTUs, PLCs…

Como parte del HMI también se integran las bases de datos donde los datos recogidos en otros dispositivos serán almacenados permitiendo la creación de gráficas, así como datos de logística, mantenimiento, etc. para facilitar la localización de averías.

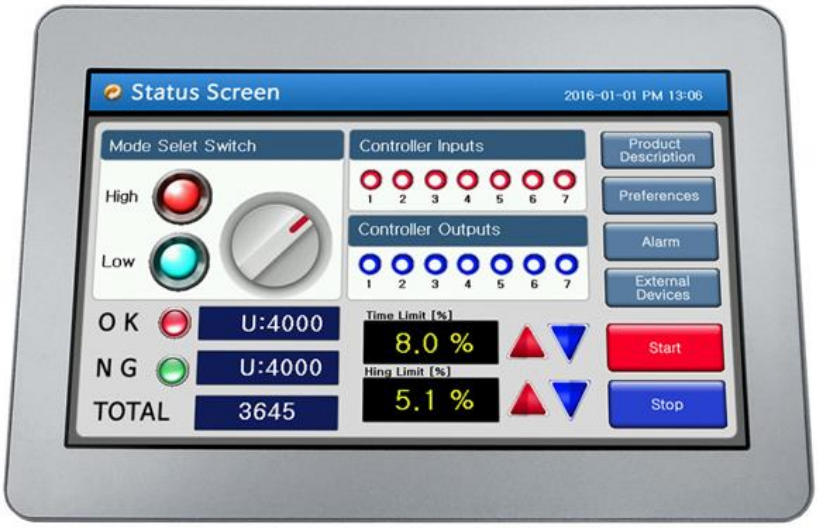

Figura 2.2. Human Machine Interface

2.2.5. SCADA

Supervisory Control And Data Acquisition. Se trata de un sistema que supervisa y controla datos a distancia basados en la información recogida en los RTUs y PLCs. Estos sistemas utilizan ordenadores y redes de comunicación para el monitoreo y control de sistemas industriales. Hay de diferentes tipos según su utilización en la industria, pero todos tienen en común la dispersión geográfica. Esto es debido a todos los sensores, PLCs… que se encuentran distribuidos geográficamente en una gran extensión. Los sistemas SCADA soportan diversos protocolos como pueden ser Modbus, DNP3, BACnet, etc. Dependiendo de la finalidad, se utilizará uno u otro, pero en el documento se hará mayor hincapié en Modbus por ser uno de los más populares.

Desde el punto de vista del auditor, el área de ataque es determinado por el diagrama de la arquitectura de red y lo bien segregado que esté del resto de la red. Puede darse el caso de que la red se encuentre aislada o que esté compartida con la red corporativa, siendo este último lo más común en los sistemas industriales. A continuación, se muestra un diagrama de red típico en estos casos:

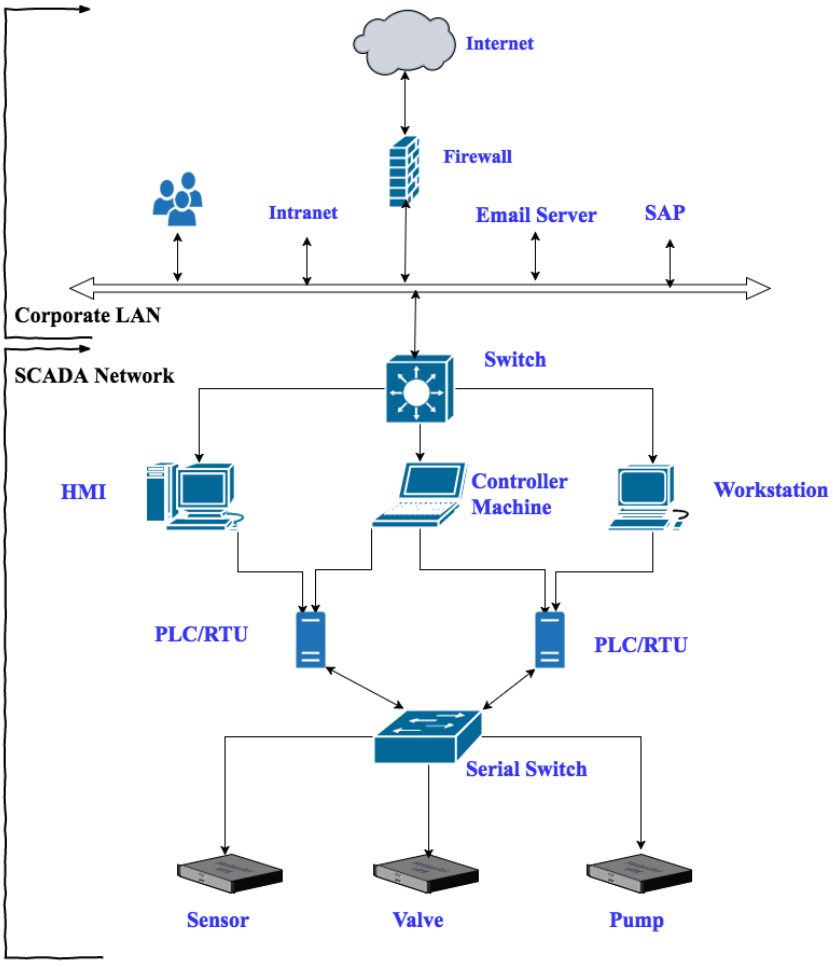

Figura 2.3. Diagrama de red SCADA

2.3. Protocolos de comunicación en sistemas industriales

2.3.1. Modbus

Modbus es uno de los protocolos más antiguos y más ampliamente usados en la industria. Se trata de un protocolo a nivel de aplicación diseñado para comunicación con los controladores de campo y debido a su popularidad la mayoría de controladores lo soportan. Inicialmente fue creado por la compañía Schneider Electrics, pero en la actualidad es gestionado por la organización Modbus.

Existen tres implementaciones ampliamente usadas que son las siguientes:

- TCP/IP
- RTU
- ASCII

Nos centraremos en la primera que es el usado a través de redes TCP/IP que es el más comúnmente utilizado. La capa física no está especificada, por lo que no está limitada por un único tipo de conector.

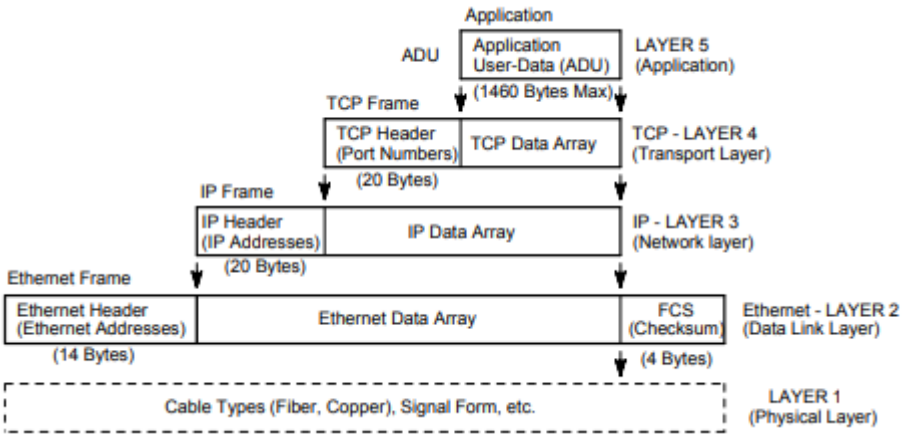

Figura 2.4. Paquete de datos protocolo Modbus TCP/IP

2.3.2. Siemens S7

La implementación del protocolo S7, base de las comunicaciones entre dispositivos SIEMENS, se basa en el servicio de transporte ISO orientado a bloques. El protocolo S7 está envuelto en los protocolos TPKT e ISO-COTP, lo que permite que la PDU (Unidad de Datos de Protocolo) se transporte a través de TCP. La comunicación ISO sobre TCP se define en RFC1006, la ISO-COTP se define en RFC2126 que se basa en el protocolo ISO 8073 (RFC905) y titulada como “ISO Transport Service on top of TCP”.

El protocolo S7 está orientado a funciones o comandos, lo que significa que una transmisión consiste en una solicitud S7 y una respuesta apropiada (con muy pocas excepciones). El número

de la transmisión en paralelo y la longitud máxima de una PDU se negocia durante la configuración de la conexión.

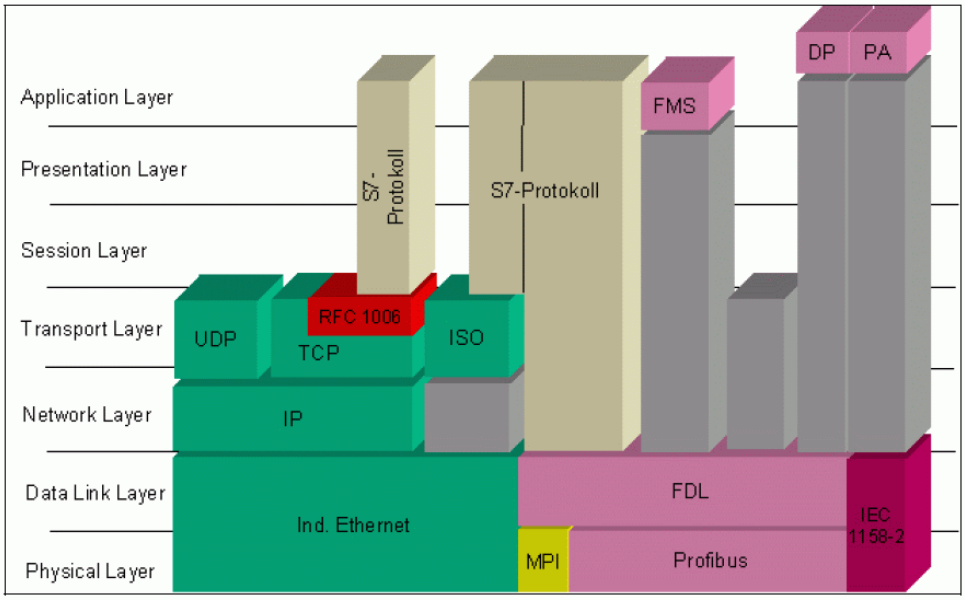

Figura 2.5. Protocolo Siemens S7

Capítulo 3. Planificación del Proyecto y Resumen del Presupuesto

3.1. Planificación

Para la realización del proyecto se han definido una serie de **hitos** que se irán completando a lo largo del proyecto y de esta forma, conocer el estado y el avance del mismo. Son los siguientes:

Hitos
Preparación propuesta TFM entregada
Propuesta del TFM entregada
Tema del TFM adjudicado
Situación actual estudiada
Equipo formado en seguridad
Arquitectura del Sistema diseñada
Interfaz creada
Implementación completada
Pruebas unitarias realizadas
Pruebas en entornos simulados realizadas

Elaboración de la documentación
Manuales del sistema creados
Entrega del proyecto
Preparación de la presentación
Presentación del proyecto
Cierre del proyecto

Tras la definición de estos hitos, se crearon las tareas para poder alcanzarlos a lo largo del desarrollo del proyecto. Dichas tareas están agrupadas en función del hito a cumplir y se ha estimado el tiempo necesario para llevarlas a cabo. Por tanto, la duración total del proyecto se ha estimado en 295 horas.

Tareas	Duración
Preparación propuesta TFM	**15 horas**
Selección tema	4 horas
Definición del alcance	4 horas
Objetivo del proyecto	4 horas
Documentación formulario TFM	3 horas
Estudio de la situación actual	**17 horas**
Evaluación de alternativas	4 horas
Investigación del estado de los sistemas de control industrial	5 horas
Análisis de la normativa actual referente a sistemas de control industrial	8 horas
Formación en seguridad del equipo	**22 horas**
Formación en sistemas de control industrial y sus protocolos	10 horas
Formación en herramientas y estándares para sistemas industriales	12 horas
Diseño de la Arquitectura del Sistema	**34 horas**
Obtención de requisitos	6 horas
Diseño de los diagramas de clases	8 horas
Creación de casos de uso	8 horas
Elaboración de escenarios de casos de uso	7 horas
Diseño de los diagramas de paquetes	5 horas
Creación de la Interfaz	**2 horas**

Implementación	**78 horas**
Desarrollo módulo de *Footprinting*	16 horas
Desarrollo módulo de *Fingerprinting*	35 horas
Desarrollo módulo de *Explotación*	12 horas
Desarrollo módulo de *Reportes*	15 horas
Pruebas unitarias	**38 horas**
Configuración de herramienta de integración continua	1 hora
Desarrollo pruebas para el módulo de *Footprinting*	10 horas
Desarrollo pruebas para el módulo de *Fingerprinting*	15 horas
Desarrollo pruebas para el módulo de *Explotación*	5 horas
Desarrollo pruebas para el módulo de *Reportes*	7 horas
Pruebas en entornos simulados	**12 horas**
Realización de pruebas en un entorno Modbus	6 horas
Realización de pruebas en un entorno Siemens	6 horas
Elaboración de la documentación	**45 horas**
Creación de manuales del sistema	**14 horas**
Creación de manual de instalación	2 horas
Creación de manual de ejecución	2 horas
Creación de manual de usuario	5 horas
Creación de manual del programador	5 horas
Entrega del proyecto	**1 hora**
Preparación de la presentación	**14 horas**
Presentación del proyecto	**1 hora**
Cierre del proyecto	**2 horas**

Id	Modo de tarea	Nombre de tarea	Duración	Hito
1		**Preparación propuesta TFM**	**15 horas**	**No**
6		Preparación propuesta TFM entregada	0 horas	Sí
7		**Estudio de la situación actual**	**17 horas**	**No**
11		Situación actual estudiada	0 horas	Sí
12		**Formación en seguridad del equipo**	**22 horas**	**No**
15		Equipo formado en seguridad	0 horas	Sí
16		**Diseño de la Arquitectura del Sistema**	**34 horas**	**No**
22		Arquitectura del Sistema diseñada	0 horas	Sí
23		Creación de la Interfaz	2 horas	No
24		Interfaz creada	0 horas	Sí
25		**Implementación**	**78 horas**	**No**
30		Implementación completada	0 horas	Sí
31		**Pruebas unitarias**	**38 horas**	**No**
37		Pruebas unitarias realizadas	0 horas	Sí
38		**Pruebas en entornos simulados**	**12 horas**	**No**
41		Pruebas en entornos simulados realizada	0 horas	Sí
42		Elaboración de la documentación	45 horas	No
43		Elaboración de la documentación	0 horas	Sí
44		**Creación de manuales del sistema**	**14 horas**	**No**
50		Entrega del proyecto	1 hora	No
51		Entrega del proyecto	0 horas	Sí
52		Preparación de la presentación	14 horas	No
53		Preparación de la presentación	0 horas	Sí
54		Presentación del proyecto	1 hora	No
55		Presentación del proyecto	0 horas	Sí
56		Cierre del proyecto	2 horas	No
57		Cierre del proyecto	0 horas	Sí

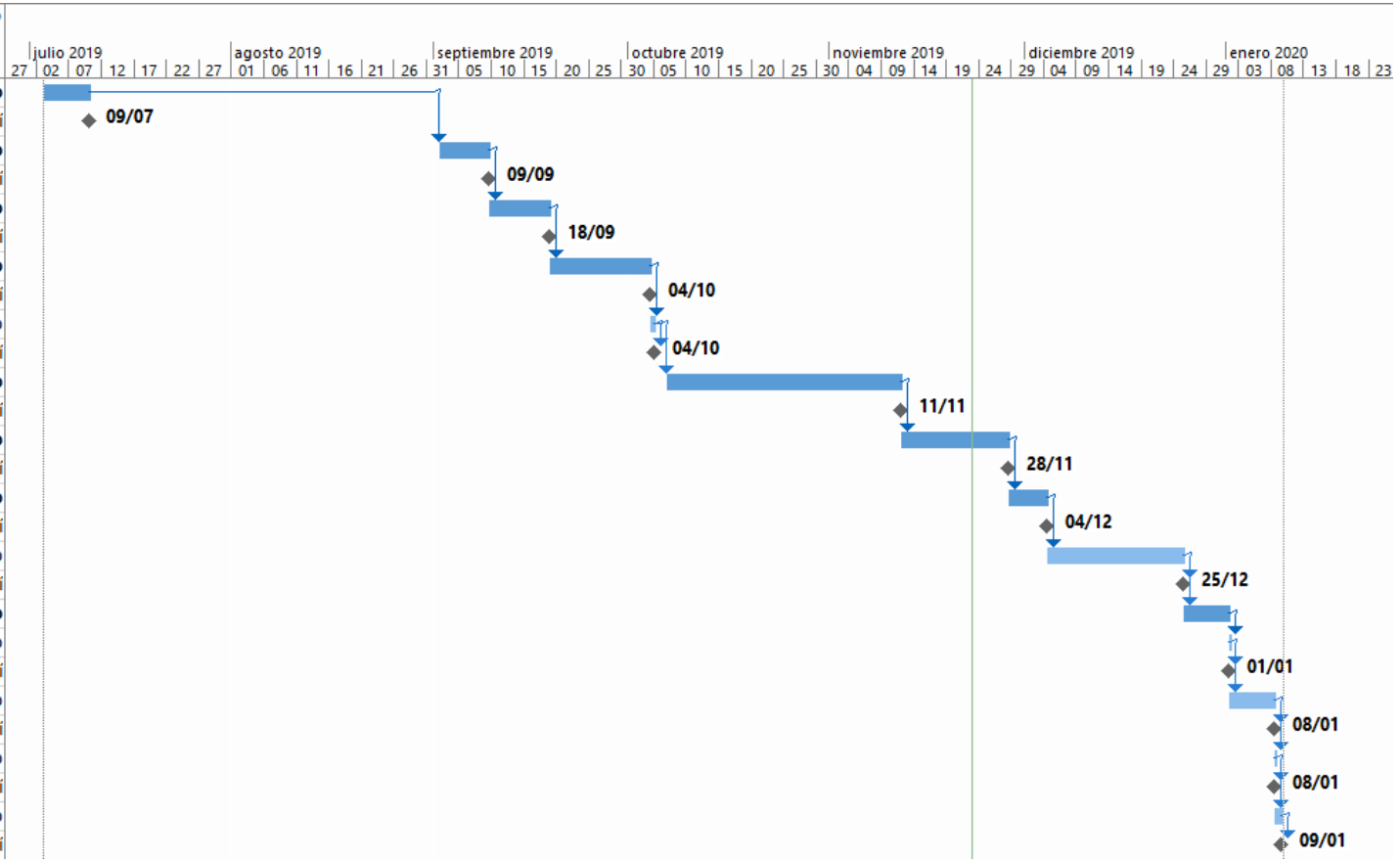

Figura 3.1. Planificación del Proyecto

3.2. Resumen del Presupuesto

El siguiente presupuesto es el del cliente, todos los detalles seguidos para su cálculo se encuentran descritos en el capítulo 10 Presupuesto:

Concepto	Precio
Preparación propuesta TFM	723,94€
Estudio de la situación actual	820,46€
Formación en seguridad del equipo	1061,78€
Diseño de la Arquitectura del Sistema	1640,93€
Creación de la Interfaz	96,53€
Implementación	3764,48€
Pruebas unitarias	1833,98€
Pruebas en entornos simulados	579,15€
Elaboración de la documentación	2171,81€
Creación de manuales del sistema	675,68€
Total	**13368,71€**

Capítulo 4. Análisis

Este apartado contendrá toda la especificación de requisitos y toda la documentación del análisis de la aplicación, a partir de la cual se elaborará posteriormente el diseño.

4.1. Definición del Sistema

4.1.1. Determinación del Alcance del Sistema

El alcance del proyecto estará acotado por la falta de tiempo y de un equipo completo de desarrollo para la implementación de un sistema que comprenda la mayoría de los protocolos industriales actuales y todos los elementos de una red de este tipo. Por ello, a lo largo del proyecto nos centraremos en dos de los protocolos más utilizados en los dispositivos que son Modbus y Siemens S7.

En el Radar de Sistemas de Control Industrial de Shodan, se puede observar todos los dispositivos industriales que están conectados a Internet y el protocolo que utilizan. En la siguiente imagen se ven los que están conectados, diferenciando entre los dispositivos, las honeypots y los crawlers de Shodan:

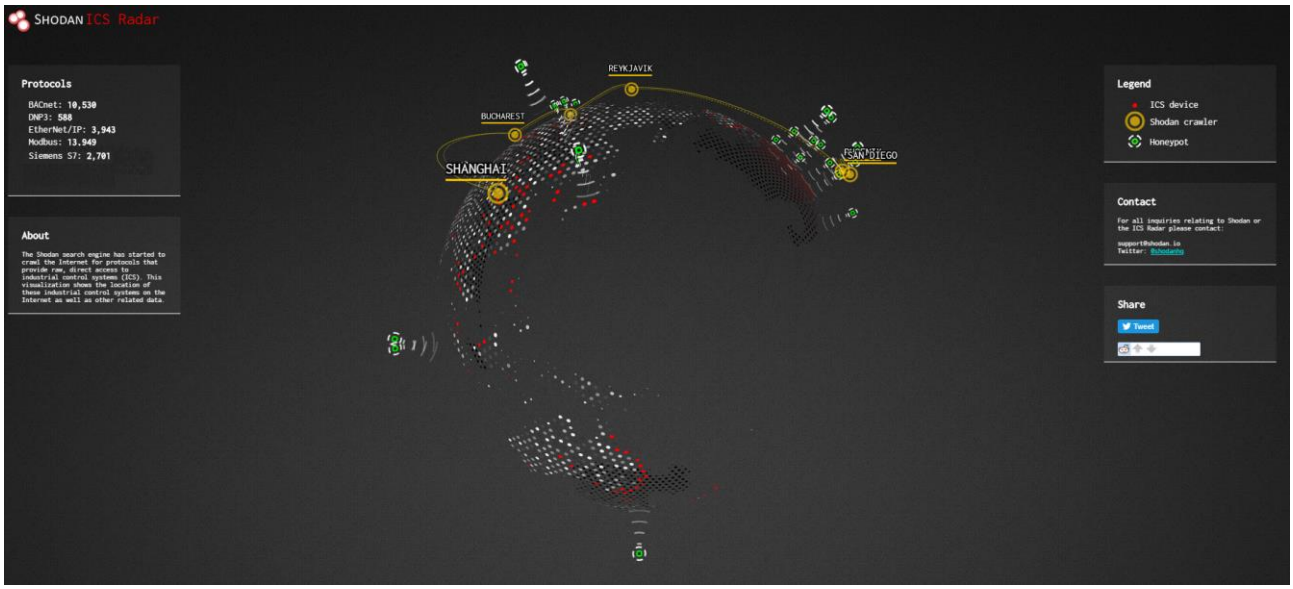

Figura 4.1. Mapa con los protocolos ICS utilizados obtenido de Shodan ICS Radar

Si nos fijamos en los protocolos, el más utilizado es Modbus. Por tanto, será en el que más hincapié se haga en el proyecto. Y también se estudiará el protocolo de Siemens por ser una de las marcas más importantes en el sector. Para futuras ampliaciones, se podría tener en cuenta BACnet y Niagara (que no aparece ahí pero también es ampliamente usado).

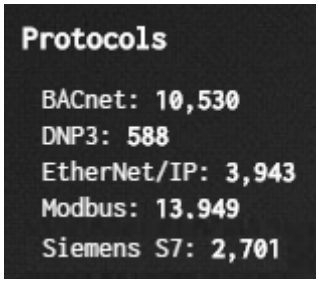

Figura 4.2. Protocolos ICS utilizados obtenido de Shodan ICS Radar

Y después, nos centraremos en sistemas SCADA, debido a que son los más vulnerables y expuestos a ciberataques. Según un estudio de Kaspersky:

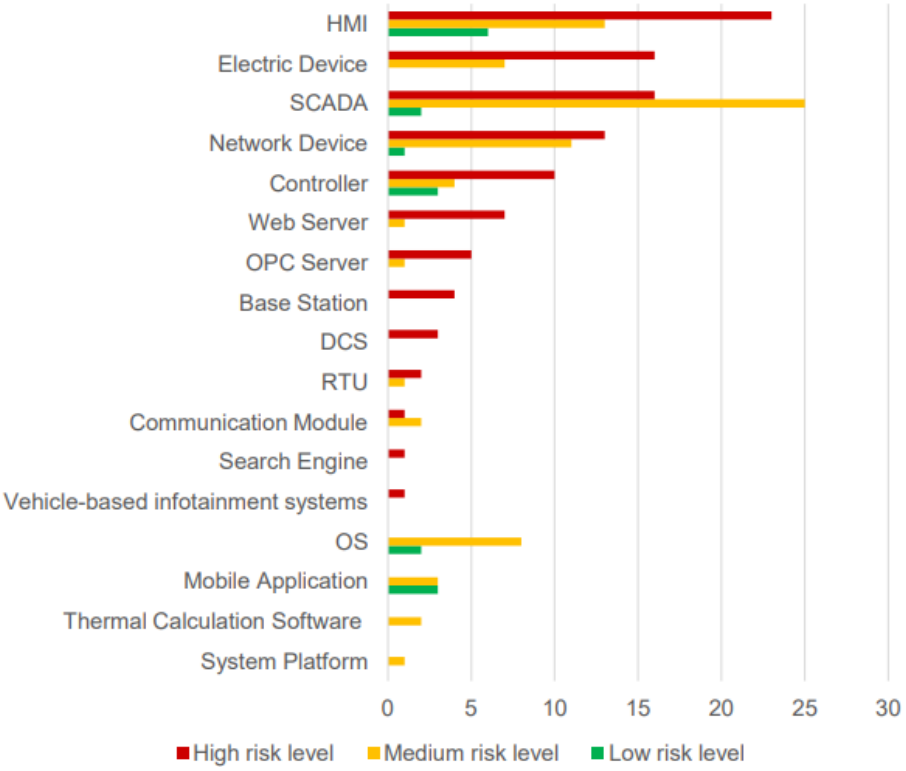

Figura 4.3. Estudio de las vulnerabilidades por tipo de dispositivo industrial (Kaspersky)

4.2. Requisitos del Sistema

4.2.1. Requisitos Funcionales

Código	Descripción
R1	El sistema debe permitir al usuario automatizar las tareas de un test de penetración para sistemas de control industrial.
R2	El sistema debe diferenciar las siguientes fases del test: • Footprinting • Fingerprinting • Explotación • Reportes
R3	Dentro de la etapa de *Footprinting*, el sistema debe poder llevar a cabo lo siguiente:
R3.1	El sistema debe recoger la información pública de la empresa que el usuario introduzca su dirección IP: • Nombre de la organización • País • Ciudad

	• Coordenadas geográficas
R3.1.1	El sistema debe validar la dirección IP introducida por el usuario según el requisito R7.
R3.1.2	El sistema debe realizar dichas consultas a través del buscador *Shodan*.
R3.1.3	El sistema debe validar los resultados que no han sido posibles obtener para mostrar solamente aquellos no nulos.
R3.2	El sistema debe consultar los registros DNS (*DNS Lookup*) del nombre de dominio introducido por el usuario.
R3.2.1	El sistema debe validar el nombre de dominio según R8 introducido por el usuario antes de realizar la consulta.
R3.3	El sistema debe realizar la consulta a los DNS inversos de la dirección IP o nombre de dominio introducido por el usuario para obtener posibles nuevos dominios que sean interesantes para la auditoría.
R3.3.1	El sistema debe validar el valor introducido por el usuario para la dirección IP según R7 o nombre de dominio objetivo según R8.
R3.4	El sistema debe consultar los registros *Whois* para conocer detalles de la organización, así como del proveedor de servicios Internet (ISP), de la dirección IP o nombre de dominio introducido por el usuario.
R3.4.1	El sistema debe validar el valor introducido por el usuario para la dirección IP según el requisito R7 o nombre de dominio objetivo según R8.
R3.5	El sistema debe obtener las propiedades de la subred de la dirección IP de la organización introducida por el usuario.
R3.5.1	El sistema debe validar dicha dirección IP según el requisito R7.
R3.6	El sistema debe comprobar la configuración del DNS del dominio dado por el usuario a través de una transferencia de zona.
R3.6.1	El sistema debe validar el nombre de dominio introducido según R8.
R4	Dentro de la etapa de *Fingerprinting*, el sistema debe poder llevar a cabo lo siguiente:
R4.1	El sistema debe hallar los puertos abiertos de la máquina que tiene como dirección IP la dada como parámetro por el usuario.
R4.1.1	El sistema debe validar dicha dirección IP según el requisito R7.
R4.2	El sistema debe realizar un escaneo sobre los PLCs que están corriendo en la máquina objetivo de la dirección IP pasado como parámetro por el usuario.
R4.2.1	El sistema debe validar la dirección IP según R7.

R4.3	El sistema debe permitir realizar las siguientes operaciones sobre los dispositivos que utilicen el protocolo Modbus.
R4.3.1	El sistema debe descubrir y recoger toda la información posible acerca de los esclavos del protocolo Modbus que estén corriendo en la dirección del objetivo.
R4.3.1.1	El sistema debe validar la dirección IP introducida por el usuario según R7.
R4.3.2	El sistema debe descubrir los UIDs de Modbus que han sido habilitados en el objetivo que tenga por dirección IP la dada por el usuario.
R4.3.2.1	El sistema debe validar la dirección IP introducida por el usuario según R7.
R4.3.3	El sistema debe enumerar las funciones del protocolo Modbus que tenga habilitadas en la máquina objetivo que sea la dirección IP dada por el usuario.
R4.3.3.1	El sistema debe validar la dirección IP introducida por el usuario según R7.
R4.3.4	El sistema debe obtener el valor de los registros (holding registers) disponibles en uno del esclavo que pida el usuario en la dirección IP dada.
R4.3.4.1	El sistema debe permitir escoger al usuario el UID del esclavo a leer el registro y validar la entrada según R9.
R4.3.4.2	El sistema debe permitir escoger al usuario la dirección del registro a leer y validar la entrada según R9.
R4.3.4.3	El sistema debe validar la dirección IP dada por el usuario según R7.
R4.3.5	El sistema debe obtener el valor de las bobinas (coils) disponibles en uno de los esclavos que pida el usuario de la dirección IP dada.
R4.3.5.1	El sistema debe permitir escoger al usuario el UID del esclavo a leer el registro y validar la entrada según R9.
R4.3.5.2	El sistema debe permitir escoger al usuario la dirección de la bobina a leer y validar la entrada según R9.
R4.3.5.3	El sistema debe validar la dirección IP dada por el usuario según R7.
R4.3.6	El sistema debe obtener el estado de las entradas discretas (discrete inputs) del esclavo que pida el usuario en de la dirección IP dada.
R4.3.6.1	El sistema debe permitir escoger al usuario el UID del esclavo a leer el registro y validar la entrada según R9.
R4.3.6.2	El sistema debe permitir escoger al usuario la dirección de la entrada a leer y validar la entrada según R9.

R4.3.6.3	El sistema debe validar la dirección IP dada por el usuario según R7.
R4.3.7	El sistema debe obtener el contenido de los registros de entrada (input registers) del esclavo que pida el usuario en de la dirección IP dada.
R4.3.7.1	El sistema debe permitir escoger al usuario el UID del esclavo a leer el contenido del registro y validar la entrada según R9.
R4.3.7.2	El sistema debe permitir escoger al usuario la dirección de la entrada a leer y validar la entrada según R9.
R4.3.7.3	El sistema debe validar la dirección IP dada por el usuario según R7.
R4.3.8	El sistema debe obtener el resultado de la respuesta de una excepción (exception response) por parte del esclavo que pida el usuario en de la dirección IP dada.
R4.3.8.1	El sistema debe permitir escoger al usuario el UID del esclavo a leer obtener la excepción y validar la entrada según R9.
R4.3.8.2	El sistema debe validar la dirección IP dada por el usuario según R7.
R4.4	El sistema debe permitir realizar las siguientes operaciones sobre los dispositivos que utilicen el protocolo Siemens S7.
R4.4.1	El sistema debe recoger toda la información posible acerca de los dispositivos PLC existentes en la red Siemens S7 que tenga como dirección IP la dada por el usuario.
R.4.4.1.1	El sistema debe validar la dirección IP dada por el usuario según R7.
R4.4.2	El sistema debe enumerar los PLCs activos en una red que utiliza el protocolo Siemens S7 dada como parámetro por el usuario.
R4.4.2.1	El sistema debe validar la dirección IP introducida por el usuario según R7.
R4.4.3	El sistema debe obtener información acerca de cada PLC encontrado en la red objetivo que el usuario introdujo como parámetro.
R4.4.3.1	El sistema debe validar la dirección IP introducida por el usuario según R7.
R5	Dentro de la etapa de *Exploitation*, el sistema debe poder llevar a cabo lo siguiente:
R5.1	El sistema debe mostrar al usuario un conjunto de contraseñas por defecto de sistemas típicos de control industrial, así como el dispositivo y la empresa propietaria de este.
R5.2	El sistema debe permitir escribir los registros (holding registers) de un UID del esclavo de Modbus que estén habilitados en la dirección IP pasada como parámetro por el usuario.

R5.2.1	El sistema debe permitir escoger al usuario el UID del esclavo a leer.
R5.2.2	El sistema debe permitir escoger al usuario la dirección del registro a leer.
R5.2.3	El sistema debe validar la dirección IP introducida por el usuario según R7.
R5.3	El sistema debe permitir escribir sobre los valores de las bobinas (coils) de un UID del esclavo de Modbus que estén habilitados en la dirección IP pasada como parámetro por el usuario.
R5.3.1	El sistema debe permitir escoger al usuario el UID del esclavo a leer.
R5.3.2	El sistema debe permitir escoger al usuario la dirección del registro a leer.
R5.3.3	El sistema debe validar la dirección IP introducida por el usuario según R7.
R5.4	El sistema debe extraer los *hashes* de la contraseñas que estén almacenadas en ficheros del tipo PLF.
R5.4.1	El sistema debe permitir al usuario introducir la ruta al fichero con extensión .PLF.
R6	Dentro de la etapa de *Reports*, el sistema debe poder llevar a cabo lo siguiente:
R6.1	El sistema debe permitir la creación de un informe sobre las pruebas llevadas a cabo por el usuario y sus resultados.
R6.2	El sistema debe mostrar al usuario los pasos que realizó y el resultado de cada uno en el informe.
R7	El sistema debe validar la dirección IP según el protocolo IPv4.
R7.1	El sistema deberá comprobar que: - Los valores están separados por un punto y se repiten 4 veces del modo <X.X.X.X>. - Los números están comprendidos entre 0 y 255 (ambos incluidos).
R8	El sistema debe validar la dirección IP según el formato <nombre.extensión>.
R8.1	El sistema debe realizar la comprobación con todo el nombre en minúsculas.
R8.2	El sistema debe comprobar que solamente se permitan los caracteres: - a-z - 0-9 - - (guión) - ! " # $ % & ' ( ) * + , - / : ; < = > ? @ [\ ] ^ _ ` { | } ~ .

R9	El sistema debe validar los UIDs, registros y direcciones de Modbus de la siguiente forma:
R9.1	Los esclavos o UIDs de Modbus que sean introducidos por el usuario deberán ser: - Un número entero positivo. - Comprendido entre 0 y 255 (ambos incluidos).
R9.2	Las direcciones de memoria de los registros que sean introducidos por el usuario deberán ser: - Un número entero positivo. - Comprendido entre 0 y 65535 (ambos incluidos).
R9.3	Los valores de los estados, tanto para las bobinas (coils) como para las entradas discretas (discrete inputs), que sean introducidos por el usuario deberán ser dos posibles valores: - ON - OFF

4.2.2. Requisitos No Funcionales

Requisitos de usuario

Código	Descripción
RU1	Los usuarios de la aplicación no deberán tener un alto conocimiento en seguridad informática para llevar a cabo las pruebas, únicamente para interpretar algunos de los resultados.
RU2	Los administradores de la aplicación serán expertos en seguridad encargados de mantenerla y actualizarla.

Requisitos tecnológicos

Código	Descripción
RT1	El sistema debe ser multiplataforma y poder ser ejecutado en cualquier entorno.

Requisitos de fiabilidad

Código	Descripción
RF1	El sistema debe garantizar que los resultados obtenidos no presenten falsos positivos.

4.3. Identificación de los Actores del Sistema

Existen dos tipos de roles diferenciados:

- El **usuario** final de la aplicación que su propósito será realizar test de penetración y pruebas de seguridad sobre sus sistemas de control industrial u otros ajenos (siempre y cuando se tenga el permiso necesario) como si de una auditoría se tratase.
- El **administrador** del sistema que será el encargado de mantener la aplicación y actualizarla con nuevas técnicas que surjan a lo largo del tiempo.

4.4. Identificación de los Módulos en la Fase de Análisis

El proyecto se ha dividido en cuatro módulos para encapsular cada una de las etapas de un test de penetración habitual y así, facilitar el uso a los usuarios como auditores.

4.4.1. Footprinting

El footprinting es la primera etapa de un test de intrusión en la que el atacante recoge la mayor cantidad posible de información pública sobre el objetivo. Su fuente es todo Internet. Se utilizarán buscadores como Shodan, información alojada en servidores DNS, en registros Whois y demás servicios en el que conocer datos acerca del objetivo.

4.4.2. Fingerprinting

En esta parte, se lleva a cabo una recolección de información que consiste en interactuar directamente con el sistema objetivo para conocer más datos sobre él, su configuración y comportamiento. A diferencia de la primera parte, aquí estamos interactuando directamente sobre el objetivo y puede detectar que estamos realizando alguna acción maliciosa. Por ello, para la realización de las pruebas, se hará en entornos locales y controlados, ya que por la Ley de Hacking de 2010 cualquier acción de este tipo contra sistemas sin su permiso expreso es ilegal.

4.4.3. Explotación

Esta fase del pentesting, tiene como objetivo comprobar la seguridad del sistema descubierta en la fase anterior y verificar si las vulnerabilidades realmente pueden ser explotables. Aquí, se realizarán modificaciones en el sistema final e incluso intrusiones aprovechándose de alguno de los fallos de seguridad detectados en fases previas.

4.4.4. Reportes

Por lo general, al finalizar todas las pruebas, se entrega al cliente un informe con todos los pasos seguidos detalladamente, así como una serie de resultados sobre las vulnerabilidades encontradas y la solución a las mismas. En el proyecto, se intentará automatizar este proceso y generar un reporte sobre las pruebas que realizó el usuario a lo largo del proceso y los resultados obtenidos.

4.5. Especificación de Casos de Uso

4.5.1. Casos de Uso del módulo de Footprinting

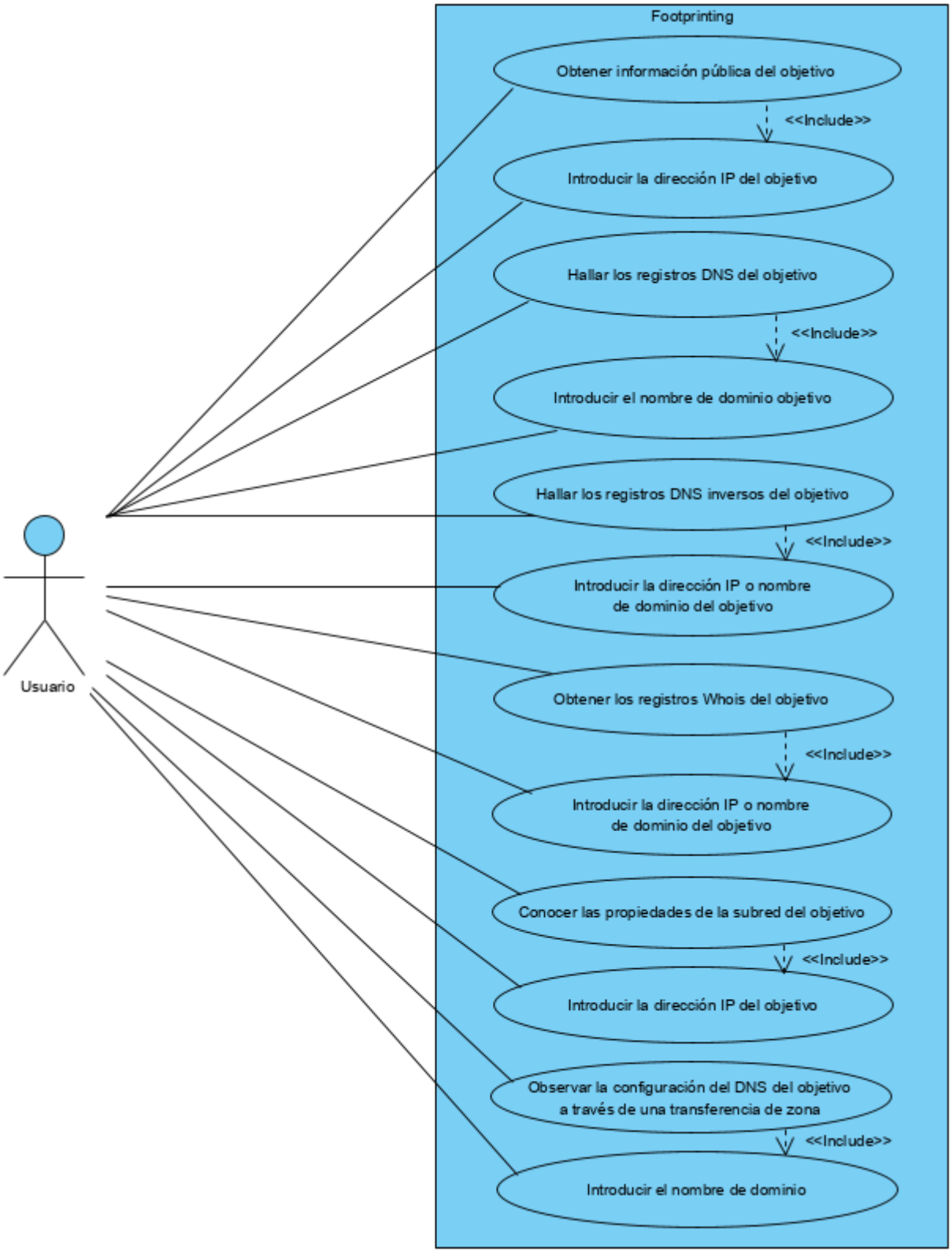

Figura 4.4. Casos de uso para el módulo de Footprinting

Nombre del Caso de Uso
Escoger una opción del menú del módulo
Descripción
El usuario podrá elegir de entre todas las opciones del menú la que él desee para realizar la prueba oportuna.

Nombre del Caso de Uso
Obtener información pública del objetivo
Descripción
Al elegir esta opción, el usuario obtendrá información pública de la empresa expuesta en buscadores como Shodan.

Nombre del Caso de Uso
Introducir dirección IP objetivo
Descripción
El usuario introducirá una dirección IP según el protocolo IPv4 que será el objetivo a obtener la información pública.

Nombre del Caso de Uso
Hallar los registros DNS del objetivo
Descripción
Al elegir esta opción, se le mostrarán al usuario los registros DNS obtenidos de la dirección de dominio objetivo.

Nombre del Caso de Uso
Introducir el nombre de dominio objetivo
Descripción
El usuario introducirá el nombre del dominio del objetivo a hallar los registros DNS.

Nombre del Caso de Uso
Hallar los registros DNS inversos del objetivo
Descripción

Al escoger esta opción, el usuario podrá conocer los registros DNS inversos del objetivo.

Nombre del Caso de Uso
Introducir dirección IP o nombre de dominio del objetivo
Descripción
El usuario introducirá la dirección IP o el nombre de dominio del objetivo a hallar los registros DNS inversos.

Nombre del Caso de Uso
Obtener los registros Whois del objetivo
Descripción
Al elegir esta opción, se le mostrará al usuario el contenido de los registros Whois para conocer detalles de la organización, así como del proveedor de servicios Internet (ISP).

Nombre del Caso de Uso
Introducir la dirección IP o nombre de dominio del objetivo
Descripción
El usuario introducirá la dirección IP o nombre de dominio del objetivo a obtener los registros Whois.

Nombre del Caso de Uso
Conocer las propiedades de la subred del objetivo
Descripción
Al elegir esta opción, se le mostrará al usuario las propiedades de la subred de la dirección IP objetivo.

Nombre del Caso de Uso
Introducir dirección IP del objetivo
Descripción

El usuario introducirá la dirección IP del objetivo a conocer las propiedades de la subred.

Nombre del Caso de Uso
Observar la configuración del DNS del objetivo a través de una transferencia de zona
Descripción
Al elegir esta opción, se le mostrará al usuario la configuración que tiene el DNS del objetivo al responder a una transferencia de zona.

Nombre del Caso de Uso
Introducir el nombre de dominio del objetivo
Descripción
El usuario introducirá el nombre de dominio del objetivo a obtener la configuración del DNS al realizar la transferencia de zona.

4.5.2. Casos de Uso del módulo de Fingerprinting

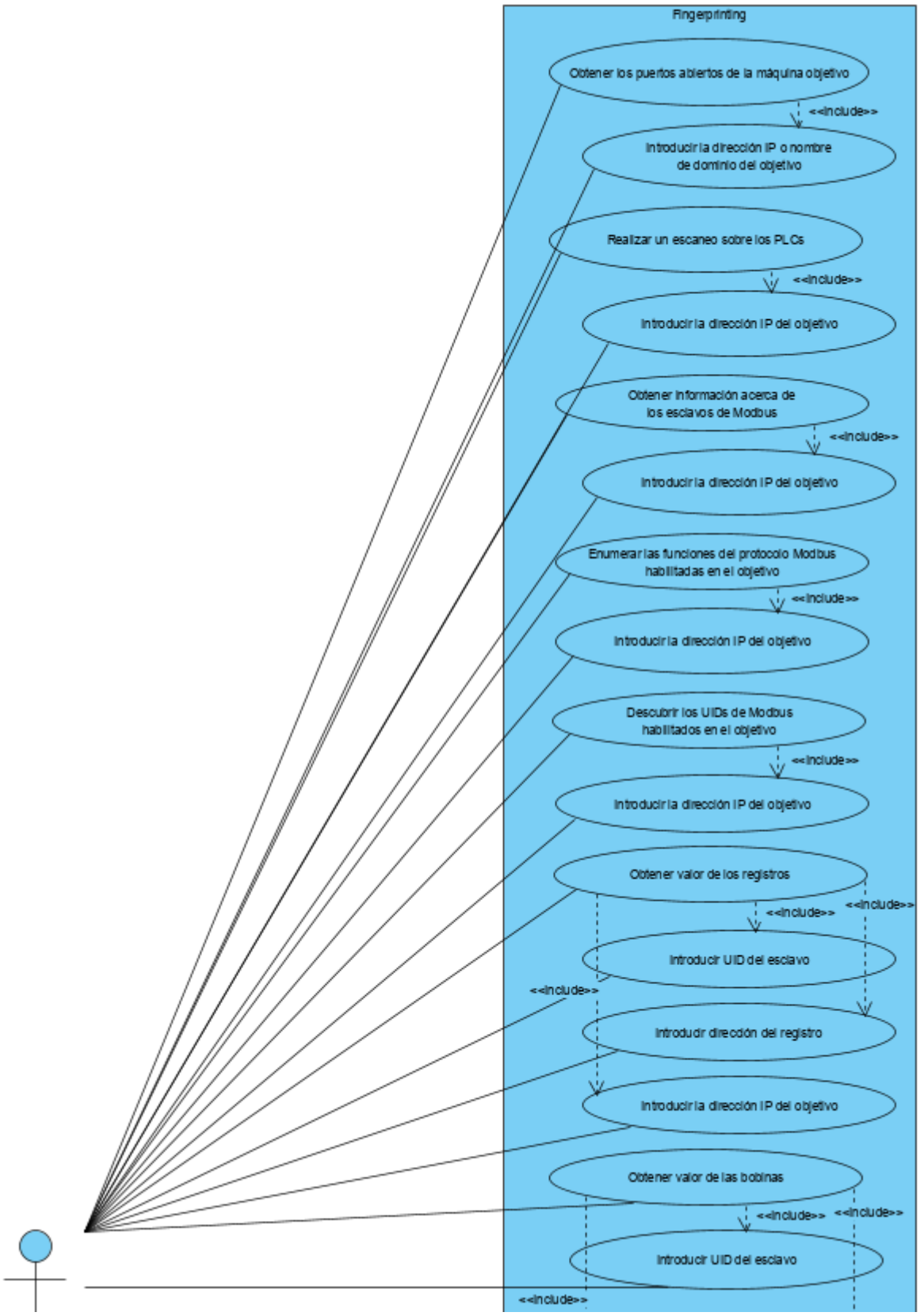

Figura 4.5. Casos de uso del módulo de Fingerprinting (I)

Figura 4.6. Casos de uso del módulo de Fingerprinting (II)

Nombre del Caso de Uso
Obtener los puertos abiertos de la máquina objetivo
Descripción
Al elegir esta opción, el usuario obtendrá los puertos que estén abiertos de la máquina e información de los protocolos que están corriendo en dichos puertos.

Nombre del Caso de Uso
Introducir la dirección IP o nombre de dominio del objetivo
Descripción
El usuario introducirá la dirección IP o el nombre de dominio a conocer los puertos abiertos del objetivo.

Nombre del Caso de Uso
Realizar un escaneo sobre los PLCs
Descripción
Al escoger esta opción, el sistema realizará un escaneo sobre los PLCs activos que estén corriendo en la máquina objetivo.

Nombre del Caso de Uso
Introducir dirección IP del objetivo
Descripción
El usuario introducirá la dirección IP del objetivo a conocer los PLCs e información de ellos.

Nombre del Caso de Uso
Obtener información acerca de los esclavos de Modbus
Descripción
El sistema debe descubrir y recoger toda la información posible acerca de los esclavos del protocolo Modbus que estén corriendo en la dirección del objetivo.

Nombre del Caso de Uso
Introducir dirección IP del objetivo
Descripción
El usuario introducirá la dirección IP del objetivo a obtener información acerca de los esclavos de Modbus.

Nombre del Caso de Uso
Descubrir los UIDs de Modbus habilitados en el objetivo
Descripción
El usuario obtendrá los UIDs de todos los esclavos que estén habilitados en la red Modbus del objetivo.

Nombre del Caso de Uso
Introducir dirección IP del objetivo
Descripción
El usuario introducirá la dirección IP del objetivo a conocer los UIDs habilitados.

Nombre del Caso de Uso
Enumerar las funciones del protocolo Modbus
Descripción
El sistema enumerará las funciones del protocolo Modbus que tenga habilitadas en la máquina objetivo que sea la dirección IP dada por el usuario.

Nombre del Caso de Uso
Introducir dirección IP del objetivo
Descripción
El usuario introducirá la dirección IP del objetivo a conocer las funciones habilitadas.

Nombre del Caso de Uso
Obtener valor de los registros (holding registers)
Descripción

El sistema obtendrá el valor de los registros (holding registers) disponibles en uno de los esclavos que pida el usuario de la dirección IP dada.

Nombre del Caso de Uso
Introducir el UID del esclavo
Descripción
El usuario introducirá el UID del esclavo a leer el registro.

Nombre del Caso de Uso
Introducir dirección del registro
Descripción
El usuario introducirá la dirección del registro a leer.

Nombre del Caso de Uso
Introducir dirección IP del objetivo
Descripción
El usuario introducirá la dirección IP del objetivo a leer los registros.

Nombre del Caso de Uso
Obtener el valor de las bobinas (coils)
Descripción
El sistema obtendrá el valor de las bobinas (coils) disponibles en uno de los esclavos que pida el usuario de la dirección IP dada.

Nombre del Caso de Uso
Introducir el UID del esclavo
Descripción
El usuario introducirá el UID del esclavo a leer el valor de la bobina.

Nombre del Caso de Uso
Introducir dirección de la bobina
Descripción

El usuario introducirá la dirección de la bobina a leer.

Nombre del Caso de Uso
Introducir dirección IP del objetivo
Descripción
El usuario introducirá la dirección IP del objetivo a leer el valor de la bobina.

Nombre del Caso de Uso
Obtener el valor de las entradas discretas (discrete inputs)
Descripción
El sistema obtendrá el estado de las entradas discretas (discrete inputs) del esclavo que pida el usuario en de la dirección IP dada.

Nombre del Caso de Uso
Introducir el UID del esclavo
Descripción
El usuario introducirá el UID del esclavo a leer el valor de las entradas discretas.

Nombre del Caso de Uso
Introducir dirección de la entrada
Descripción
El usuario introducirá la dirección de la entrada a leer.

Nombre del Caso de Uso
Introducir dirección IP del objetivo
Descripción
El usuario introducirá la dirección IP del objetivo a leer la entrada.

Nombre del Caso de Uso
Obtener el contenido de los registros de entrada (input registers)

Descripción
El sistema obtendrá el contenido de los registros de entrada (input registers) del esclavo que pida el usuario en de la dirección IP dada.

Nombre del Caso de Uso
Introducir el UID del esclavo
Descripción
El usuario introducirá el UID del esclavo a leer el contenido de los registros de entrada.

Nombre del Caso de Uso
Introducir dirección de los registros de entrada
Descripción
El usuario introducirá la dirección de los registros de entrada a leer.

Nombre del Caso de Uso
Introducir dirección IP del objetivo
Descripción
El usuario introducirá la dirección IP del objetivo a leer los registros de entrada.

Nombre del Caso de Uso
Obtener respuesta de excepción del esclavo de Modbus
Descripción
El sistema obtendrá el resultado de la respuesta de una excepción (exception response) por parte del esclavo de Modbus que pida el usuario en de la dirección IP dada.

Nombre del Caso de Uso
Introducir el UID del esclavo
Descripción
El usuario introducirá el UID del esclavo a obtener la excepción.

Nombre del Caso de Uso

Introducir dirección IP del objetivo
Descripción
El usuario introducirá la dirección IP del objetivo a obtener la excepción.

Nombre del Caso de Uso
Identificar los PLCs activos en una red Siemens S7
Descripción
El usuario obtendrá cada uno de los PLCs que estén funcionando en la red utilizando el protocolo Siemens S7.

Nombre del Caso de Uso
Introducir dirección IP del objetivo
Descripción
El usuario introducirá la dirección IP del objetivo a identificar los PLCs Siemens S7.

Nombre del Caso de Uso
Obtener información acerca de cada PLC de la red objetivo
Descripción
El usuario obtendrá cada uno de los PLCs que estén funcionando en la red utilizando el protocolo Siemens S7.

Nombre del Caso de Uso
Introducir dirección IP del objetivo
Descripción
El usuario introducirá la dirección IP del objetivo a hallar información sobre los PLCs.

4.5.3. Casos de Uso del módulo Exploitation

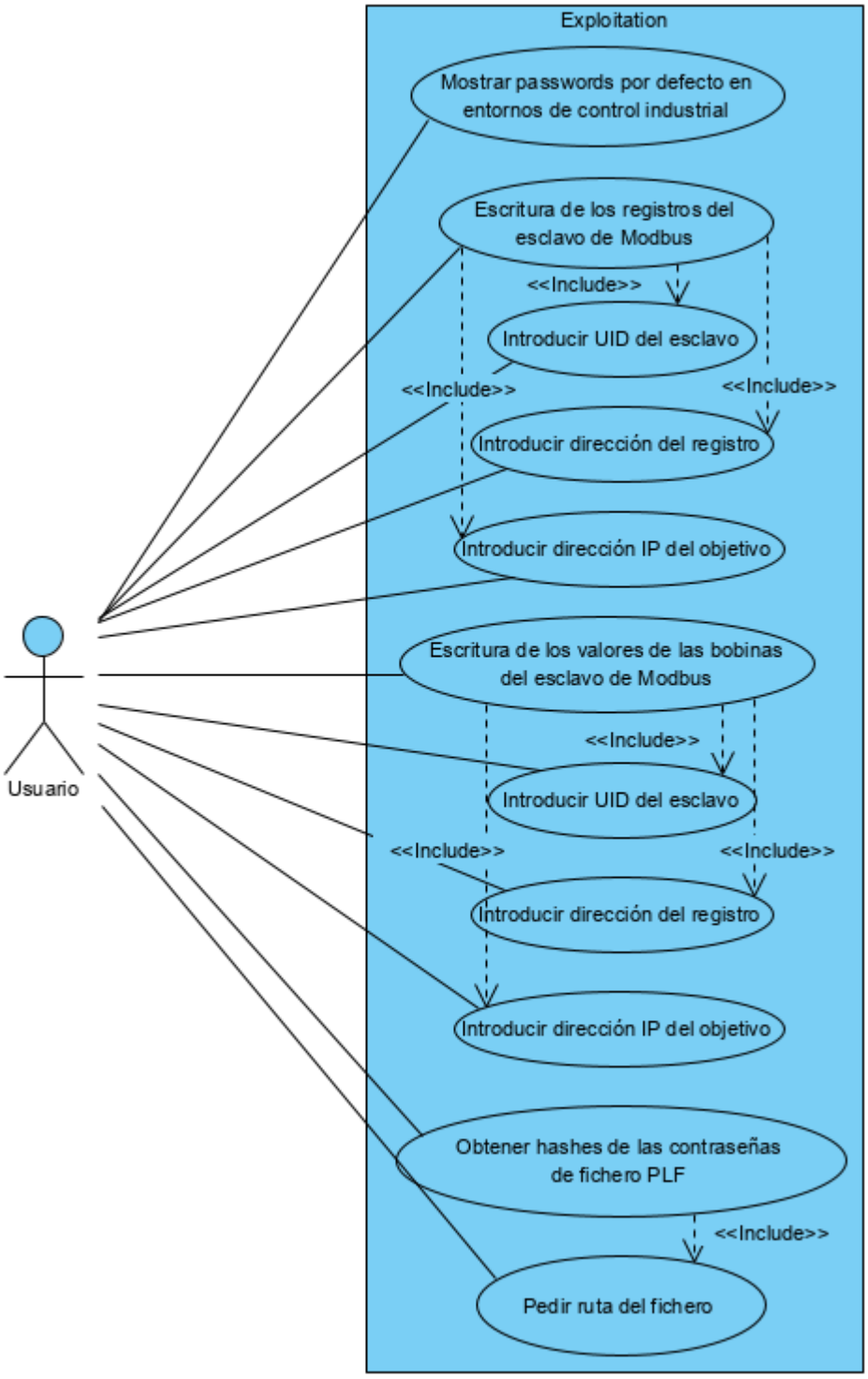

Figura 4.7. Casos de uso del módulo de Exploitation

Nombre del Caso de Uso
Passwords por defecto en entornos de control industrial

Descripción
El sistema mostrará al usuario las passwords por defecto típicas en sistemas de control industrial, así como el dispositivo y su propietario.

Nombre del Caso de Uso
Escritura de registros en esclavo de Modbus
Descripción
El sistema escribirá el registro que quiera el usuario y que tenga habilitado en un esclavo de Modbus.

Nombre del Caso de Uso
Introducir UID del esclavo de Modbus
Descripción
El usuario introducirá el UID del esclavo de Modbus sobre el que quiera escribir un registro.

Nombre del Caso de Uso
Introducir dirección del registro
Descripción
El usuario introducirá la dirección del registro del que quiera conocer el valor.

Nombre del Caso de Uso
Introducir dirección IP del objetivo
Descripción
El usuario introducirá la dirección IP del objetivo a escribir sobre los registros.

Nombre del Caso de Uso
Escritura de los valores de las bobinas (coils)
Descripción
El usuario introducirá el UID del esclavo de Modbus sobre el que quiera escribir un valor de las bobinas (coils).

Nombre del Caso de Uso
Introducir UID del esclavo de Modbus
Descripción
El usuario introducirá el UID del esclavo de Modbus sobre el que quiera escribir un registro.

Nombre del Caso de Uso
Introducir dirección de la bobina (coil)
Descripción
El usuario introducirá la dirección de la bobina (coil) que desee conocer el valor.

Nombre del Caso de Uso
Introducir dirección IP del objetivo
Descripción
El usuario introducirá la dirección IP del objetivo a hallar información sobre los PLCs.

Nombre del Caso de Uso
Obtener hashes contraseñas
Descripción
El sistema deberá obtener las hashes de las contraseñas de los ficheros con extensión .PLF que son utilizados en las redes Siemens S7.

Nombre del Caso de Uso
Pedir ruta del fichero
Descripción
El sistema deberá pedir la ruta del fichero con extensión .PLF para obtener las hashes de las contraseñas de estos.

4.5.4. Casos de Uso del módulo Reportes

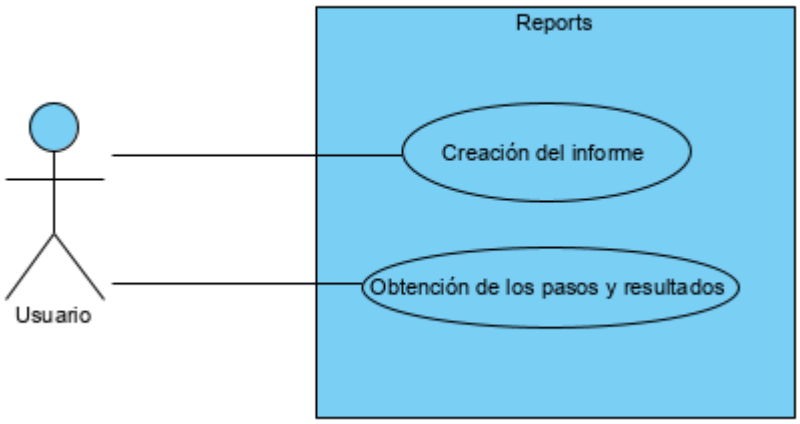

Figura 4.8. Casos de uso del módulo de Reportes

Nombre del Caso de Uso
Creación del informe
Descripción
El sistema deberá elaborar un informe sobre las pruebas llevadas a cabo y los resultados de las mismas.

Nombre del Caso de Uso
Obtención de los pasos y resultados
Descripción
El sistema mostrará al usuario los pasos que este realizó y el resultado de cada uno de ellos en el informe.

4.6. Análisis de Casos de Uso y Escenarios

En esta sección se describirán los escenarios para los casos de uso identificados anteriormente de forma detallada:

4.6.1. Escenarios para los casos de uso del módulo Footprinting

Obtener información pública del objetivo	
Precondiciones	-
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El usuario obtendrá información pública de la empresa expuesta en buscadores como Shodan.

Variaciones (escenarios secundarios)	**Alternativa 1**: El sistema obtiene y muestra toda la información disponible. **Alternativa 2**: El sistema no puede obtener toda la información disponible y alguna es nula. En ese caso, no mostrará nada.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Introducir dirección IP objetivo	
Precondiciones	Conocer la dirección IP del objetivo
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El usuario introducirá una dirección IP según el protocolo IPv4 que será el objetivo a obtener la información pública.
Variaciones (escenarios secundarios)	**Alternativa 1**: El sistema valida la dirección IP según el protocolo IPv4. **Alternativa 2**: La dirección IP no coincide con el formato del protocolo IPv4. **Alternativa 3**: El valor introducido, por ejemplo una cadena de caracteres, es inválido.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Hallar los registros DNS del objetivo	
Precondiciones	-
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El usuario conocerá los registros DNS del objetivo.

Variaciones (escenarios secundarios)	**Alternativa 1**: El sistema obtiene y muestra toda la información disponible que devuelvan los registros DNS. **Alternativa 2**: El sistema no puede obtener toda la información disponible y alguna es nula. En ese caso, no mostrará nada.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Introducir el nombre de dominio objetivo	
Precondiciones	Conocer el nombre de dominio objetivo
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El usuario introducirá el nombre de dominio del objetivo a conocer el valor de los registros DNS.
Variaciones (escenarios secundarios)	**Alternativa 1**: El valor introducido coincide con un nombre de dominio correcto y real. De esta forma, se obtendrá información. **Alternativa 2**: El valor introducido no corresponde con ningún nombre de dominio y no se obtendrá ningún resultado.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Hallar los registros DNS inversos del objetivo	
Precondiciones	-
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El usuario conocerá los registros DNS inversos del objetivo.

Variaciones (escenarios secundarios)	**Alternativa 1**: El sistema obtiene y muestra toda la información disponible que devuelvan los registros DNS inversos. **Alternativa 2**: El sistema no puede obtener toda la información disponible y alguna es nula. En ese caso, no mostrará nada.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Introducir dirección IP o nombre de dominio del objetivo	
Precondiciones	Conocer la dirección IP o el nombre de dominio objetivo
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El usuario introducirá la dirección IP o el nombre de dominio del objetivo a conocer el valor de los registros DNS inversos.
Variaciones (escenarios secundarios)	**Alternativa 1**: El valor introducido coincide con una dirección IP válida según el protocolo IPv4. De esta forma, se obtendrá información. **Alternativa 2**: El valor introducido coincide con un nombre de dominio correcto y real. De esta forma, se obtendrá información. **Alternativa 3**: El valor introducido no corresponde con ningún nombre de dominio y no se obtendrá ningún resultado.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

4.6.2. Escenarios para los casos de uso del módulo Fingerprinting

Obtener los puertos abiertos de la máquina objetivo	
Precondiciones	Conocer la dirección IP del objetivo

Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	Se le mostrará al usuario los puertos que estén abiertos de la máquina e información de los protocolos que están corriendo en dichos puertos.
Variaciones (escenarios secundarios)	**Alternativa 1**: El valor introducido coincide con una dirección IP válida según el protocolo IPv4. De esta forma, se obtendrá información. **Alternativa 2**: El valor introducido coincide con un nombre de dominio correcto y real. De esta forma, se obtendrá información. **Alternativa 3**: El valor introducido no corresponde con ningún nombre de dominio y no se obtendrá ningún resultado.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Realizar un escaneo sobre los PLCs	
Precondiciones	Conocer la dirección IP del objetivo
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El sistema realizará un escaneo sobre los PLCs activos que estén corriendo en la máquina objetivo.
Variaciones (escenarios secundarios)	**Alternativa 1**: Se obtienen los PLCs que están corriendo e información sobre ellos. **Alternativa 2**: No se encuentra ningún dispositivo y no se muestra ningún resultado.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.

Notas	-

Obtener información acerca de los esclavos de Modbus	
Precondiciones	Conocer la dirección IP del objetivo
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El sistema debe descubrir y recoger toda la información posible acerca de los esclavos del protocolo Modbus que estén corriendo en la dirección del objetivo.
Variaciones (escenarios secundarios)	**Alternativa 1**: Se muestra toda la información obtenida acerca de los dispositivos Modbus encontrados. **Alternativa 2**: No se encuentra información debido a que está el puerto cerrado. **Alternativa 3**: No se encuentra información debido a que no está corriendo Modbus en la dirección IP dada.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Descubrir los UIDs de Modbus habilitados en el objetivo	
Precondiciones	Conocer la dirección IP del objetivo
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El usuario obtendrá los UIDs de todos los esclavos que estén habilitados en la red Modbus del objetivo.
Variaciones (escenarios secundarios)	**Alternativa 1**: Se mostrarán todos los UIDs que están corriendo. **Alternativa 2**: No se llegará a mostrar ningún UID porque no se encuentra corriendo ningún dispositivo Modbus.

Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Enumerar las funciones del protocolo Modbus	
Precondiciones	Conocer la dirección IP del objetivo
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El sistema enumerará las funciones del protocolo Modbus que tenga habilitadas en la máquina objetivo que sea la dirección IP dada por el usuario.
Variaciones (escenarios secundarios)	**Alternativa 1**: Se mostrarán todas las funciones habilitadas. **Alternativa 2**: No se llegará a mostrar ninguna función porque no se encuentra corriendo ningún dispositivo Modbus.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Obtener valor de los registros (holding registers)	
Precondiciones	Conocer la dirección IP del objetivo
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El sistema obtendrá el valor de los registros (holding registers) disponibles en uno de los esclavos que pida el usuario de la dirección IP dada.
Variaciones (escenarios secundarios)	**Alternativa 1**: Se mostrará el valor del registro pedido por el usuario. **Alternativa 2**: No se mostrará el valor del registro porque no existe en el UID pedido. **Alternativa 3**:

	No se mostrará el valor del registro porque no está corriendo ningún dispositivo Modbus.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Obtener el valor de las bobinas (coils)	
Precondiciones	Conocer la dirección IP del objetivo
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El sistema obtendrá el valor de las bobinas (coils) disponibles en uno de los esclavos que pida el usuario de la dirección IP dada.
Variaciones (escenarios secundarios)	**Alternativa 1**: Se mostrará el valor de la bobina pedido por el usuario. **Alternativa 2**: No se mostrará el valor de la bobina porque no existe en el UID pedido. **Alternativa 3**: No se mostrará el valor de la bobina porque no está corriendo ningún dispositivo Modbus.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Obtener el valor de las entradas discretas (discrete inputs)	
Precondiciones	Conocer la dirección IP del objetivo
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El sistema obtendrá el estado de las entradas discretas (discrete inputs) del esclavo que pida el usuario en de la dirección IP dada.
Variaciones (escenarios secundarios)	**Alternativa 1**: Se mostrará el valor del estado pedido por el usuario.

	Alternativa 2: No se mostrará el valor del estado porque no existe en el UID pedido. **Alternativa 3**: No se mostrará el valor del estado porque no está corriendo ningún dispositivo Modbus.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Obtener el contenido de los registros de entrada (input registers)	
Precondiciones	Conocer la dirección IP del objetivo
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El sistema obtendrá el contenido de los registros de entrada (input registers) del esclavo que pida el usuario en de la dirección IP dada.
Variaciones (escenarios secundarios)	**Alternativa 1**: Se mostrará el valor del registro pedido por el usuario. **Alternativa 2**: No se mostrará el valor del registro porque no existe en el UID pedido. **Alternativa 3**: No se mostrará el valor del registro porque no está corriendo ningún dispositivo Modbus.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Obtener respuesta de excepción del esclavo de Modbus	
Precondiciones	Conocer la dirección IP del objetivo
Postcondiciones	-
Actores	Iniciado por el usuario

Descripción	El sistema obtendrá el resultado de la respuesta de una excepción (exception response) por parte del esclavo de Modbus que pida el usuario en de la dirección IP dada.
Variaciones (escenarios secundarios)	**Alternativa 1**: Se mostrará el valor de la respuesta de excepción. **Alternativa 2**: No se mostrará el valor de la respuesta de excepción porque no existe en el UID pedido. **Alternativa 3**: No se mostrará el valor de la respuesta de excepción porque no está corriendo ningún dispositivo Modbus.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Introducir dirección IP objetivo	
Precondiciones	Conocer la dirección IP del objetivo
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El usuario introducirá una dirección IP según el protocolo IPv4 que será el objetivo a obtener información sobre los esclavos de Modbus.
Variaciones (escenarios secundarios)	**Alternativa 1**: El sistema valida la dirección IP según el protocolo IPv4. **Alternativa 2**: La dirección IP no coincide con el formato del protocolo IPv4. **Alternativa 3**: El valor introducido, por ejemplo una cadena de caracteres, es inválido.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Introducir UID del esclavo de Modbus	
Precondiciones	Conocer el UID
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El usuario introducirá un UID que se corresponderá con un esclavo de Modbus.
Variaciones (escenarios secundarios)	**Alternativa 1**: El valor introducido no es un número entero positivo, por lo que se produce un error y se le muestra al usuario. **Alternativa 2**: El valor introducido no está comprendido entre 0 y 255 (ambos incluidos), por lo que se produce un error y se muestra el usuario. **Alternativa 3**: El valor introducido es correcto y continúa la ejecución del programa.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Introducir dirección de memoria del registro de Modbus	
Precondiciones	Conocer la dirección de memoria
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El usuario introducirá una dirección de memoria del registro a conocer.
Variaciones (escenarios secundarios)	**Alternativa 1**: El valor introducido no es un número entero positivo, por lo que se produce un error y se le muestra al usuario. **Alternativa 2**: El valor introducido no está comprendido entre 0 y 65635 (ambos incluidos), por lo que se produce un error y se muestra el usuario.

	Alternativa 3: El valor introducido es correcto y continúa la ejecución del programa.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Introducir el valor de los estados de entradas para Modbus	
Precondiciones	Conocer el valor del estado
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El usuario introducirá un valor para el estado para bobinas o entradas discretas, depende del caso que se esté ejecutando (la validación es la misma).
Variaciones (escenarios secundarios)	**Alternativa 1**: El valor introducido es distinto a los dos posibles valores (ON/OFF). **Alternativa 2**: El valor introducido se corresponde con uno de los dos posibles valores (ON/OFF).
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

4.6.3. Escenarios para los casos de uso del módulo Exploitation

Passwords por defecto en entornos de control industrial	
Precondiciones	-
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El sistema mostrará al usuario las passwords por defecto típicas en sistemas de control industrial, así como el dispositivo y su propietario.

Variaciones (escenarios secundarios)	**Alternativa 1**: Se muestran las contraseñas por defecto al usuario. **Alternativa 2**: Se produce un error al *parsear* el fichero con las contraseñas por defecto y no se muestran al usuario.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Escritura de registros en esclavo de Modbus	
Precondiciones	Conocer el UID del esclavo de Modbus Conocer la dirección del registro Conocer la dirección IP de la máquina que esté corriendo Modbus
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El sistema escribirá el registro que quiera el usuario y que tenga habilitado en un esclavo de Modbus.
Variaciones (escenarios secundarios)	**Alternativa 1**: El UID introducido no se corresponde con UID válido. Se muestra un error y no se lleva a cabo. **Alternativa 2**: La dirección del registro introducida no se corresponde con ninguna dirección válida. **Alternativa 3**: La dirección IP no tiene el formato adecuado y no se lleva a cabo. **Alternativa 4**: Todos los valores introducidos son validados, se lleva a cabo la prueba y se escribe sobre el registro dado.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Introducir UID del esclavo de Modbus	
Precondiciones	Conocer el UID
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El usuario introducirá el UID del esclavo
Variaciones (escenarios secundarios)	**Alternativa 1**: El valor introducido coincide con un UID válido. De esta forma, se llevará a cabo la prueba. **Alternativa 2**: El valor introducido no corresponde con ningún UID y no se obtendrá ningún resultado. Se mostrará un error al usuario.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Introducir la dirección del registro	
Precondiciones	Conocer la dirección del registro
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El usuario introducirá la dirección del registro a obtener.
Variaciones (escenarios secundarios)	**Alternativa 1**: El valor introducido coincide con una dirección de registro válida. De esta forma, se lleva a cabo la prueba. **Alternativa 2**: El valor introducido no se corresponde con ninguna dirección de registro y no se obtendrá ningún resultado. Se mostrará un error al usuario.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Introducir dirección IP objetivo	
Precondiciones	Conocer la dirección IP del objetivo

Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El usuario introducirá una dirección IP según el protocolo IPv4 que será el objetivo a escribir sobre los registros de Modbus
Variaciones (escenarios secundarios)	**Alternativa 1**: El sistema valida la dirección IP según el protocolo IPv4. **Alternativa 2**: La dirección IP no coincide con el formato del protocolo IPv4. **Alternativa 3**: El valor introducido, por ejemplo, una cadena de caracteres, es inválido.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Escritura de los valores de las bobinas (coils) de Modbus	
Precondiciones	Conocer el UID del esclavo de Modbus Conocer la dirección del registro Conocer la dirección IP de la máquina que esté corriendo Modbus
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El sistema escribirá sobre los valores de las bobinas que quiera el usuario y que tenga habilitado en un esclavo de Modbus.
Variaciones (escenarios secundarios)	**Alternativa 1**: El UID introducido no se corresponde con UID válido. Se muestra un error y no se lleva a cabo. **Alternativa 2**: La dirección de la bobina introducida no se corresponde con ninguna dirección válida. **Alternativa 3**: La dirección IP no tiene el formato adecuado y no se lleva a cabo.

	Alternativa 4: Todos los valores introducidos son validados, se lleva a cabo la prueba y se escribe sobre el registro dado de la bobina.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Introducir UID del esclavo de Modbus	
Precondiciones	Conocer el UID
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El usuario introducirá el UID del esclavo
Variaciones (escenarios secundarios)	**Alternativa 1**: El valor introducido coincide con un UID válido. De esta forma, se llevará a cabo la prueba. **Alternativa 2**: El valor introducido no corresponde con ningún UID y no se obtendrá ningún resultado. Se mostrará un error al usuario.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Introducir la dirección de la bobina (coil)	
Precondiciones	Conocer la dirección de la bobina
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El usuario introducirá la dirección de la bobina a obtener.
Variaciones (escenarios secundarios)	**Alternativa 1**: El valor introducido coincide con una dirección válida. De esta forma, se lleva a cabo la prueba. **Alternativa 2**:

	El valor introducido no se corresponde con ninguna dirección y no se obtendrá ningún resultado. Se mostrará un error al usuario.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Introducir dirección IP objetivo	
Precondiciones	Conocer la dirección IP del objetivo
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El usuario introducirá una dirección IP según el protocolo IPv4 que será el objetivo a escribir sobre las bobinas de Modbus
Variaciones (escenarios secundarios)	**Alternativa 1**: El sistema valida la dirección IP según el protocolo IPv4. **Alternativa 2**: La dirección IP no coincide con el formato del protocolo IPv4. **Alternativa 3**: El valor introducido, por ejemplo, una cadena de caracteres, es inválido.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Obtener hashes de las contraseñas	
Precondiciones	Tener un fichero con extensión .PLF de la red de Siemens S7
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El sistema deberá obtener las hashes de las contraseñas del fichero con extensión .PLF, que son utilizados en las redes Siemens S7, pasado por el usuario.
Variaciones (escenarios secundarios)	**Alternativa 1**: El fichero tiene la extensión correcta y se obtienen los hashes.

	Alternativa 2: El fichero no tiene el formato correcto, no se ejecuta la prueba y se muestra un error al usuario.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

Pedir ruta de fichero PLF	
Precondiciones	Conocer la ruta del fichero
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El usuario introducirá la ruta al fichero.
Variaciones (escenarios secundarios)	**Alternativa 1**: La ruta es correcta y se encuentra al fichero. De esta forma, se obtendrá información. **Alternativa 2**: La ruta introducida no se corresponde con ningún fichero y no se obtendrá ningún resultado.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

4.6.4. Escenarios para los casos de uso del módulo Reportes

Creación del informe	
Precondiciones	-
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El sistema deberá elaborar un informe sobre las pruebas llevadas a cabo y los resultados de las mismas.
Variaciones (escenarios secundarios)	**Alternativa 1**: Se genera el reporte con los resultados y se muestra al usuario.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.

Notas	-

Obtención de los pasos	
Precondiciones	-
Postcondiciones	-
Actores	Iniciado por el usuario
Descripción	El sistema mostrará al usuario los pasos que este realizó y el resultado de cada uno de ellos en el informe.
Variaciones (escenarios secundarios)	**Alternativa 1**: Se muestran todos los pasos que fue realizando el usuario y el resultado de los mismos.
Excepciones	Se produce una interrupción de teclado (Ctrl + C) que es capturada y finaliza la ejecución del programa.
Notas	-

4.7. Análisis de la Interfaz de Usuario

La aplicación dispondrá de una interfaz sencilla por consola de comandos como es habitual en las herramientas de seguridad actuales. La idea es crear un menú con diferentes opciones para los múltiples casos que pueden darse dentro de un pentesting o auditoría de seguridad y separarlo en las distintas partes de estos. Se puede tomar como referencia la interfaz de la herramienta The Social Engineer Toolkit (SET) de David Kennedy para realizar ingeniería social.

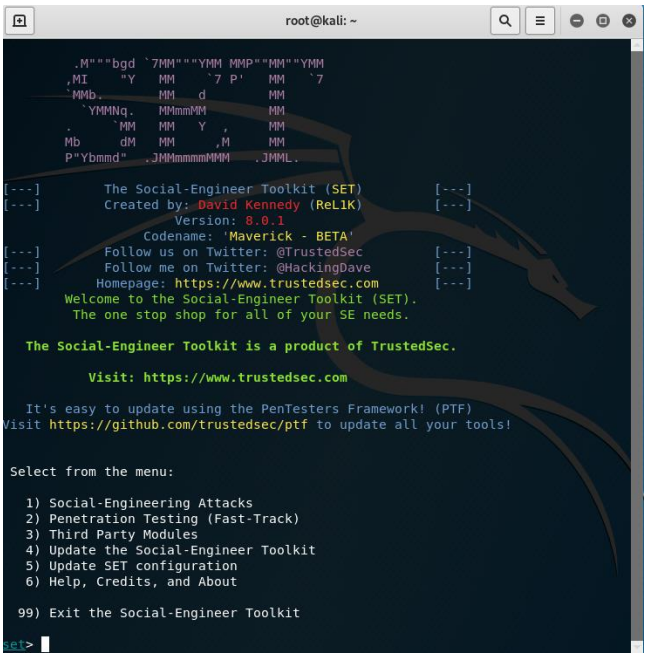

Figura 4.9. Interfaz Social Engineer Toolkit (I)

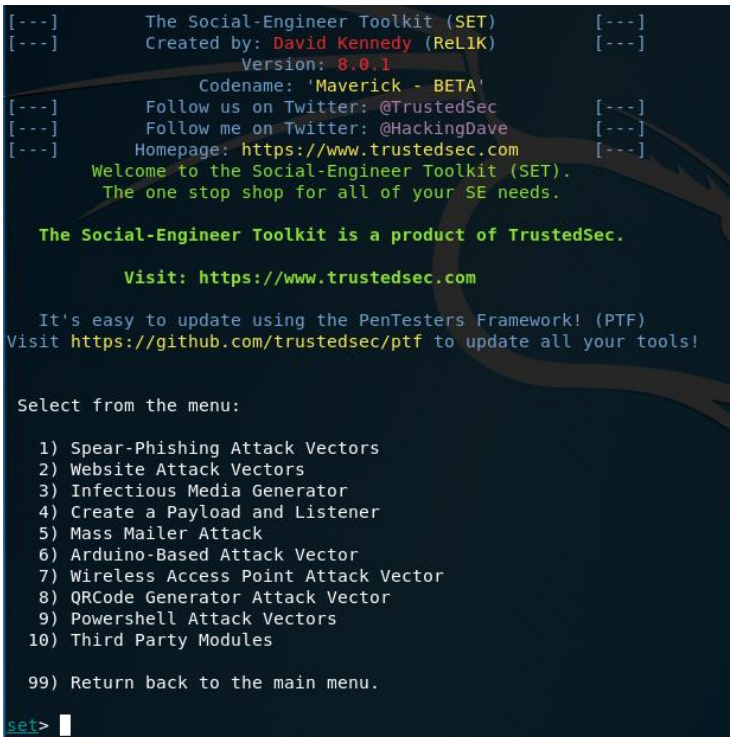

Figura 4.10. Interfaz Social Engineer Toolkit (II)

Entonces, la interfaz, con las partes del pentesting, quedaría de la siguiente forma:

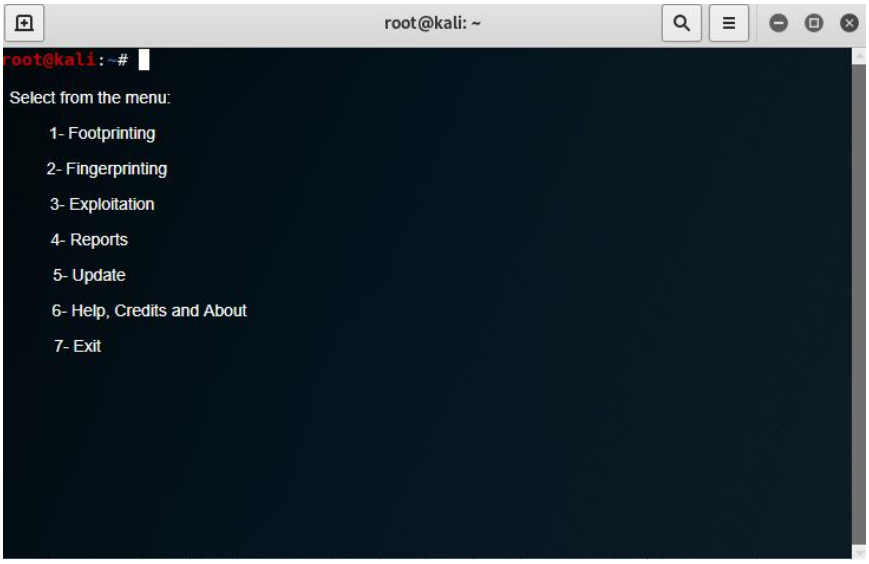

Figura 4.11. Diseño inicial interfaz de la aplicación

De esta forma, cada parte estará diferenciada y se podrá ir realizando todo en orden.

4.8. Especificación del Plan de Pruebas

En esta sección crearemos y diseñaremos el plan de pruebas de la aplicación donde se detallarán todas las pruebas a realizar que se corresponden con los casos de uso analizados previamente. En

el apartado del desarrollo de las pruebas, será donde se presenten los resultados del siguiente diseño de conjunto de pruebas:

Caso de Uso: Obtener información pública del objetivo	
Prueba	**Resultado Esperado**
Introducir una dirección IPv4 válida	Se ejecuta y se obtiene la información
Prueba	**Resultado Esperado**
Introducir una dirección IP inválida	No se llega a ejecutar (se valida antes) y se muestra un error
Prueba	**Resultado Esperado**
Introducir caracteres aleatorios que no se correspondan con una dirección	No se llega a ejecutar (se valida antes) y se muestra un error

Caso de Uso: Hallar los registros DNS del objetivo	
Prueba	**Resultado Esperado**
Introducir un nombre de dominio correcto	Se ejecuta y se obtienen los registros DNS
Prueba	**Resultado Esperado**
Introducir un nombre de dominio incorrecto	No se llega a ejecutar (se valida antes) y se muestra un error
Prueba	**Resultado Esperado**
Introducir un nombre de dominio correcto pero que no existe	Se ejecuta, pero no se obtienen resultados y se muestra un error
Prueba	**Resultado Esperado**
Introducir caracteres que no se correspondan con un nombre de dominio	No se ejecuta el análisis y se muestra un error

Caso de Uso: Hallar los registros DNS inversos del objetivo	
Prueba	**Resultado Esperado**

Introducir una dirección IPv4 válida	Se ejecuta el análisis
Prueba	**Resultado Esperado**
Introducir una dirección IPv4 inválida	No se ejecuta el análisis y se muestra un error
Prueba	**Resultado Esperado**
Introducir un nombre de dominio válido	Se ejecuta el análisis
Prueba	**Resultado Esperado**
Introducir un nombre de dominio inválido	No se ejecuta el análisis y se muestra un error
Prueba	**Resultado Esperado**
Introducir caracteres que no se correspondan ni con una dirección IP ni con un nombre de dominio	No se ejecuta el análisis y se muestra un error

Caso de Uso: Obtener los registros Whois del objetivo	
Prueba	**Resultado Esperado**
Introducir una dirección IPv4 válida	Se ejecuta y se obtienen los registros
Prueba	**Resultado Esperado**
Introducir una dirección IPv4 inválida	No se llega a ejecutar (se valida antes) y se muestra un error
Prueba	**Resultado Esperado**
Introducir un nombre de dominio válido	Se ejecuta y se obtienen los registros
Prueba	**Resultado Esperado**
Introducir un nombre de dominio inválido	No se llega a ejecutar (se valida antes) y se muestra un error
Prueba	**Resultado Esperado**
Introducir caracteres que no se correspondan ni con una dirección IPv4 ni con un nombre de dominio	No se ejecuta y se muestra un error

Caso de Uso: Conocer las propiedades de la subred del objetivo	
Prueba	**Resultado Esperado**
Introducir una dirección IPv4 válida	Se ejecuta el análisis
Prueba	**Resultado Esperado**
Introducir una dirección IPv4 inválida	No se ejecuta el análisis y se muestra un error
Prueba	**Resultado Esperado**
Introducir caracteres que no se correspondan con una dirección IPv4	No se ejecuta el análisis y se muestra un error

Caso de Uso: Observar la configuración del DNS del objetivo a través de una transferencia de zona	
Prueba	**Resultado Esperado**
Introducir un nombre de dominio válido	Se ejecuta y se obtiene información acerca de la transferencia de zona
Prueba	**Resultado Esperado**
Introducir un nombre de dominio inválido	No se llega a ejecutar (se valida antes) y se muestra un error
Prueba	**Resultado Esperado**
Introducir un nombre de dominio válido pero que no exista	Se intenta realizar la transferencia de zona, pero falla y se muestra un error
Prueba	**Resultado Esperado**
Introducir caracteres que no se correspondan con un nombre de dominio	No se ejecuta el análisis y se muestra un error

Caso de Uso: Obtener los puertos abiertos de la máquina objetivo	
Prueba	**Resultado Esperado**
Introducir una IP válida	Se ejecuta el análisis
Prueba	**Resultado Esperado**
Introducir una IP que no exista	No se ejecuta el análisis y se muestra un error

Prueba	**Resultado Esperado**
Introducir caracteres que no se correspondan con una IP	No se ejecuta el análisis y se muestra un error

Caso de Uso: Realizar un escaneo sobre los PLCs	
Prueba	**Resultado Esperado**
Introducir una IP válida	Se ejecuta el análisis
Prueba	**Resultado Esperado**
Introducir una IP que no exista	No se ejecuta el análisis y se muestra un error
Prueba	**Resultado Esperado**
Introducir caracteres que no se correspondan con una IP	No se ejecuta el análisis y se muestra un error

Caso de Uso: Obtener información acerca de los esclavos de Modbus	
Prueba	**Resultado Esperado**
Introducir una IP válida	Se ejecuta el análisis
Prueba	**Resultado Esperado**
Introducir una IP que no exista	No se ejecuta el análisis y se muestra un error
Prueba	**Resultado Esperado**
Introducir caracteres que no se correspondan con una IP	No se ejecuta el análisis y se muestra un error

Caso de Uso: Descubrir los UIDs de Modbus habilitados en el objetivo	
Prueba	**Resultado Esperado**
Introducir una IP válida	Se ejecuta el análisis
Prueba	**Resultado Esperado**
Introducir una IP que no exista	No se ejecuta el análisis y se muestra un error
Prueba	**Resultado Esperado**

Introducir caracteres que no se correspondan con una IP	No se ejecuta el análisis y se muestra un error

Caso de Uso: Enumerar las funciones del protocolo Modbus	
Prueba	**Resultado Esperado**
Introducir una IP válida	Se ejecuta el análisis
Prueba	**Resultado Esperado**
Introducir una IP que no exista	No se ejecuta el análisis y se muestra un error
Prueba	**Resultado Esperado**
Introducir caracteres que no se correspondan con una IP	No se ejecuta el análisis y se muestra un error

Caso de Uso: Obtener valor de los registros (holding registers)	
Prueba	**Resultado Esperado**
Introducir un UID válido	Se ejecuta el análisis (si el resto de entradas son válidas)
Prueba	**Resultado Esperado**
Introducir un UID no válido	No se ejecuta el análisis y da un error
Prueba	**Resultado Esperado**
Introducir dirección de registro válida	Se ejecuta el análisis (si el resto de entradas son válidas)
Prueba	**Resultado Esperado**
Introducir dirección de registro inválida	No se ejecuta el análisis y da un error
Prueba	**Resultado Esperado**
Introducir una dirección IP válida	Se ejecuta el análisis (si el resto de entradas son válidas)
Prueba	**Resultado Esperado**
Introducir una dirección IP no válida	No se ejecuta el análisis y da un error
Prueba	**Resultado Esperado**
UID, dirección del registro y dirección IP son válidos	Se ejecuta el análisis

Caso de Uso: Obtener el valor de las bobinas (coils)	
Prueba	**Resultado Esperado**
Introducir un UID válido	Se ejecuta el análisis (si el resto de entradas son válidas)
Prueba	**Resultado Esperado**
Introducir un UID no válido	No se ejecuta el análisis y da un error
Prueba	**Resultado Esperado**
Introducir dirección de registro válida	Se ejecuta el análisis (si el resto de entradas son válidas)
Prueba	**Resultado Esperado**
Introducir dirección de registro inválida	No se ejecuta el análisis y da un error
Prueba	**Resultado Esperado**
Introducir una dirección IP válida	Se ejecuta el análisis (si el resto de entradas son válidas)
Prueba	**Resultado Esperado**
Introducir una dirección IP no válida	No se ejecuta el análisis y da un error
Prueba	**Resultado Esperado**
UID, dirección del registro y dirección IP son válidos	Se ejecuta el análisis

Caso de Uso: Obtener el valor de las entradas discretas (discrete inputs)	
Prueba	**Resultado Esperado**
Introducir un UID válido	Se ejecuta el análisis (si el resto de entradas son válidas)
Prueba	**Resultado Esperado**
Introducir un UID no válido	No se ejecuta el análisis y da un error
Prueba	**Resultado Esperado**
Introducir dirección de registro válida	Se ejecuta el análisis (si el resto de entradas son válidas)
Prueba	**Resultado Esperado**
Introducir dirección de registro inválida	No se ejecuta el análisis y da un error
Prueba	**Resultado Esperado**

Introducir una dirección IP válida	Se ejecuta el análisis (si el resto de entradas son válidas)
Prueba	**Resultado Esperado**
Introducir una dirección IP no válida	No se ejecuta el análisis y da un error
Prueba	**Resultado Esperado**
UID, dirección del registro y dirección IP son válidos	Se ejecuta el análisis

Caso de Uso: Obtener el contenido de los registros de entrada (input registers)	
Prueba	**Resultado Esperado**
Introducir un UID válido	Se ejecuta el análisis (si el resto de entradas son válidas)
Prueba	**Resultado Esperado**
Introducir un UID no válido	No se ejecuta el análisis y da un error
Prueba	**Resultado Esperado**
Introducir dirección de registro válida	Se ejecuta el análisis (si el resto de entradas son válidas)
Prueba	**Resultado Esperado**
Introducir dirección de registro inválida	No se ejecuta el análisis y da un error
Prueba	**Resultado Esperado**
Introducir una dirección IP válida	Se ejecuta el análisis (si el resto de entradas son válidas)
Prueba	**Resultado Esperado**
Introducir una dirección IP no válida	No se ejecuta el análisis y da un error
Prueba	**Resultado Esperado**
UID, dirección del registro y dirección IP son válidos	Se ejecuta el análisis

Caso de Uso: Obtener respuesta de excepción del esclavo de Modbus	
Prueba	**Resultado Esperado**
Introducir un UID válido	Se ejecuta el análisis (si el resto de entradas son válidas)
Prueba	**Resultado Esperado**
Introducir un UID no válido	No se ejecuta el análisis y da un error

Prueba	**Resultado Esperado**
Introducir dirección de registro válida	Se ejecuta el análisis (si el resto de entradas son válidas)
Prueba	**Resultado Esperado**
Introducir dirección de registro inválida	No se ejecuta el análisis y da un error
Prueba	**Resultado Esperado**
Introducir una dirección IP válida	Se ejecuta el análisis (si el resto de entradas son válidas)
Prueba	**Resultado Esperado**
Introducir una dirección IP no válida	No se ejecuta el análisis y da un error
Prueba	**Resultado Esperado**
UID, dirección del registro y dirección IP son válidos	Se ejecuta el análisis

Caso de Uso: Passwords por defecto en entornos de control industrial	
Prueba	**Resultado Esperado**
Obtener las passwords por defecto	Se ejecuta el análisis y se muestran

Caso de Uso: Escritura de registros en esclavo de Modbus	
Prueba	**Resultado Esperado**
Introducir un UID válido	Se ejecuta el análisis (si el resto de entradas son válidas)
Prueba	**Resultado Esperado**
Introducir un UID no válido	No se ejecuta el análisis y da un error
Prueba	**Resultado Esperado**
Introducir dirección de registro válida	Se ejecuta el análisis (si el resto de entradas son válidas)
Prueba	**Resultado Esperado**
Introducir dirección de registro inválida	No se ejecuta el análisis y da un error
Prueba	**Resultado Esperado**
Introducir una dirección IP válida	Se ejecuta el análisis (si el resto de entradas son válidas)

Prueba	**Resultado Esperado**
Introducir una dirección IP no válida	No se ejecuta el análisis y da un error
Prueba	**Resultado Esperado**
UID, dirección del registro y dirección IP son válidos	Se ejecuta el análisis

Caso de Uso: Escritura de los valores de las bobinas (coils)	
Prueba	**Resultado Esperado**
Introducir un UID válido	Se ejecuta el análisis (si el resto de entradas son válidas)
Prueba	**Resultado Esperado**
Introducir un UID no válido	No se ejecuta el análisis y da un error
Prueba	**Resultado Esperado**
Introducir dirección de la bobina válida	Se ejecuta el análisis (si el resto de entradas son válidas)
Prueba	**Resultado Esperado**
Introducir dirección de la bobina inválida	No se ejecuta el análisis y da un error
Prueba	**Resultado Esperado**
Introducir una dirección IP válida	Se ejecuta el análisis (si el resto de entradas son válidas)
Prueba	**Resultado Esperado**
Introducir una dirección IP no válida	No se ejecuta el análisis y da un error
Prueba	**Resultado Esperado**
UID, dirección del registro y dirección IP son válidos	Se ejecuta el análisis

Caso de Uso: Obtener hashes de las contraseñas	
Prueba	**Resultado Esperado**
Introducir una ruta al fichero .PLF válida	Se ejecuta el análisis
Prueba	**Resultado Esperado**
Introducir una ruta al fichero .PLF no válida	No se ejecuta el análisis y da un error

Capítulo 5. Diseño del Sistema

5.1. Diagrama de Paquetes

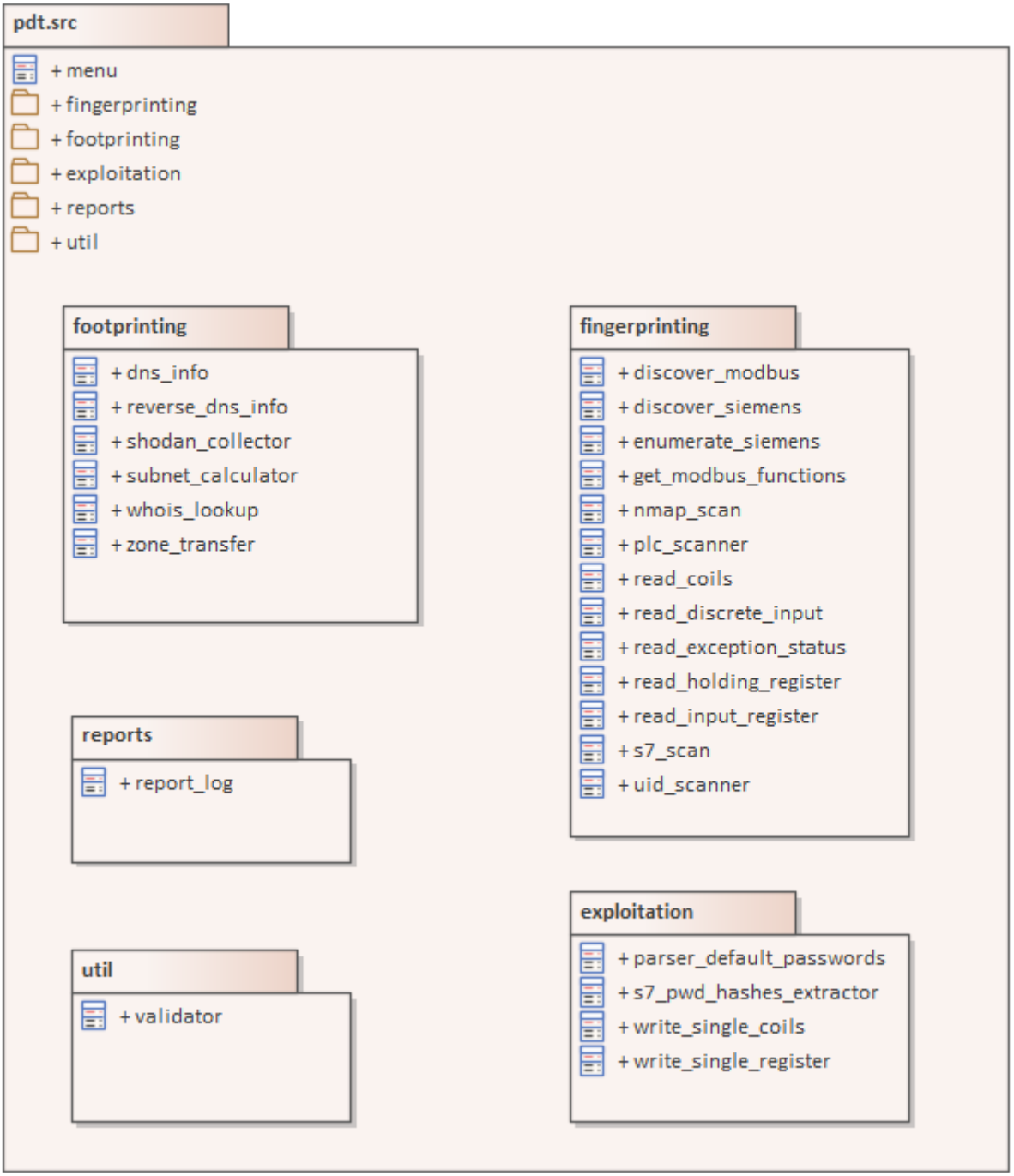

Figura 5.1. Diagrama de Paquetes

5.1.1. Paquete footprinting

En este paquete se encontrarán las pruebas que se realizan en la primera etapa de una auditoría para el reconocimiento del objetivo y recolección de información.

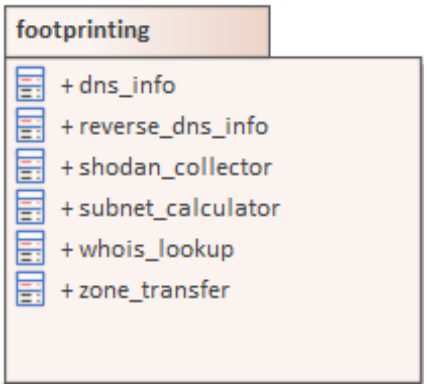

Figura 5.2. Paquete footprinting

5.1.2. Paquete fingerprinting

Aquí se hallan las clases que interactúan directamente contra el sistema objetivo, obteniendo una información determinada en cada una de las pruebas. Existen tanto genéricas, por ejemplo, un escaneo de puertos de la máquina objetivo, como específicas para un protocolo en concreto, como puede ser la lectura de registros de Modbus.

A lo largo del desarrollo del proyecto, se han tratado los protocolos Modbus y Siemens S7 y por ello existen pruebas para estos concretamente, pero se podrían añadir para cualquier otro en este paquete.

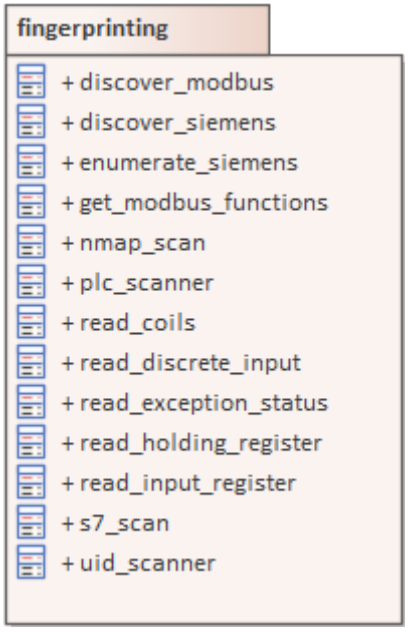

Figura 5.3. Paquete fingerprinting

5.1.3. Paquete exploitation

En dicho paquete se pueden encontrar las pruebas relacionadas con la explotación que dependiendo del tipo de auditoría o análisis de vulnerabilidades, en algunos casos no se llega a

realizar y solamente se da un informe con la información obtenida y si existe algún tipo de vulnerabilidad. Aquí se incluye una lista de contraseñas por defecto en los sistemas de control industrial, así como pruebas específicas para protocolos en concretos, al igual que en el paquete de fingerprinting.

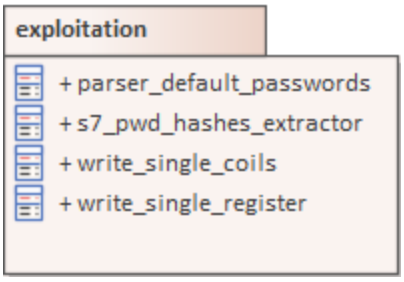

Figura 5.4. Paquete exploitation

5.1.4. Paquete reports

Paquete en el que están las clases para la realización de un informe. Actualmente, lo muestra por pantalla al usuario cuando este lo pida y lo guarda en un fichero txt para su consulta posterior.

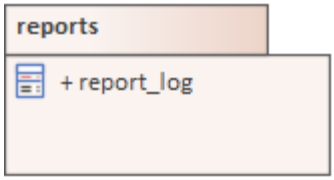

Figura 5.5. Paquete reports

5.1.5. Paquete útil

En este se encontrarán las clases de utilidad como método auxiliar al resto de paquetes. En este caso, se dispone de una clase que realiza la validación de todas las entradas del usuario y se localiza toda en el mismo sitio para una mayor claridad.

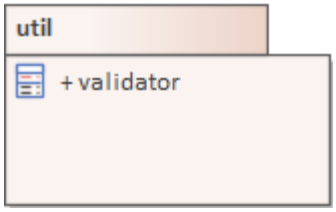

Figura 5.6. Paquete util

5.2. Diseño de Clases

5.2.1. Paquete footprinting

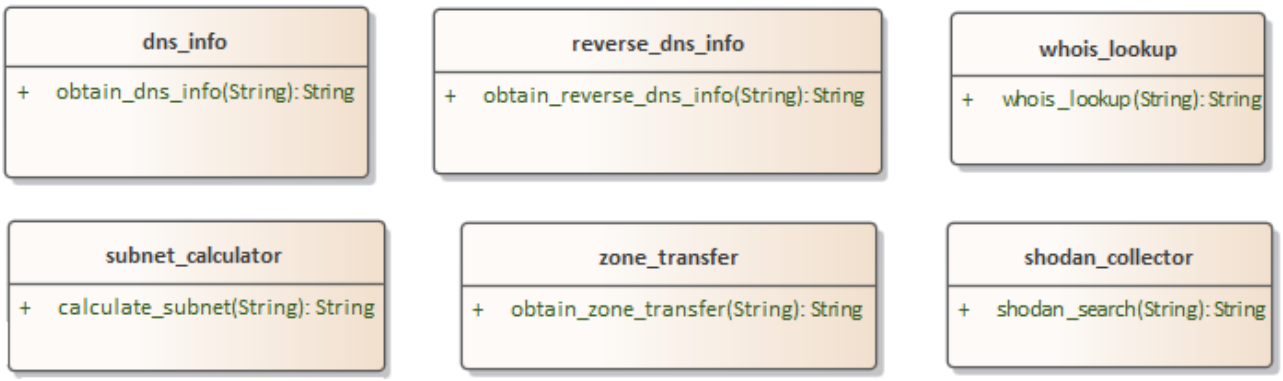

Figura 5.7. Diseño de clases (footprinting)

5.2.2. Paquete fingerprinting

Figura 5.8. Diseño de clases (fingerprinting)

5.2.3. Paquete exploitation

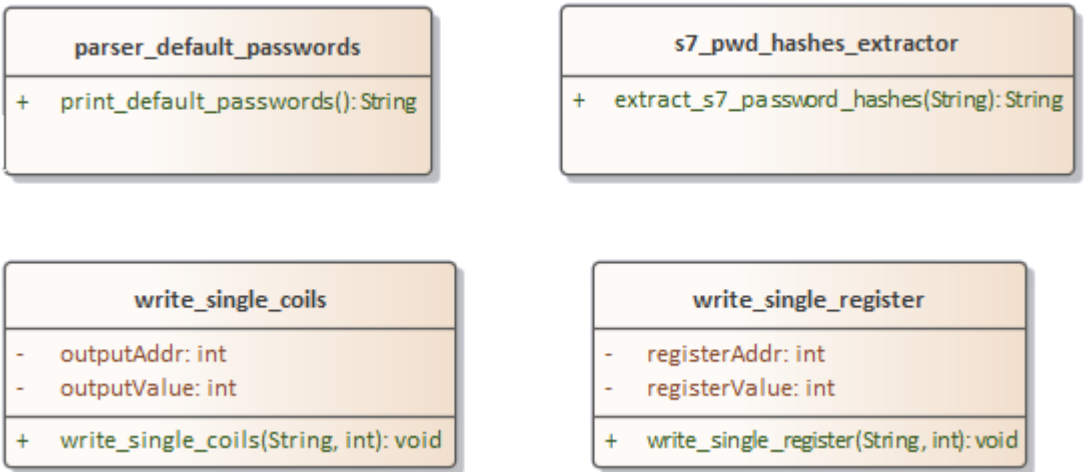

Figura 5.9. Diseño de clases (exploitation)

5.2.4. Paquete reports

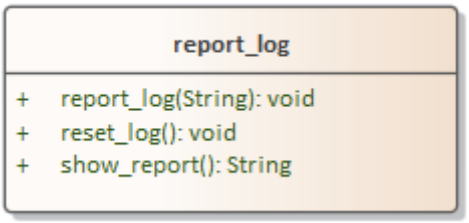

Figura 5.10. Diseño de clases (reports)

5.2.5. Paquete util

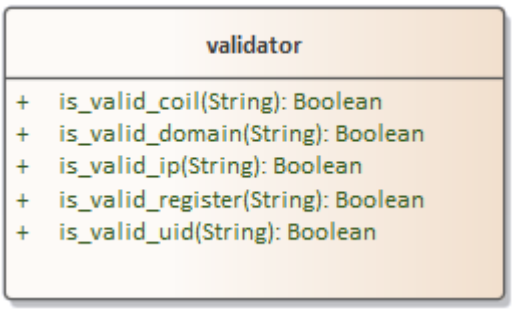

Figura 5.11. Diseño de clases (util)

5.3. Diseño de la Interfaz

Figura 5.12. Diseño de la interfaz (I)

Figura 5.12. Diseño de la interfaz (II)

Figura 5.12. Diseño de la interfaz (III)

Figura 5.12. Diseño de la interfaz (IV)

Capítulo 6. Implementación del Sistema

6.1. Lenguajes de Programación

6.1.1. Python

Python es el lenguaje de programación elegido para el desarrollo del proyecto debido a que es el más usado en la realización de tests de penetración. Aunque este lenguaje es multiplataforma, lo más recomendable sería utilizar un entorno de auditoría como puede ser Kali Linux donde vienen preinstaladas la mayoría de las herramientas y librerías utilizadas, o cualquier derivado de Linux (Ubuntu o Debian) puesto que la herramienta Scapy no viene preinstalada en entornos Windows. De esta forma no hace falta realizar ninguna instalación adicional por parte del usuario.

6.2. Herramientas y Programas Usados para el Desarrollo

6.2.1. Visual Studio Code

Se ha usado dicho editor para realizar el desarrollo del proyecto con el plugin de Kite para el autocompletado de código de Python.

6.2.2. Github

La plataforma Github ha sido la elegida como repositorio de código para gestionar las diferentes versiones de la herramienta y poder tener un control de los cambios realizados en el proyecto.

6.2.3. Travis-CI

Es el sistema de integración continua elegida para el proyecto que además tiene parte gratuita para proyectos Open Source. Se integra sin problemas con GitHub y automáticamente ejecuta el pipeline definido en cada push o pull requests. En este caso, ejecutará las pruebas que tengamos creadas en los tests en el propio código.

6.2.4. Visual Paradigm

Programa que proporciona una gran facilidad a la hora de realizar diagramas para el diseño y análisis de proyectos software. Se ha utilizado la versión académica para el desarrollo de diagramas UML, así como para los casos de uso.

6.2.5. Draw.io

Aplicación web que permite crear diagramas de todo tipo desde el navegador sin falta de instalar ningún programa. Se ha utilizado para el diseño de la interfaz de la aplicación. Dispone de una serie de “mockups” para todo tipo de UI (User Interface).

6.2.6. ModbusPal.jar

Es un entorno de pruebas que simula una red Modbus y permite realizar las pruebas en local en vez de tener que atacar sistemas en producción.

6.2.7. Snap7

Se trata de un entorno de pruebas que simula una red Siemens S7 y con ello, se pueden realizar las pruebas en local en vez de tener que atacar sistemas en producción.

6.2.8. Enterprise Architect

Es una herramienta de diseño y modelado visual para diagramas UMLque soporta el diseño y construcción de sistemas de software, modelado de procesos de negocio y modelado de dominios. Se ha utilizado para el diagrama de paquetes y de clases ya que proporciona mayores ventajas que Visual Paradigm. Se ha usado la versión de prueba de 30 días de la aplicación para el desarrollo de estos.

Capítulo 7. Desarrollo de las Pruebas

7.1. Integración continua

Para el desarrollo del proyecto, se ha configurado el repositorio de código de Github con Travis-CI para que al realizar un *push*, se ejecuten una serie de pruebas para comprobar el correcto funcionamiento de la aplicación. Estas son probadas en una máquina virtual de la plataforma y de esta forma, se puede comprobar que funciona perfectamente en cualquier máquina que cumpla los requisitos y tenga instalados los requerimientos de la aplicación. En el propio repositorio, se dispone de un enlace que lleva a Travis-CI donde se puede ver el historial de pruebas llevadas a cabo.

7.2. Resultado de las pruebas

Se describe el resultado de las pruebas unitarias diseñadas y descritas anteriormente en el apartado *4.8. Especificación del Plan de Pruebas*, así como los resultados obtenidos.

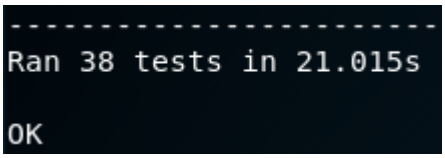

Figura 7.1. Resultado de la ejecución de las pruebas

Capítulo 8. Manuales del Sistema

8.1. Manual de Instalación

Clonar el repositorio de Github:

```
git clone https://github.com/dave36/pdt.git
```

Instalar los requerimientos de la aplicación:

```
pip install -r requirements.txt
```

8.2. Manual de Ejecución

Una vez descargada la aplicación e instaladas las dependencias, bastará con ejecutar:

```
python pdt.py
```

Si se quieren ejecutar los tests, sería de la siguiente forma (desde el directorio raíz del proyecto):

```
python -m unittest discover -s tests
```

8.3. Manual de Usuario

Las formas de uso de la herramientas y ejemplos con los resultados que se obtienen, se encuentran disponibles en la Wiki del repositorio de Github en https://github.com/dave36/pdt/wiki.

Capítulo 9. Conclusiones y Ampliaciones

9.1. Conclusiones

Tras el desarrollo del proyecto, se ha llegado a diseñar e implementar una herramienta que ayuda al auditor en aquellas pruebas que no son típicas de Tecnologías de la Información y están más relacionadas con los Sistemas de Control Industrial donde el campo no es tan familiar como en las auditorías de seguridad en TI.

La solución está dividida en una serie de menús para facilitar la tarea al usuario y en las partes correspondientes a un test de penetración habitual: recolección de información pública (footprinting), interacción con el sistema (fingerprinting), explotación y reporte.

En esta primera versión del proyecto, se ha llegado a desarrollar una serie de pruebas para dos de los protocolos más usados actualmente en Sistemas de Control Industrial, Modbus y Siemens S7. Esto se podría ampliar, bastaría con añadir un mayor número de pruebas para tratar otro tipo de protocolos y dispositivos.

Además, la aplicación está pensada para el uso en entornos de pentesting como puede ser Kali Linux y está integrada con herramientas de seguridad ampliamente utilizadas y reconocidas.

9.2. Ampliaciones

Como se ha explicado a lo largo del documento, la aplicación actualmente abarca dos protocolos de Sistemas de Control Industrial ampliamente utilizados que son Modbus y Siemens S7. En futuras versiones, se podría ampliar y añadir nuevas pruebas para otro tipo de protocolos como pueden ser BACnet, DNP3 o Profibus. Bastaría con extender la funcionalidad existente en el apartado correspondiente de la auditoría en el que se llevaría a cabo.

Otra ampliación, sería la forma de reporte que se le da al usuario a través del informe. Actualmente se muestra los pasos que realizó con sus resultados. En posteriores versiones, se podría ampliar para que realizase completos informes en formato HTML o PDF como realizan algunas herramientas de escaneo de vulnerabilidades como puede ser Nessus, pero que no se ha llevado a cabo por falta de tiempo y de un equipo completo de desarrollo.

Capítulo 10. Presupuesto

10.1. Unidades de Trabajo

Las unidades de trabajo realizadas para el desarrollo del proyecto son las siguientes (las horas de cada una de ellas se han obtenido de la planificación del proyecto en el apartado 3.1):

- Preparación propuesta TFM
- Estudio de la situación actual
- Formación en seguridad del equipo
- Diseño de la Arquitectura del Sistema
- Creación de la Interfaz
- Implementación
- Pruebas unitarias
- Pruebas en entornos simulados
- Elaboración de la documentación
- Creación de manuales del sistema

10.2. Cálculo de Costes

10.2.1. Costes Directos

Estos son los precios definidos para los recursos de trabajo y de materiales.

Recursos de trabajo

COD	UD	DESCRIPCIÓN	PRECIO
RT01	H	Ingeniero Informático	35€

Recursos materiales

COD	UD	DESCRIPCIÓN	PRECIO
RM01	h	Ordenador	0,5€
RM02	h	Monitor	0,25€

10.2.2. Precios Unitarios

Tras definir los costes directos del proyecto, se calcula el precio de la hora:

COD	UD	DESCRIPCIÓN	PRECIO
PU01	h	Ingeniero Informático	35€

PU02	h	Ordenador	0,5€
PU03	h	Monitor	0,25€
		Importe/hora	**35,75€**

10.2.3. Precios por Unidades de Trabajo

A partir de los precios definidos, se calculan los que corresponden a cada unidad de trabajo, previamente descritas en el apartado 10.1.

Preparación propuesta TFM			
Medición	**Concepto**	**Coste Unitario**	**Total**
15h	Ingeniero Informático	35,75€	536,25€
		Total	**536,25€**

Estudio de la situación actual			
Medición	**Concepto**	**Coste Unitario**	**Total**
17h	Ingeniero Informático	35,75€	607,75€
		Total	**607,75€**

Formación en seguridad del equipo			
Medición	**Concepto**	**Coste Unitario**	**Total**
22h	Ingeniero Informático	35,75€	786,5€
		Total	**786,5€**

Diseño de la Arquitectura del Sistema			
Medición	**Concepto**	**Coste Unitario**	**Total**
34h	Ingeniero Informático	35,75€	1.215,5€
		Total	**1.215,5€**

Creación de la Interfaz			
Medición	**Concepto**	**Coste Unitario**	**Total**
2h	Ingeniero Informático	35,75€	71,5€
		Total	**71,5€**

Implementación			
Medición	**Concepto**	**Coste Unitario**	**Total**
78h	Ingeniero Informático	35,75€	2.788,5€
		Total	**2.788,5€**

Pruebas unitarias			
Medición	**Concepto**	**Coste Unitario**	**Total**
38h	Ingeniero Informático	35,75€	1.358,5€
		Total	**1.358,5€**

Pruebas en entornos simulados			
Medición	**Concepto**	**Coste Unitario**	**Total**
12h	Ingeniero Informático	35,75€	429€
		Total	**429€**

Elaboración de la documentación			
Medición	**Concepto**	**Coste Unitario**	**Total**
45h	Ingeniero Informático	35,75€	1.608,75€
		Total	**1.608,75€**

Creación de manuales del sistema			
Medición	**Concepto**	**Coste Unitario**	**Total**
14h	Ingeniero Informático	35,75€	500,5€
		Total	**500,5€**

10.2.4. Costes Indirectos

Los costes indirectos del proyecto se estiman en un 15% del coste total del proyecto y en ellos están incluidos los siguientes gastos:

- Electricidad
- Local
- Internet
- Material de oficina
- Personal administrativo

10.3. Presupuesto Interno

Medición	Concepto	Coste Unitario	Total
1	Preparación propuesta TFM	536,25€	**536,25€**
1	Estudio de la situación actual	607,75€	**607,75€**
1	Formación en seguridad del equipo	786,5€	**786,5€**
1	Diseño de la Arquitectura del Sistema	1.215,5€	**1.215,5€**
1	Creación de la Interfaz	71,5€	**71,5€**
1	Implementación	2.788,5€	**2.788,5€**
1	Pruebas unitarias	1.358,5€	**1.358,5€**
1	Pruebas en entornos simulados	429€	**429€**
1	Elaboración de la documentación	1.608,75€	**1.608,75€**
1	Creación de manuales del sistema	500,5€	**500,5€**
		SUMA	9.902,75€
		Costes indirectos (15%)	1.485,41€
		Beneficios (20%)	1.980,55€
		Total	**13.368,71€**

10.4. Presupuesto del Cliente

Concepto	Precio
Preparación propuesta TFM	723,94€
Estudio de la situación actual	820,46€
Formación en seguridad del equipo	1.061,78€
Diseño de la Arquitectura del Sistema	1.640,93€
Creación de la Interfaz	96,53€
Implementación	3.764,48€
Pruebas unitarias	1.833,98€
Pruebas en entornos simulados	579,15€
Elaboración de la documentación	2.171,81€
Creación de manuales del sistema	675,68€
Total	**13.368,71€**

Capítulo 11. Referencias Bibliográficas

11.1. Libros y Artículos

Alonso, C., González, P., Palop, I., Rando, E., Alonso, R., Moreno, J. y Fernández, M. (2018). Pentesting con FOCA. Madrid: 0xWord.

González, P. (2015). Ethical Hacking. Madrid: 0xWord.

Bolívar, J.F. (2016). Infraestructuras críticas y sistemas industriales. Auditorías de seguridad y fortificación. Madrid: 0xWord.

NIST Special Publication 800-82 (2015). Guide to Industrial Control Systems (ICS) Security.

NIST Special Publication 800-53 (2013). Security and Privacy Controls for Federal Information Systems and Organizations.

Kaspersky Report (2015). Industrial Control Systems Vulnerabilities Statistics. Kaspersky Lab.

11.2. Referencias en Internet

Nilesh Kapoor, N.K. (2017), SCADA Penetration Testing: Do I need to be prepared? Recuperado de https://research.aurainfosec.io/scada-penetration-testing/

Soullie, A. Pentesting ICS. ICS tools. Recuperado de https://pentesting-ics.com/ics-tools/

INCIBE (2015), IEC 62443: Evolución de la ISA 99. Recuperado de https://www.incibe-cert.es/blog/iec62443-evolucion-isa99

Official website of the Department of Homeland Security. Industrial Control Systems. Recuperado de https://www.us-cert.gov/ics

INCIBE (2015), Normativas de seguridad en sistemas de control. Recuperado de https://www.incibe-cert.es/blog/normativas-seguridad-sistemas-control

Shodan. Motor de búsqueda para dispositivos conectados a Internet. Recuperado de https://www.shodan.io/

Simply Modbus (2019). Data Communication Test Software. Recuperado de http://www.simplymodbus.ca/index.html

Capítulo 12. Código Fuente

El código fuente está disponible en el siguiente repositorio de GitHub:

https://github.com/dave36/pdt