UNIVERSIDAD POLITÉCNICA DE CARTAGENA

Escuela Técnica Superior de Ingeniería Industrial

Desarrollo de un sistema demo de hacking ético para Autómatas Programables industriales y SCADA

TRABAJO FIN DE GRADO

GRADO EN INGENIERÍA ELECTRÓNICA INDUSTRIAL Y AUTOMÁTICA

Autor: Antonio Fco Pastor Férez
Director: Héctor David Puyosa Piña

Cartagena, 01 Febrero 2017

AGRADECIMIENTOS:

Después de un intenso período de cuatro años, hoy es el día: escribo este apartado de agradecimientos para finalizar mi trabajo fin de grado y con este mis estudios en grado de ingeniería. Ha sido un período de aprendizaje intenso, no solo en el campo científico, sino como reto a nivel personal. Obtener esta titulación ha tenido un gran impacto en mí y es por eso que me gustaría agradecer a todas aquellas personas que me han ayudado y apoyado durante este proceso.

Primero de todo, me gustaría agradecer a mis compañeros de carrera por ayudarme siempre que lo he necesitado. Me habéis apoyado enormemente y siempre habéis estado ahí para ayudarme cuando lo necesitaba. Particularmente me gustaría nombrar a mi grupo más íntimo, Ana, Iván, Raquel, Varinia y Héctor. Me gustaría agradecerles toda la ayuda y ánimos en épocas de desánimo y gran estrés.

Además, me gustaría darle las gracias a mi tutor Héctor David Puyosa Piña, por su valiosa ayuda. Y a los demás profesores que me han levantado pasión por sus asignaturas como, Miguel Almonacid Kroeger, Jose Alfonso Vera Repullo y Pedro Díaz Hernández entre otros. Definitivamente me habéis brindado todas las herramientas necesarias para completar mis estudios y trabajo de fin de grado satisfactoriamente.

También me gustaría agradecer a mis padres por sus sabios consejos y su comprensión. Siempre habéis estado ahí para mí. Y es muy posible que no hubiera llegado tan lejos sin su ayuda. Ya que en la primera semana de clase ya pensaba en no volver y ellos me hicieron luchar por este reto y sacar mejores notas día tras día.

Finalmente, decirles a mis compañeros que ya los considero mis amigos. No solo habéis estado ahí para apoyarnos entre nosotros en los momentos difíciles, sino que también hemos tenido descansos y días de todo sobre otras cosas no relacionadas con los estudios.

¡Muchas gracias a todos!

ÍNDICE:

Índice de figuras:

1. INTRODUCCIÓN:

El mundo de las Tecnologías de la Información y Comunicación (TIC) evoluciona cada día más rápido. La utilización de servicios TIC en todas las tareas cotidianas de nuestra vida es cada vez mayor. La mayor parte de las personas utilizan servicios de Internet para hacer sus gestiones, conectarse con sus conocidos, o acceder a su correo electrónico. Con esta gran dependencia que cada vez tenemos más del mundo informático, es normal que también nos preocupe la seguridad de todos estos servicios que nos ofrece Internet.

Por tanto, se puede decir que toda persona preocupada por estar al día en el mundo de las TIC, que se preocupa por aprender y reciclarse, descubriendo los agujeros de seguridad de los sistemas, es un hacker. Esta es la filosofía del verdadero hacker. No importa lo que podemos conseguir al encontrar un fallo de seguridad en alguna aplicación o bien algún agujero que pueda afectar al servicio, en ese caso lo más importante es el hecho de haber logrado identificar un fallo y ayudar a mitigar ese problema. El gran reto del profesional de seguridad informática está en el descubrimiento de la vulnerabilidad y bloquearla.

La seguridad de la información en sistemas de control industrial (SCI) y, por extensión, la protección de infraestructuras críticas, ha tomado en los últimos años importancia cada vez mayor en algunos países.

Los sistemas de control industrial más conocidos son los llamados SCADA (Control de supervisión y Adquisición de Datos) y DCS (Sistemas de control distribuidos). De manera genérica, los sistemas SCADA son redes de sistemas de control industrial que contienen computadoras y aplicaciones que realizan funciones clave para el suministro de servicios esenciales para una nación, por ejemplo, agua, energía, gas, gasolina o petróleo.

Estos sistemas inicialmente se diseñaron para trabajar aislados, pero no han escapado a la imperiosa necesidad de interconectarse con los aplicativos de negocio para facilitar la toma de decisiones, heredando con ello el mundo de vulnerabilidades de los sistemas abiertos. Cualquier infraestructura de red gubernamental o industrial basada en SCADA está sujeta a potenciales ataques, razón por la cual hoy día se requiere tratar el tema de seguridad en sistemas de control industrial.

Recientemente, en enero de 2016, se dio a conocer un informe del US-CERT que indica que el número de ataques cibernéticos registrados en 2015 creció 52%. Los sistemas de electricidad, de agua y nucleares son el principal blanco de los cibercriminales. Durante 2015 se registraron 198 ataques, de los cuales 82 fueron en contra del sector energético, 29 contra del de agua, 7 a instalaciones químicas y 6 contra plantas nucleares. Estas cifras corresponden a ataques denunciados, y puesto que la mayoría de empresas opta por no denunciar las brechas de seguridad, una gran cantidad de ataques permanece desconocida.

En España, ha sido necesario tener evidencias de amenazas dirigidas a los entornos de control industrial para llegar a un nivel de concienciación adecuado por parte de las compañías que operan las infraestructuras industriales

Con este propósito nace este proyecto, que se basa en conocer las metodologías aplicadas en la práctica del Hacking Ético y utilización de herramientas de seguridad para encontrar vulnerabilidades y así evitarlas, con el fin de tener una red de automatización a través de nuestros PLCs interconectados a una misma red en nuestra empresa sin miedo a ser Hackeados.

Inicialmente explicaremos algunos conceptos teóricos, tipos de vulnerabilidades, cómo se detectan, metodologías que hay que seguir, herramientas principales, buenas prácticas, instalación del laboratorio y ejecución de pruebas.

En la parte de diseño del trabajo, se ha utilizado la red del laboratorio de robótica, que es una red segura y los PLCs necesarios para probar el Hacking. Dentro de este entorno se propone definir un ejemplo de la intrusión a un PLC.

Se instalará un sistema operativo Linux para conseguir la clave Wifi del laboratorio, para estudio de vulnerabilidad se usará otro programa y se enseñará posibles ataques que se pueden realizar.

2. ANTECEDENTES

Este proyecto se basa en sistemas de seguridad de la red de control y de producción aplicado al ámbito de la electrónica industrial y automática por ello además de atacar una red Wifi se va a modificar los datos de un PLC, esta idea ya fue conseguida hace pocos años y tuvo un antes y un después en cuanto a seguridad nacional se refiere, ya que se denominó Stuxnet y consiguió manejar una central nuclear.

Stuxnet fue un gusano informático que afectó a equipos con Windows, descubierto en junio de 2010 por VirusBlokAda, una empresa de seguridad ubicada en Bielorrusia. Es el primer gusano conocido que espío y reprogramó sistemas industriales, en concreto sistemas SCADA de control y monitorización de procesos, pudiendo afectar a infraestructuras críticas como centrales nucleares.

Stuxnet fue capaz de reprogramar controladores lógicos programables y ocultar los cambios realizados. También es el primer gusano conocido que incluye un rootkit para sistemas reprogramables PLC.

La compañía europea de seguridad digital Kaspersky Lab describía a Stuxnet en una nota de prensa como "un prototipo funcional y aterrador de un arma cibernética que conducirá a la creación de una nueva carrera armamentística mundial". Kevin Hogan, un ejecutivo de Symantec, advirtió que el 60% de los ordenadores contaminados por el gusano se encuentran en Irán, sugiriendo que sus instalaciones industriales podrían ser su objetivo. Kaspersky concluye que los ataques sólo pudieron producirse "con el apoyo de una nación soberana", convirtiendo a Irán en el primer objetivo de una guerra cibernética real.

El objetivo más probable del gusano, según corroboran medios como BBC o el Daily Telegraph, pudieron ser infraestructuras de alto valor pertenecientes a Irán y con sistemas de control de Siemens. Medios como India Times apuntan que el ataque pudo haber retrasado la puesta en marcha de la planta nuclear de Bushehr. Fuentes iraníes han calificado el ataque como "guerra electrónica" aunque minimizan el impacto de los daños en sus instalaciones. Algunos medios como el norteamericano New York Times han atribuido su autoría a los servicios secretos estadounidenses e israelíes.

STUXNET

En enero de 2010, los inspectores de la Agencia Internacional de Energía Atómica que visitaban una planta nuclear en Natanz, Irán, notaron con desconcierto que las centrifugadoras usadas para enriquecer uranio estaban fallando. Curiosamente, los técnicos iraníes que reemplazaban las máquinas también parecían asombrados.

El fenómeno se repitió cinco meses después en el país, pero esta vez los expertos pudieron detectar la causa: un malicioso virus informático.

El "gusano" - ahora conocido como Stuxnet - tomó el control de 1.000 máquinas que participaban en la producción de materiales nucleares y les dio instrucciones de autodestruirse.

Fue la primera vez que un ataque cibernético logró dañar la infraestructura del "mundo real".

Durante el análisis del gusano, los analistas hicieron un descubrimiento sorprendente, señala Gordon Corera, corresponsal de temas de seguridad de la BBC. El código altamente avanzado de Stuxnet había sido diseñado con una mentalidad bélica.

A continuación, se resume el ciberataque en cuatro pasos:

1. **Stuxnet penetró en la red**

   Según la firma de seguridad cibernética Symantec, Stuxnet probablemente llegó al programa nuclear de Natanz de Irán en una memoria USB infectada.

   Alguien habría tenido que insertar físicamente el USB a una computadora conectada a la red. El gusano penetró así en el sistema informático de la planta.

2. **El gusano se propagó a través de las computadoras**

   Una vez dentro del sistema informático, Stuxnet buscó el software que controla las máquinas llamadas centrifugadoras.

   Las centrífugas giran a altas velocidades para separar componentes. En la planta de Natanz, las centrifugadoras estaban separando los diferentes tipos de uranio, para

aislar el uranio enriquecido que es fundamental tanto para la energía como para las armas nucleares.

3. Stuxnet reprogramó las centrifugadoras

El gusano encontró el software que controla las centrifugadoras y se insertó en él, tomando el control de las máquinas.

Stuxnet llevó a cabo dos ataques diferentes. En primer lugar, hizo que las centrifugadoras giraran peligrosamente rápido, durante unos 15 minutos, antes de volver a la velocidad normal. Luego, aproximadamente un mes después, desaceleró las centrifugadoras durante unos 50 minutos. Esto se repitió en distintas ocasiones durante varios meses.

4. Destrucción de las máquinas

Con el tiempo, la tensión provocada por las velocidades excesivas causó que las máquinas infectadas, unas 1000, se dañasen.

Durante el ataque cibernético, alrededor del 20 por ciento de las centrifugadoras en la planta de Natanz quedaron fuera de servicio.

INTRUSIÓN EN LA RED

El gusano aprovechó cuatro debilidades previamente desconocidas en el sistema operativo Windows de Microsoft. Una ayudó a Stuxnet a llegar a la red a través de una memoria USB y otra usó impresoras compartidas para penetrar más profundamente. Las dos restantes le permitieron a Stuxnet controlar otras partes menos seguras de la red.El gusano fue programado específicamente para apuntar y destruir las centrifugadoras.

Una vez dentro del sistema de Natanz, Stuxnet escaneó todas las computadoras con sistema operativo Windows que estaban conectadas a la red, en busca de un determinado tipo de circuito llamado Programmable Logic Controller (Controlador Lógico Programable) o PLC, que controla las máquinas. En este caso, el PLC que fue blanco del ataque controlaba la velocidad específica de las centrifugadoras.
A diferencia de la mayoría de los gusanos informáticos, Stuxnet no hizo nada en las computadoras que no cumplían con requisitos específicos. Pero una vez que encontró lo que estaba buscando, se insertó en los PLC, listo para tomar el control de las centrifugadoras.

MODO DE ATAQUE

Para infiltrarse en el sistema sin ser detectado, el gusano utiliza una "firma digital" -una clave larga, cifrada, robada de piezas genuinas de software- para parecer legítimo. Windows suele comprobar esas claves cuando se instalan nuevos programas. Usando ese modo de acceso, Stuxnet se deslizó sin generar sospechas.

El gusano permaneció latente durante casi un mes después de infectar el PLC de las máquinas. En ese tiempo observó cómo opera el sistema normalmente y registró los datos generados.

Una vez las centrifugadoras en Natanz quedaron fuera de control, el gusano reprodujo los datos grabados cuando todo estaba funcionando normalmente.

Esto permitió que permaneciera indetectable por los operadores humanos de la fábrica, mientras las centrifugadoras quedaban destruidas.

Stuxnet fue incluso capaz de anular los interruptores de apagado de emergencia, cuando los operadores de las centrifugadoras se percataron de que las cosas estaban fuera de control, Stuxnet contenía un código que impidió el apagado de las máquinas.

Todavía se desconoce con seguridad quién o quiénes fueron responsables de la creación de Stuxnet. Symantec considera que se necesitaron entre 5 y 10 expertos en software, que trabajaron hasta 6 meses para crear el sofisticado gusano cibernético.

SOLUCIÓN

Siemens ha puesto a disposición del público una herramienta de detección y eliminación de Stuxnet. Siemens recomienda contactar con su soporte técnico si se detecta una infección, instalar los parches de Microsoft que eliminan las vulnerabilidades de Windows y prohibir en las instalaciones industriales el uso de memorias USB ajenas o no controladas. También la empresa BitDefender ha desarrollado una herramienta gratuita para eliminar Stuxnet.

ORIGEN

Un portavoz de Siemens declaró que el gusano Stuxnet fue encontrado en 15 sistemas, cinco de los cuales eran plantas de fabricación en Alemania. Según Siemens, no se han encontrado infecciones activas y tampoco existen informes de daños causados por el gusano.

Symantec afirma que la mayor parte de los equipos infectados estaban en Irán, lo que ha dado pie a especulaciones de que el objetivo del ataque eran infraestructuras "de alto valor" en ese país, incluyendo la Central Nuclear de Bushehr o el Complejo Nuclear de Natanz.

En 2011, Ralph Langner, un investigador alemán de seguridad informática, dijo que Stuxnet era un arma diseñada "para disparar un solo tiro" y que el objetivo deseado por sus creadores fue probablemente alcanzado, aunque ha admitido que esto son solo especulaciones. Bruce Schneier, otro investigador en seguridad, ha calificado estas teorías como interesantes, si bien señala que existen pocos datos objetivos para fundamentarlas.

Algunos especialistas (pendiente de encontrar referencias) señalaban a Israel como principal sospechoso, y en concreto a la Unidad 8200 de las Fuerzas de Defensa de Israel.

Finalmente, el New York Times eliminó todo rumor o especulación confirmando que se trata de un troyano desarrollado y financiado por Israel y Estados Unidos con el fin de atacar las centrales nucleares iraníes. Aun así las autoridades, entre ellas el equipo más especializado del FBI nunca encontró al culpable.

3. REQUERIMIENTOS:

Para conseguir el propósito de este proyecto se necesitará varias herramientas básicas como pueden ser, un portátil potente para poder arrancar TIA PORTAL V13 y sus complementos (imagen inferior) aparte de otros programas para hacer uso del Hacking ético, un PLC S7 1200 y una red local segura como la que tenemos en el laboratorio.

Herramientas al detalle:

- **Software:** Linux*, Wifislax, Windows 7, Nmap, plataforma virtualización Virtual box,

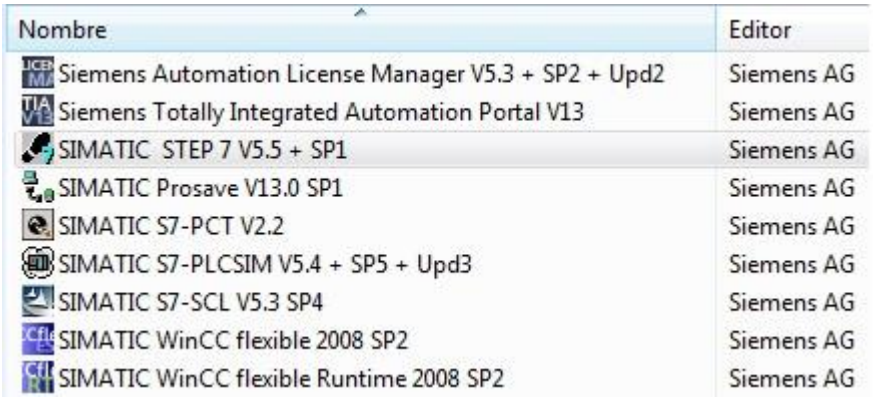

Figura 1.- Tia portal V13 y todas sus extensiones y licencias.

- **Hardware**:
  - Pc: procesador Intel core i5 2.5GHZ, memoria RAM* 8Gb, disco duro de 1Tb.
  - PLc: S7 1200 CPU 1214C AC/DC/RLy

- **Datos**: se utiliza un Pen drive para cargar el Wifislax y guardar datos y capturas del programa de Siemens.

4. PLANIFICACIÓN:

Se crea una planificación en diagrama de GANTT de manera resumida con la planificación esperada por tareas principales, esta planificación puede sufrir cambios debido a algún retraso, posibles complicaciones o desvío del objetivo, pero siempre centrado en el tema principal que es el hacking ético de nuestro PLC.

1. Lectura de documentación técnica y compilación de información. Presentación de documentos y toma de decisión. Ajustes técnicos con el Tutor.
2. Lectura de documentación, redacción de textos científicos, redactar los primeros puntos para la elaboración del índice del TFE. Documentación teórica y técnica de aspectos relacionados con la seguridad de redes y hacking.
3. Revisar documentación técnica y teórica. Instalación y configuración de máquina virtual con wifislax. Preparar laboratorio, servidores necesarios y herramientas.
4. Poner en producción máquinas virtuales y comprobar funcionamiento. Realizar documentación de pruebas e inicio de análisis de vulnerabilidad real con PLCs
5. Estudio comparativo de herramientas. Estudio de Metodología de seguridad.
6. Estudio de vulnerabilidades en redes WiFi
7. Pruebas finales.
8. Revisión de la memoria.

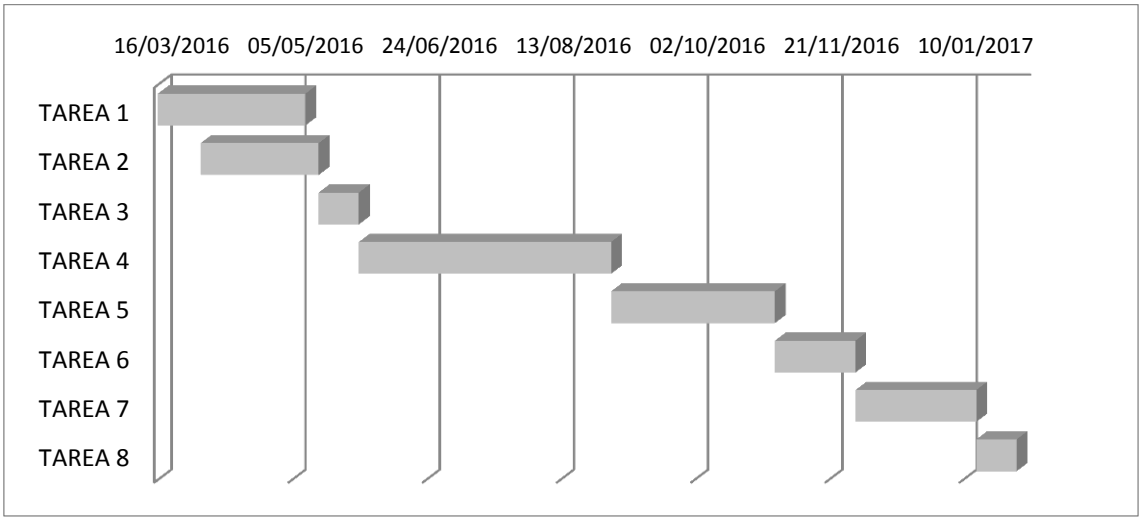

Figura 2.- Planificación del proyecto

HACKING ÉTICO

Se tendrá en cuenta que hacking informático y hacking ético son dos cosas muy distintas. Existen varias leyes que regulan muchos de los aspectos que son utilizados en esta área. Podemos entender que el hacking ético son técnicas utilizadas por profesionales de la seguridad para ayudar a las defensas de los ataques informáticos. A diferencia del hacker informático, el Hacking ético se basa en encontrar soluciones en seguridad informática, realizando pruebas en redes y buscando vulnerabilidades, para después reportarlas y que se tomen medidas, sin hacer daño.

Para realizar estas pruebas se utilizan herramientas que se llaman "*pen tests*" o "*penetration tests*" (pruebas de penetración).

En la mayoría de los casos las herramientas que utilizan los Hackers (Atacantes maliciosos), son las mismas que utilizan los profesionales de la seguridad. Por eso es fácil de entender que el profesional que trabaja como Hacker Ético, siguen los mismos procedimientos y procesos que los Hackers que no son éticos.

El Hacker ético tiene que saber lo que hacen los Hackers no éticos, tiene que conocer las técnicas que utilizan, cómo actúan y también mantenerse siempre actualizado porque cada día aparecen nuevas técnicas.

La palabra Hacker se utiliza para nombrar a aquella persona experta en el mundo de la informática, que se relaciona con el software, los sistemas, las redes, los sistemas operativos, etc. El término se ha degradado hoy en día, por las acciones maliciosas de personas que utilizan el conocimiento para sus fines.

Se podría decir que toda persona que se preocupa por estar al día en el mundo de la informática, aprender y reciclarse, y es autodidacta podría ser un buen candidato a ser un hacker.

El gran reto para un verdadero hacker está en el descubrimiento de la vulnerabilidad, no en su explotación. Por eso que todas las personas que tras una vulnerabilidad desarrollan aplicaciones para explotarlas ponen en peligro la información de millones de personas en Internet y nos hacen mal, pues se debe tener una red segura para que los usuarios confíen en ella y cada día sea más utilizada y difundida.

4.1. Piratería y Hacker

La piratería está relacionada con la forma ilegal de hacer copias de software, con lo cual viola el derecho del autor. Pero también puede hacer referencia a los hacker que se dedican a robar informaciones por Internet. La piratería se puede ver desde distintos puntos:

1. *Como piratería del usuario final*. Se trata de la más común de todas, donde se realiza copias de un determinado software sin hacer uso de la licencia. Por ejemplo hacer copia del Windows y pasarla a un amigo.

2. *Piratería de carga de disco duro*. Las empresas venden los ordenadores con software previamente instalado sin proveer las licencias.

3. *Piratería de Internet.* Se trata de la distribución de software por Internet de forma no autorizada, también podría agregar los ataques y robos de información en la red tales como robo de número de tarjetas de crédito, clonaje, suplantaciones de ID, etc.

5. VULNERABILIDADES INFORMÁTICAS

En informática el termino vulnerabilidad informática hace referencia a cualquier debilidad en un sistema o medio físico, donde permite que un determinado atacante pueda violar la integridad, hacer daños en los datos, robar la información, alterar aplicaciones y fraudes.

La vulnerabilidad puede ser un pequeño bug en una aplicación, fallo en el desarrollo o hasta algún descuido por parte del usuario como utilizar contraseñas fácil de descifrar o dejarlas apuntadas en algún sitio de fácil acceso.

5.1. Tipos de vulnerabilidades

Hay innumerables vulnerabilidades informáticas y según sus características se clasifican en tipos distintos. A continuación, se comentan algunas de las vulnerabilidades más conocidas:

5.1.1. Vulnerabilidad de desbordamiento de buffer

Se trata de un tipo de ataque muy común entre los hackers donde se puede causar un problema muy serio en los sistemas. Existen innumerables tipos de ataques de desbordamiento pero el más común es el de la pila. Este ataque consiste cuando un determinado programa por fallo en su implementación no es capaz de controlar la cantidad de datos que están en el buffer, haciendo que por final ultrapase la capacidad del buffer. Debido a ese fallo los datos son movidos a otro lado sobrescribiendo o modificándolos, con eso se puede conseguir tener un control del propio sistema. Ese tipo de ataque se hace muy común porque la mayoría de los sistemas están desarrollado en C, porque en su diseño el lenguaje ha priorizado espacio y rapidez sobre seguridad. Según los expertos en seguridad informática, ese tipo de ataque será durante muchos años uno de los más importantes.

5.1.2. Inyección de código

Se trata de una vulnerabilidad que está basada en la existencia de parámetros de una determinada aplicación que no son validados de manera correcta. Entonces el atacante aprovecha para lanzar algunos valores de parámetros dinámicos que

irán a enlazar con la base de datos de la aplicación. Trata de entrar con algunos valores que no son validados previamente por los desarrolladores pero que podrá modificar el comportamiento de la aplicación, como por ejemplo hasta devolver una contraseña o bien validar un usuario.

5.1.3. Vulnerabilidad de Cross Site XSS

Este tipo de vulnerabilidad ocurre cuando los campos de entrada de las aplicaciones no disponen de ningún tipo de protección y con eso permite introducir o enviar datos sin ningún tipo de validación. Se realiza creando scripts que actuaran sobre etiquetas HTML*. En este tipo de vulnerabilidad podemos dividir dos grupos según el tipo de ataque que se realice.

5.1.4. Ataque al usuario

Es el tipo de ataque más utilizado y donde se puede conseguir más informaciones como las cookies de los usuarios. También se puede utilizar el ataque por vía del correo electrónico, donde se envía un correo con un link incrustado que tiene un script. La idea es hacer que a través de este link pueda motivar al usuario a ejecutarlo y poner en machar algún comando malicioso. Para complementar el ataque del usuario, hablemos de la publicación en sitio Web vulnerables que consiste en la incluir algún dato en libro de visita, foros, blogs o cualquier sitio Web, donde permite al usuario introducir sus datos sin ser validados por el servidor, con esto se permite robar alguna información del propio usuario.

5.1.5. Ataque a la aplicación

Consiste en introducir en alguna aplicación Web vulnerable algún código malicioso, normalmente las aplicaciones Web están protegidas pero siempre es posible hacer el ataque mediante algún tag que no está previsto. Los tags son etiquetas internas que ponemos en los códigos.

5.1.6. Crack

Es la técnica donde los hackers consiguen sacar los códigos de registro de un determinado programa y con eso poder validar la aplicación para su uso. La

función principal del proceso es poder instalar una aplicación que está protegida mediante su registro. Para realizar esta técnica se requiere mucho conocimiento de programación a bajo nivel, porque en la mayoría de los casos se tiene que desensamblar algún ejecutable.

5.1.7. Ingeniería Social

Consiste en una manera que tienen los hackers para engañar a los usuarios haciéndose pasar por otras personas o entidades. Existen innumerables tipos de ingeniería social, una de las más conocidas es la técnica del Phising. Con esa técnica podemos conseguir credenciales de algún usuario, como contraseña, información de la tarjeta de crédito o hasta informaciones bancarias. El nombre se origina de la palabra pescar (fishing) en inglés. Esta técnica consiste en suplantar la identidad con fines maliciosos.

5.1.8. Condición de carrera

Ocurre cuando varios procesos intentan acceder al mismo tiempo en un determinado recurso compartido.

5.1.9. Redes Wi-Fi

Según estudios realizados cerca de 20% de las redes WiFi son fácilmente atacables, el principal motivo es porque todavía utilizan un mecanismo de autentificación y de encriptación obsoleto (WEP) y también por desconocimiento de los usuarios.

5.2. Como detectar la vulnerabilidad

Existen varias maneras para detectar vulnerabilidades, pero lo más correcto es seguir alguna metodología o estructura adecuada para llevarla a cabo.
Casi todas las metodologías trabajan con estas fases. Las metodologías las trataremos con más detalle más adelante en auditorias.

1. Fase de reconocimiento: desarrollar información sobre la red y el objetivo, búsqueda de los dominios, equipos, topología.
2. Fase de escaneo de puertos: detectar los puertos de los equipos de la red.

3. Fase de enumeración de servicios: identificar los servicios que están trabajando en cada equipo.
4. Fase de escaneo de vulnerabilidades: la fase más importante donde se detectan las vulnerabilidades en cada elemento de la red.

En cada fase se obtiene información que será útil para la siguiente fase. En cada una de las fases se trabaja con una herramienta adecuada.

Hay diversas herramientas que se utilizan para detectar vulnerabilidades de manera automática como Nmap, donde se puede realizar un análisis de vulnerabilidades de la red: identificar los puntos débiles, analizar los datos de cada exploración, detectar otros dispositivos en la misma red, etc.

5.3. Como se divulga la vulnerabilidad de manera ética.

Desafortunadamente todo software que sale al mercado están llenos de fallos. Estos fallos pueden ocasionar muchos problemas de seguridad en una empresa o cualquier usuario.

Dentro de ese proceso se encuentra por un lado la empresa que es la que tiene toda la información del código del producto con fallos, y por otro lado tiene el cliente. Lo que ocurre es que la empresa que vende el producto no quiere hacer pública la información confidencial de su producto porque puede radicar en innumerables problemas. Por eso hay que pensar en alguna manera de solucionarlo.

El primer problema sería que los detalles del fallo ayudarán a que los hackers ataquen la vulnerabilidad. El argumento del vendedor se basa en que si el asunto se mantiene confidencial mientras se desarrolla una solución, los atacantes no sabrán cómo aprovechar el fallo.

El otro problema sería que la publicación de ese fallo podría dañar la reputación de la empresa.

Para eso los hackers ético tienen que saber actuar de manera ética y saber utilizar los métodos correctos para revelar ese fallo de seguridad. No se trata de realizar un ataque y después explicar a otras personas como lo hacen, sino de trabajar en conjunto con el propietario del software para resolver la vulnerabilidad.

En este punto se hablará sobre el proceso que se tiene que tomar de manera ética cuando se detecta un bug en alguna aplicación o bien la vulnerabilidad. El proceso para la divulgación de la vulnerabilidad al público ha creado una enorme discusión entre las empresas informáticas, cada una ve el tema de manera muy distinta. Debido a los diferentes puntos de vistas varias empresas se han unido para crear políticas, e indicaciones de cómo proceder en la divulgación de la vulnerabilidad.

Con eso se pensó en la idea de crear una lista de distribución Bugtraq, donde la principal idea se centra en que las personas que descubrían vulnerabilidades y que también conocían formas para atacar a esas vulnerabilidades se comunicaban directamente al foro.

Lo que ocurre es que por lógica ese foro se ha transformado en una fuente de información para los Hackers, haciendo que muchas empresas dejarán de publicar sus informaciones y han solicitado un tratamiento más responsable de la divulgación de la información. En consecuencia varias empresas han creado su propia política de privacidad de la información formando así algunas organizaciones con enfoques y políticas distintos.

5.4.Divulgación

La divulgación adecuada de las vulnerabilidades de software, tiene un grupo gubernamental que es conocido como Centro de Coordinación CERT/CC. Se trata de una operación de investigación y desarrollo financiada de manera general que se centra en la seguridad de Internet. Fue fundada en 1988 a raíz de un virus de Internet, donde ha evolucionado con al paso de los años y adquirido roles importantes como mantenimiento de estándares industriales para la manera en que las vulnerabilidades tecnológicas se revelan y se comunican.

En el año 2000 se publicó una política que cubre las siguientes áreas:

- La publicación se anunciará dentro de los 45 días siguientes tras la comunicación a CERT/CC.
- El CERT/CC notificará la vulnerabilidad al vendedor del software de manera inmediata, de modo que se pueda desarrollar una solución lo más pronto posible.

- Junto con la descripción del problema, CERT/CC les enviará el nombre del informante, una de las personas que informa de la vulnerabilidad, a menos que esa persona solicite el anonimato.
- Durante el transcurso de 45 días, CERT/CC informará al informante que comunicó la vulnerabilidad en primer lugar sobre el estado de la vulnerabilidad sin revelar información confidencial.

La política que adopta CERT/CC indica que su propósito principal es informar al público de situaciones de amenazas emergentes y al mismo tiempo ofrecerle al propietario del software un tiempo hábil para solventar el problema. Otra información importante es que CERT actualmente trabaja con la guía de la Organización para la Seguridad en Internet (OIS) que trataremos más adelante.

5.4.1. Política de divulgación

La política de divulgación de toda información confidencial, conocida como RFP (Rainforest Puppy Policy), es la más dura con los vendedores de software de CERT/CC. Con esa política se tiene que informar de la vulnerabilidad haciendo un esfuerzo para ponerse en contacto con el vendedor y trabajar juntos para solucionar el problema, pero el hecho de estar cooperando con el vendedor es un paso que el informador no está obligado a hacer, de modo que se considera como un acto de voluntad propia. La política es muy estricta con el vendedor por tema de la confidencialidad.

Puntos importantes de esa política:

- El informador envía un email al vendedor informando del problema, con lo cual la fecha de contacto empieza a contar a partir del envío de ese email.
- Los responsables de mantenimiento de la empresa tendrán cinco días para dar una respuesta al informante. Si no le comunican en el tiempo indicado, el informante es libre para publicar la información del fallo.
- El informante tiene que hacer todo lo posible para ayudar al vendedor a reproducir el problema y cumplir con sus peticiones razonables.
- Al publicar el problema y su solución, se espera que el vendedor recompense al informante por identificar el problema.

- RainForest Puppy un hacker muy conocido que ha descubierto muchas vulnerabilidades en distintos productos. Su trabajo consiste en ayudar a desarrollar parches para los problemas que ha descubierto.

5.4.2. Organización para la Seguridad en Internet

La Organización para la Seguridad en Internet se creó para ayudar a satisfacer las necesidades de todos los grupos y es la política que mejor se adapta a una clasificación de divulgación parcial.

Básicamente existen tres tipos de información de la vulnerabilidad: completa, parcial y sin divulgación. Debido a que se crearon pautas muy estrictas que se contradecían, como las CERT y RFP. Hubo la necesidad de crear la Organización para la seguridad en Internet para intermediar esta situación.

La Organización para la seguridad en Internet es un grupo de investigadores y de fabricantes que se formó con el objetivo de mejorar la forma en la que se gestiona las vulnerabilidades de software.

No se trata de una organización privada que le exige el cumplimento de su política a todo el mundo, lo que intenta es crear un amplio y valorado debate que incluya opiniones respetadas e imparciales para tomarlas como recomendaciones.

Los objetivos para cumplir son:

1. Reducir el riesgo de la aparición de vulnerabilidad desde fuera ofreciendo un mejorado método de identificación e investigación en la solución.
2. Mejorar toda la calidad de ingeniería de software ajustando la seguridad que se aplica el producto final.

5.4.3. Descubrir el fallo en un software.

Ese proceso empieza cuando alguien encuentra algún fallo de seguridad. Una vez encontrado el fallo, se espera que el descubridor realice las siguientes diligencias:

1. Descubrir si ya se ha informado de ese fallo en el pasado.
2. Buscar parches o paquetes de servicio.
3. Averiguar si el fallo afecta a la configuración predeterminada del producto.
4. Asegurar de que el fallo puede reproducirse de manera consistente.

Después de que el descubridor esté seguro de que el fallo existe y tiene toda la información, tiene que elaborar una pauta de informe. OIS ha desarrollado lo que se llama VSR (Informe Breve de Vulnerabilidad). Ese informe incluye la siguiente información:

- La información de contacto del descubridor
- La política de respuesta de seguridad
- El estado del fallo (público o privado)
- Si el informe contiene o no información confidencial
- Los productos que están afectados
- Las configuraciones afectadas
- Descripción del fallo

5.4.4. Notificación

Este proceso consiste en ponerse en contacto con el vendedor. Se trata de una fase muy importante, la comunicación abierta y clara entre el vendedor y descubridor es clave para comprender y buscar la solución para la vulnerabilidad.

Se espera que el vendedor facilite las siguientes informaciones:

- Un único punto de contacto para informes de vulnerabilidad
- La información de contacto deber ser pública al menos en dos ubicaciones accesibles.

La información de contacto debe incluir:

- Referencia de la política de seguridad del vendedor
- Completo listado de todos los métodos de contacto
- Instrucción para realizar comunicación segura.

Facilitar un método de comunicación segura entre sí mismo y el vendedor

Colaborar con el descubridor aún en el caso de que elija utilizar métodos de comunicación inseguros.

Se espera que el descubridor envíe al vendedor cualquier fallo encontrado enviando un VSR a uno de los puntos de contacto publicados.

5.4.5. Validación

La fase de validación implica la revisión del VSR por parte del vendedor, verificar los contenidos y trabajar con el descubridor durante la investigación. OIS facilita algunas reglas generales a seguir relacionadas con las actualizaciones del estado:

- El vendedor debe facilitar información actualizada al descubridor sobre el estado de la investigación al menos cada siete días laborales a menos que ambas partes lleguen a otro tipo de acuerdo.
- Los métodos de comunicación deben ser acordados por ambas partes.
- Si el descubridor no recibe nueva información sobre el estado de la investigación en el plazo de siete días, debe emitir una solicitud de estado.
- En ese caso, el vendedor tiene tres días laborales para responder.

5.4.6. Investigación

El trabajo de investigación que debe realizar el vendedor debe ser minucioso y abarcar todos los productos que están relacionados con la vulnerabilidad. A menudo, el VSR del descubridor no abarca todos los aspectos del fallo y, la última instancia, es responsabilidad del vendedor investigar todas las áreas que están afectadas por el problema en sí.

Los pasos de la investigación son los siguientes:

1. Investigar el fallo de producto descrito en el VSR.
2. Investigar si el fallo también existe en los productos compatibles que no están incluidos en el VSR. Investigar los vectores de ataque de la vulnerabilidad

3. Llevar una lista pública de qué productos o versiones son compatibles con el software.

5.4.7. Descubrimientos

Finalizada la investigación la empresa tiene que enviar algunas conclusiones al descubridor del fallo:

1. Que se ha confirmado el fallo
2. Que se ha desmentido el fallo que fue notificado
3. Que no se puede probar ni desmentir el fallo

La empresa no está obligada a informar los detalles de las pruebas o bien de los procedimientos, pero tiene que demostrar que se realizó una investigación estricta y totalmente técnica, lo que se realiza facilitándole al descubridor:

4. Lista de los productos o versiones que han sido actualizadas.
5. Lista de las pruebas que fueron realizadas
6. Los resultados de las pruebas.

5.4.8. Confirmar el fallo

Cuando la empresa confirme el fallo tiene que incluir las siguientes informaciones en la confirmación:

1. Lista de los productos o versiones afectadas por el fallo que ha sido confirmado.
2. Declaración de cómo se distribuirá el parche.
3. Programar el tiempo necesario para publicar el parche.

Resolución

Cuando se confirma el fallo la empresa tiene que seguir algunos pasos adecuados para desarrollar una solución viable que arregle el problema. OIS indica los siguientes pasos cuando esté creando la solución de una vulnerabilidad:

1. La empresa determina si ya existe una solución para la vulnerabilidad. Si existe entonces se tiene que informar al descubridor de manera

inmediata. Si no existe, entonces tiene que empezar a desarrollar una solución de manera también inmediata.

2. La empresa tiene que asegurar que la solución encontrada para su producto esté disponible para todas las versiones existentes y también productos compatibles.
3. La empresa puede compartir información con el descubridor si eso ayuda en la eficacia de la solución de la vulnerabilidad. No hace falta que el descubridor tenga que participar en todo el proceso.

5.4.9. Marco temporal

Determinar un tiempo para la solución es importante, debido al riesgo al que está expuesta la aplicación. Aunque ese tiempo es importante, también tiene la misma importación el hecho de asegurar una adecuada solución para el problema.

Se espera que la empresa tenga una solución en un plazo de 30 días desde que recibe la notificación del VSR. La solución del problema tiene que solucionarlo y no debe crear ningún fallo adicional. No siempre sigue de manera estricta el tiempo de 30 días, esto es debido a que la documentación de OIS indica varios factores que se deben tener en cuenta al tomar la decisión de la fecha de publicación del parche.

Los tres tipos de soluciones que implican cambios en los software son:

- **Parches:** Cambios temporales que implican solucionar problemas específicos hasta que la futura versión pueda solventar el problema.

- **Actualizaciones:** Son versiones programadas para solucionar fallos conocidos. A menudo algunos distribuidores hacen referencia a estas actualizaciones como Service Packs (versiones de mantenimiento)

- **Versiones Futuras:** Son también versiones programadas, pero se diferencian de las actualizaciones en que se realizan grandes cambios que afectan el diseño del producto y sus características.

5.4.10. Publicación

Es el último paso de la VSR (Política de Informes de Vulnerabilidad de Seguridad) donde trabaja OIS para publicación de la información vulnerable. Se asume que la publicación de la información tiene que ser para todo el público a la vez, es decir no se debe adelantar la información a grupo específicos.

5.4.11. Razones por la que se deben publicar las vulnerabilidades

1. Los hackers no éticos ya conocen las vulnerabilidades de todas formas, ¿por qué no dárselas a conocer a los hackers éticos?
2. Si los hackers no éticos conocen la vulnerabilidad, tarde o temprano la descubrirán sin que haya una publicación oficial.
3. Conocer detalles ayuda más a los hackers éticos que a los hackers no éticos.
4. La seguridad efectiva no se puede basar en secretismos.
5. Hacer públicas las vulnerabilidades es una herramienta efectiva para conseguir que los vendedores mejoren sus productos.

5.4.12. Lenguaje de Vulnerabilidad y Evaluación

Se trata de un estándar internacional de seguridad de la información abierto, que tiene como mayor objetivo publicar contenidos seguridad y normalizar la transferencia en conjunto con las herramientas y servicios de seguridad.

6. BUENAS PRÁTICAS Y METODOLOGIAS

Puntos importantes que tiene que seguir la empresa que se dedica al hacking ético.

6.1. Aspectos legales

- Asegurarse de haber firmado con una empresa un acto de no divulgación de la información.
- Agregar el punto de no divulgación en los apéndices del documento.
- Asegurar que han firmado un acuerdo de evaluación de seguridad.
- Escaneo de direcciones IP solamente las que están previstas en el contrato.
- Definir claramente los límites de la evaluación para evitar cualquier conflicto.

6.2. Personal

- Equipo técnico que participará en el proceso de evaluación. Las siguientes informaciones deben ser documentadas y evaluadas por la empresa.
- Experiencia con las plataformas, aplicaciones, protocolos de red y dispositivos de hardware.
- Certificaciones y cursos relacionados con Pentesting.
- Años de experiencia en Pentesting.
- Attack scripting/lenguaje de programación por cada miembro
- Información pública que demuestra participación en la comunidad de cada miembro, como artículos, foro, mensajes, participación en eventos etc.
- Los empleados de la empresa que participan en el proceso tienen que haber firmado un acuerdo de confidencialidad.

6.3. Procesos

- Dejar claro que tipo de pruebas se realizaran o indicar que será solamente una auditoria describiendo los fallos y problemas.
- Evaluar la seguridad de un sistema primario o secundario. Ambos métodos tienen sus ventajas y desventaja, pero en general se recomienda evaluar la seguridad de los servidores secundarios en lugar de los primarios.
- Asegurarse de que la infraestructura es segura.
- Asegurarse que el evaluador realizará las pruebas en equipos oficiales, si no es así, asegurarse de que lo notifique.

- Asegurar que el equipo técnico proporciona información precisa sobre el hardware que será evaluado y también localización lógica.
- Definir fecha, hora y día para la evaluación.
- La empresa de evaluación debe tener definido todo el proceso para la gestión de los tests.
- Asegurar que tanto la empresa que evalúa como la que será evaluada puedan intercambiar información acerca del personal técnico.

6.4. Entregables

- El equipo de evaluación debe mostrar un claro enfoque sobre sus metas y la ruta del ataque.
- La empresa evaluadora debe enseñar una copia de los informes de evaluación anterior. Asegurar no revelar ninguna información del cliente y siempre esconder informaciones importantes como números de IPS.
- Asegurar no tener ningún protocolo/servicios bloqueados.
- Asegurar que la empresa evaluadora no cambie la IP sin su permiso.
- Asegurar tener disponible kit de evaluación para el personal técnico.
- Asegurar que el equipo técnico comprenda los requisitos del cliente.
- Asegurar que esté utilizando un equipo para test.
- Asegurar que el proceso esté disponible para la recogida de resultados de las pruebas y que se presentan en un formato adecuado.
- Asegurar que todo el proceso de prueba esta supervisado y debidamente documentado, con el fin de facilitar la identificación de las comunicaciones, problemas y falsos positivos
- Evitar brechas en la confidencialidad mediante liberación de datos del cliente.
- Asegurar que el servidor de almacenamiento para los resultados de la prueba está seguro y bien protegido.
- Asegurar tráfico seguro de la información.

6.5. OTP (Proyecto de Pruebas OWASP)

Se trata de una de las principales metodologías para realizar testes en aplicaciones Web. Tiene como objetivo ayudar a las organizaciones a construir un proceso completo de pruebas. Se trata de una solución flexible que puede ser extendida y amoldada para encajar en el proceso de desarrollo y cultura de una empresa.

Está estructurado con cinco fases:

Fase 1. Antes de empezar el desarrollo:

- Antes de empezar se realiza una planificación del trabajo y medición. Se define los criterios que deben ser medidos. Es importante definir las métricas antes de empezar el desarrollo del proyecto.

Fase 2. Durante el diseño y definición:

- Se revisan los requisitos de seguridad. Es indispensable que los requisitos de seguridad sean aprobados.

Fase 3. Durante el desarrollo:

- Durante la fase del desarrollo se realiza la implementación del diseño, con eso el desarrollador deberá afrontar decisiones.
- Durante esa fase tendrá la inspección del código por fases donde el equipo de seguridad deberá realizar la inspección por fases en conjunto con los desarrolladores.

Fase 4. Durante la implementación:

- Una vez finalizado la fase de los requisitos y analizado el diseño, se debe asumir que se han identificado todas las incidencias.
- En esta fase se tiene la comprobación de gestión de configuraciones. Aunque la aplicación pueda estar segura, siempre puede haber algún fallo en la configuración y podría dejar esta misma, vulnerable a explotación con lo cual debemos revisar las configuraciones.

Fase 5. Mantenimiento y operaciones:

- Ejecución de comprobaciones periódicas de mantenimiento. Se realiza comprobaciones de mantenimiento mensual e trimestral, sobre la aplicación e infraestructura con el objetivo de asegurar que no se han introducido ningún riesgo a la seguridad.
- Comprobar los cambios realizados también es una tarea que se realiza en esa fase. Es importante que una vez finalizado todos los cambios asegurar que el nivel de seguridad no haya sido afectado por dicho cambio.

6.6. OSSTMM

OSSTMM es un framework desarrollado por ISECOM (Instituto para la Seguridad y Metodologías Abiertas), para crear una metodología estándar que permita evaluar qué nivel de seguridad dispone la empresa evaluada. Para realizar han creado cinco secciones durante todo el proceso que se evalúa:

- Controles aplicados sobre los datos
- Niveles de concienciación de seguridad del personal
- Niveles de control sobre la ingeniería del software y el fraude
- Seguridad en sistemas y redes de comunicaciones
- Dispositivos inalámbricos
- Dispositivos móviles
- Seguridad de los controles de acceso físico
- Procesos de seguridad
- Ubicaciones físicas
- Perímetros

La metodología está centrada en todos los detalles técnicos de lo que se debe realizar previamente, durante y después de un teste de seguridad. Esa metodología es evaluada continuamente para saber lo que se puede mejorar en las prácticas, leyes y regulaciones.

El mapa de la seguridad es una representación visual de la presencia de seguridad. Esa presencia de seguridad corresponde con el entorno de la prueba y

está compuesta por seis secciones. Las pruebas adecuadas de cualquier sección se deben incluir los elementos de todas las otras secciones, sean directas o indirectas.

1. Información de la Seguridad (Information Security)
2. Proceso de la Seguridad (Process Security)
3. Internet Security Technology (Internet Technology Security)
4. Seguridad en las Comunicaciones (Communications Security)
5. Seguridad inalámbrica. (Wireless Security)
6. Seguridad Física (Physical Security)

Para la realización de la prueba de seguridad con OSSTMM es importante tratar cada sección en particular. Todas las secciones deben ser probadas y de lo que la infraestructura no existe o no puede ser verificada, se determina como no aplicable y deberá ser informada.

6.6.1. Detalles de las pruebas por secciones

- **Pruebas de seguridad de la información.**
    1. Valoración inicial.
    2. Repaso de la integridad de la información.
    3. Encuesta de Inteligencia
    4. Recolecta de información.
    5. Análisis con Recursos Humanos.
- **Prueba de seguridad del proceso**
    1. Revisión
    2. Solicitar pruebas
    3. Revisión de las pruebas solicitas
    4. Estructurar las pruebas solicitadas
    5. Pruebas en personas de confianza
- **Pruebas de seguridad fisica**
    1. Revisión
    2. Pruebas de los controles de acceso
    3. Revisión del perímetro comentario
    4. Revisión del monitoreo

5. Revisión las respuestas de alarma
6. Revisión de la localidad
7. Revisión del medio ambiente

- **Pruebas de seguridad de las comunicaciones**

1. Revisión
2. Revisión PBX
3. Teste en Voicemail
4. Teste en FAX
5. Análisis del Modem
6. Teste Remote Access Control
7. Teste en Voice over IP
8. Teste en X.25 Packet Switched Networks

- **Pruebas de seguridad tecnológica en Internet**

1. Logística y Controles.
2. Revisión.
3. Revisar Intrusión y Detección.
4. Topografía de red.
5. Servicios de la identificación de sistema.
6. Exploración de Inteligencia Competitiva.
7. Revisión de privacidad.
8. Documentación.
9. Teste de aplicaciones de Internet.
10. Explotar Investigación y Verificación.
11. Enrutamiento.
12. Teste de control de acceso.
13. Password Cracking.
14. Teste de medidas de contención.
15. Teste de Denegación de servicio.
16. Revisiones políticas de seguridad.
17. Revisión de alertas y logs.

- **Pruebas de seguridad inalámbrica**

1. Revisión
2. Teste en Electromagnetic Radiation (EMR).
3. Teste en redes 802.11 Wireless.
4. Teste en redes Bluetooth*
5. Teste en Dispositivos de entrada inalámbrico
6. Teste en terminales inalámbrico
7. Teste de dispositivos RFID
8. Teste de dispositivos infrarrojos

6.7. ISSAF (Marco de Evaluación de la Seguridad del Sistema de Información)

ISSAF también es una metodología estructurada de análisis de seguridad en varios dominios, con lo cual se realiza test específicos en cada una de sus fases de análisis. El principal objetivo es proporcionar procedimientos con detalles para los test que se realizan en sistemas de información. Toda la información contenida dentro de ISSAF forma un conjunto que se puede llamar "***Criterios de Evaluación***".

Se utiliza para cumplir con varios requisitos de evaluación y normas de seguridad, además se puede utilizar como referencia para nuevas implementaciones relacionadas con la seguridad. Está muy bien estructurado y ha sido revisado por profesionales expertos en seguridad de red y pasan por revisiones periódicas.

Los criterios de evaluación que trabaja esa metodología son:

- Una descripción de los resultados de evaluación
- Finalidades y objetivos
- Los prerrequisitos para la realización de las evaluaciones
- Los procesos para las evaluaciones
- Presentaciones de resultados
- Contramedidas recomendadas.
- Referencia a documentos externos

En estas fases tenemos muchos procesos que son:

- Penetración
- Obteniendo Acceso
- Escala de Privilegios
- Mantenimiento del acceso
- Cubrimiento de huellas
- Riesgos inherentes
- Regulaciones legales
- Políticas de seguridad
- Evaluaciones
- Mapeo de redes
- Informes (Reportes)
- Identificación de Vulnerabilidades
- Recolección de Información
- Identificación de Recursos

7. INFORMES

Después de realizar toda la evaluación y haber encontrado una lista de vulnerabilidades, comienza la parte más crítica del proceso. El trabajo realizado no tiene ningún valor si no se tiene un informe bien escrito y se reúne con el cliente para concienciar de la importancia de la seguridad. En la parte final de la evaluación, el jefe de equipo debería trabajar en el informe recibiendo datos del líder técnico durante todo el proceso. El responsable del equipo tiene que redactar un informe relacionado con los activos o con las vulnerabilidades para entregárselos al cliente en el final del proceso.

A menudo se trata de redactar un resumen completo de gestión y de las técnicas empleadas para la evaluación, pero debería proporcionarle un informe técnico al cliente, de modo que podrá comenzar a resolver los problemas de seguridad.

A continuación, se muestra la información expuesta en la sección de información general de una red:

1. Dirección de la red auditada.
2. Número total de equipos de la red que han sido analizados.
3. Número total de servicios abiertos de todos los equipos analizados.
4. Número total de intrusiones ejecutadas a los equipos de la red.
5. Número total de contraseñas interceptadas. Es la suma de las contraseñas Web, las contraseñas de servicios de cada equipo y las contraseñas de Windows de cada equipo.
6. Tabla con las contraseñas Web interceptadas.
7. Tabla con las comunicaciones cifradas interceptadas que no han sido encriptadas.

8. RECOMENDACIONES PARA LA IMPLANTACIÓN DE MEDIDAS DE SEGURIDAD

La estrategia de gestión de la seguridad informática para los sistemas de control se basa en tres aspectos principales:

- Formación al personal de operaciones y de control de procesos para que mejoren sus conocimientos sobre la tecnología y los problemas asociados a la seguridad informática.
- Formación al personal de tecnología de la información para que tengan un mayor conocimiento de los procesos de producción y la tecnología asociada, además de métodos y procesos relacionados con la gestión de seguridad de los procesos (PSM).
- Desarrollo de procedimientos que reúnan las habilidades y conocimientos del control y la informática.

Mediante esos tres aspectos lo que se pretende es desarrollar barreras de defensa que disminuyan la probabilidad de incidencias y que faciliten una rápida recuperación de los sistemas de control.

Esas barreras de defensa se resumen en la siguiente tabla y explican sobre qué elementos del sistema de gestión de la seguridad se deben trabajar.

BARRERAS DE DEFENSA	
Barreras	**Actuaciones**
Humanas	Políticas, formación
Aplicaciones	Sistema de control, bases de datos
Sistemas operativos	Gestión de parches, configuración
Redes	Cortafuegos, sistemas de detección
Seguridad física	Guardias, puertas, vigilancia

Figura 3.- Barreras de defensa

Los pasos a seguir son:

1. Identificar los riesgos y establecer prioridades de actuaciones.
2. Poner en marcha los elementos de organización para encarar el riesgo.
3. Identificar las medidas de seguridad y los elementos fundamentales aplicables al menos a otra generación de sistemas y equipos de control.
4. Implantar la estrategia.
5. Seguimiento y mejora de la estrategia.

En dISA-99.00.02 (Versión 1 preliminar, Edición 5) se detallan un conjunto de diecinueve actividades a seguir para la implantación de un programa de seguridad informática para los sistemas de control procesos y automatización de la producción recopilados en la siguiente tabla.

BARRERAS DE DEFENSA	
Actividad	**Descripción**
Actividad 1	Business Case
Actividad 2	Obtener compromiso, soporte y financiación de la Dirección
Actividad 3	Definir Objetivos y Alcance de la seguridad en SC&F de la empresa
Actividad 4	Constituir un equipo de interesados (seguimiento y aprobación)
Actividad 5	Aumentar la habilidades en seguridad a través de formación a los empleados
Actividad 6	Caracterizar los riesgos claves de los sistemas de control y fabricación
Actividad 7	Priorizar y calibrar los riesgos
Actividad 8	Establecer políticas de seguridad a alto nivel que admite el nivel la tolerancia al riesgo
Actividad 9	Establecer la estructura organizacional responsable de la seguridad
Actividad 10	Inventariar las redes y dispositivos de SC&F
Actividad 11	Exploración y Priorización de los sistemas de control y fabricación
Actividad 12	Realizar una evaluación detallada de la seguridad
Actividad 13	Desarrollar políticas y procedimientos detallados de seguridad informática en SC&F
Actividad 14	Definir el conjunto de estándares del control de mitigación de riesgos de seguridad en SF&C
Actividad 15	Desarrollar elementos adicionales al plan de gestión de la seguridad informática
Actividad 16	Implantar las soluciones rápidas
Actividad 17	Diseñar y ejecutar proyectos de mitigación de los riesgos de seguridad informática: - Definir objetivos y alcance - Diseñar el proyecto - Ejecutar el proyecto - Decidir la planificación de cuando hacer validaciones del programa - Validación
Actividad 18	Refinar e implantar el sistema de gestión de la seguridad informática de la empresa/unidad de negocio
Actividad 19	Adoptar medidas operacionales para la mejora continua

Figura 4.- Barreras de defensa. Descripción

La implantación de un programa de seguridad no garantiza al 100% que ocurra algún incidente de seguridad, pero sí minimizan de manera significativa la probabilidad de que ocurra y su potencial impacto.

Mediante la implantación del programa de seguridad se logra prevenir el acceso de intrusos a los sistemas de control mediante la instalación y mantenimiento de cortafuegos y enrutadores de tráfico, además de antivirus. En el caso de que la intrusión ocurra, el programa deberá incluir la dotación de herramientas de detección de intrusos y del registro de esos ataques con la finalidad de proceder a su bloqueo, realizar seguimientos, y promover las denuncias necesarias ante los cuerpos de seguridad del Estado especializados en delitos informáticos. El programa de seguridad tiene que incluir medidas de mitigación como son los protocolos y procedimientos para instalación de parches de seguridad de los sistemas y actualización del software, así como copias de respaldo y recuperación de los sistemas.

Las acciones del programa de seguridad se pueden enmarcar en cinco categorías principales:

- Tecnologías de autenticación y autorización.
  - ✓ Herramientas de autorización basada en roles.
  - ✓ Autentificación de password.
  - ✓ Autentificación por desafío (reto/respuesta).
  - ✓ Autentificación física/token.
  - ✓ Autentificación tarjeta Smart.
  - ✓ Autentificación biométrica.
  - ✓ Autentificación basada en la localización.
- Tecnologías de filtrado/bloqueo y control de acceso.
  - ✓ Cortafuegos de red
  - ✓ Cortafuegos basados en host (workstation/Controladores)
  - ✓ Redes de área local virtuales (VLAN)
- Tecnologías de encriptación y validación de datos.
  - ✓ Encriptado de clave simétrica (secreta)
  - ✓ Encriptado de clave pública y distribución de claves
  - ✓ Redes privadas virtuales (VPN)

- Herramientas de gestión: auditoria, medición, monitorización y detección.
    - ✓ Utilidades para auditar registro de acciones
    - ✓ Sistemas de detección de virus y código malicioso
    - ✓ Sistema de detección de intrusión
    - ✓ Escáner de vulnerabilidad
    - ✓ Utilidades de análisis forense
    - ✓ Herramientas de configuración del host (workstation/Controladores)
    - ✓ Herramientas para la automatización de la gestión del software.
- Control físico de la seguridad.
    - ✓ Protección física
    - ✓ Seguridad personal

9. ATAQUE A RED

9.1. HACKEO DE RED CON WIFISLAX

Una vez en el radio de Red Wifi que se quiere conseguir, en este caso, la red de laboratorio CINCUBATOR.

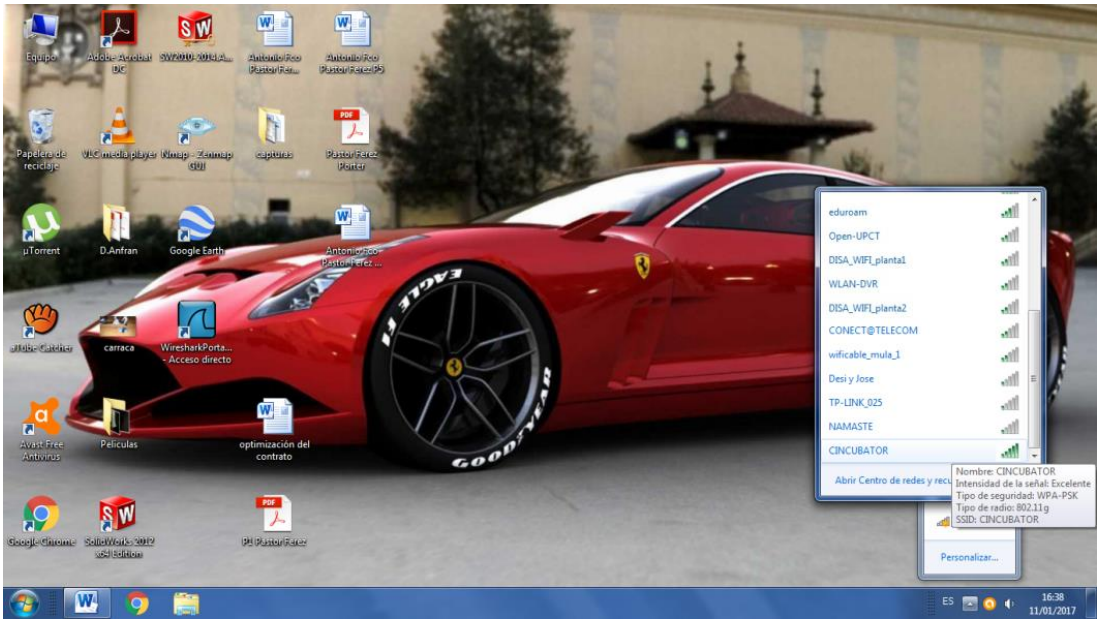

Figura 5.- Red INCUBATOR

1. Se abre el programa de Wifislax con el portátil y empieza la parte práctica del proyecto:

   Se selecciona arrancar con kernel SMP que es el recomendado y se deja que cargue el programa.

Figura 6.- Kernel

Después de unos segundos se tiene el programa iniciado y en espera en la ventana principal.

Figura 7.- Programa a cargar

2. En la ventana principal se selecciona en la parte inferior izquierda de la pantalla el icono de programas y seleccionamos la aplicación wifislax > wpa > Handshaker, y se hace clic en ésta como muestra la imagen.

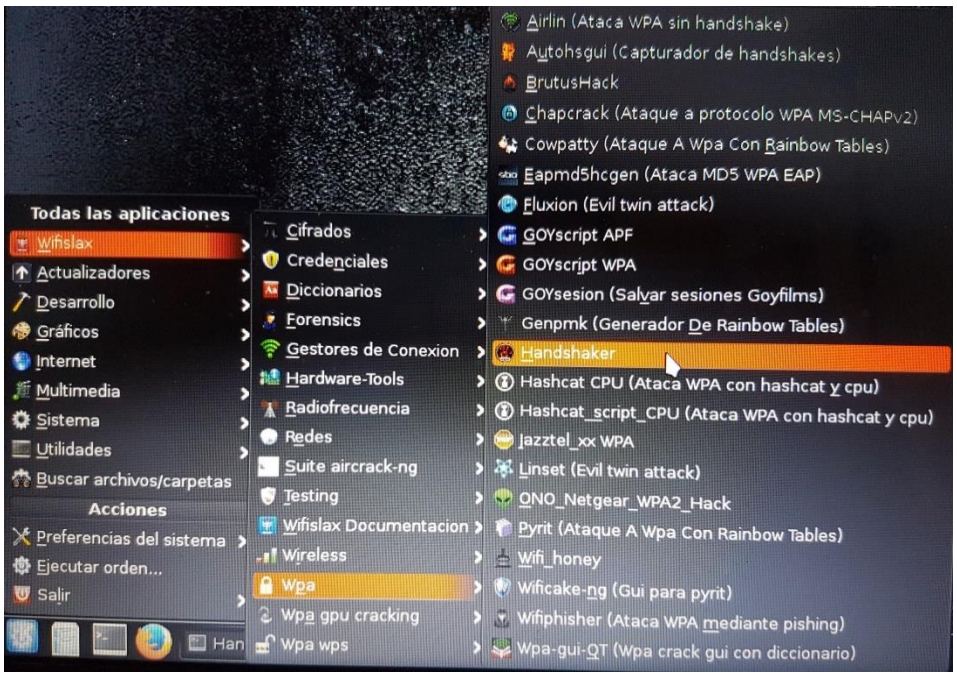

Figura 8.- Ubicación del programa Hanshaker

3. Una vez abierto el Handshaker se realiza la selección de la antena wifi para realizar el escáner de redes y cuando acabe apretamos Ctrl + C para pasar a la siguiente etapa.

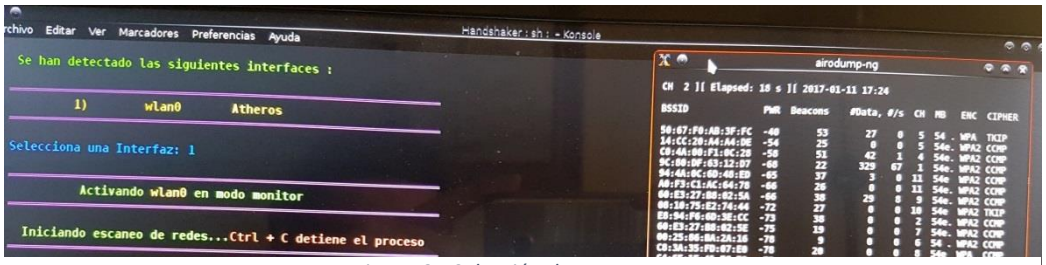

Figura 9.- Selección de antena WIFI

4. Ahora se selecciona la red que vamos a atacar, en nuestro caso la red del laboratorio de robótica CINCUBATOR, se presiona la tecla 1 que corresponde a dicha red y le se le da Enter.

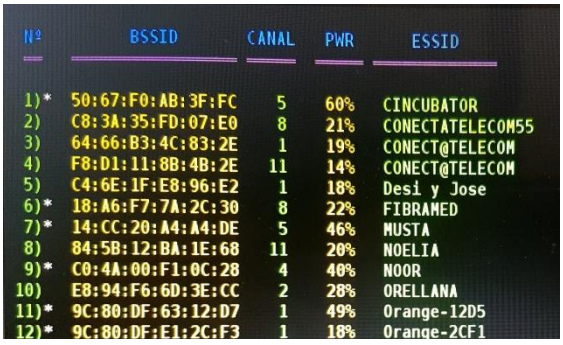

Figura 10.- Selección de red.

5. Después de seleccionar la red se deberá elegir el tipo de ataque que se desea efectuar, como no se sabe si se logrará el objetivo con el ataque Honeypot o Aireplay se efectuará la prueba con ambos para tardar menos en conseguir el objetivo, se pulsa 4 y Enter.

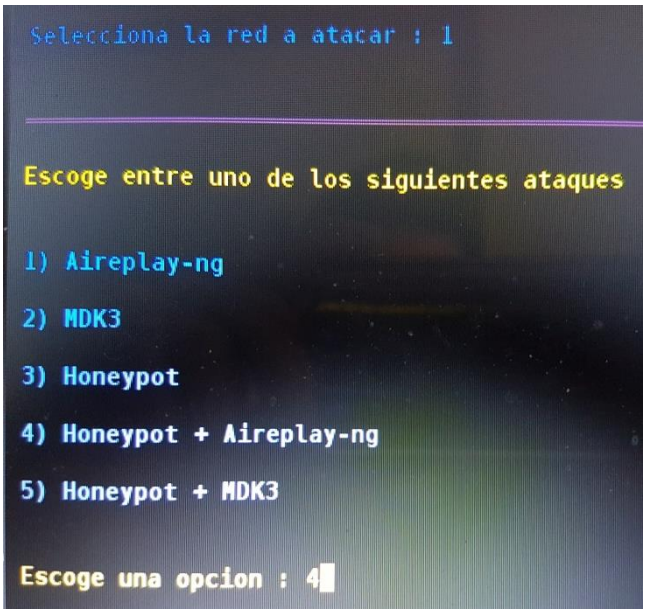

Figura 11.- Selección del tipo de ataque.

6. En pocos segundos se obtiene el pin en un tipo de archivo .cap que se usará en otro programa.

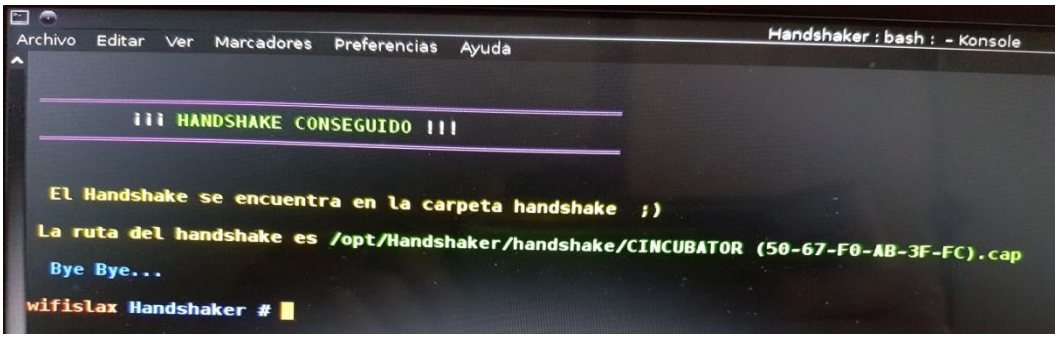

Figura 12.- Obtención del pin archivo .cap

7. Se busca la dirección del archivo en la carpeta opt/Handshaker, se copia y se pega en la carpeta de opt/Brutus.

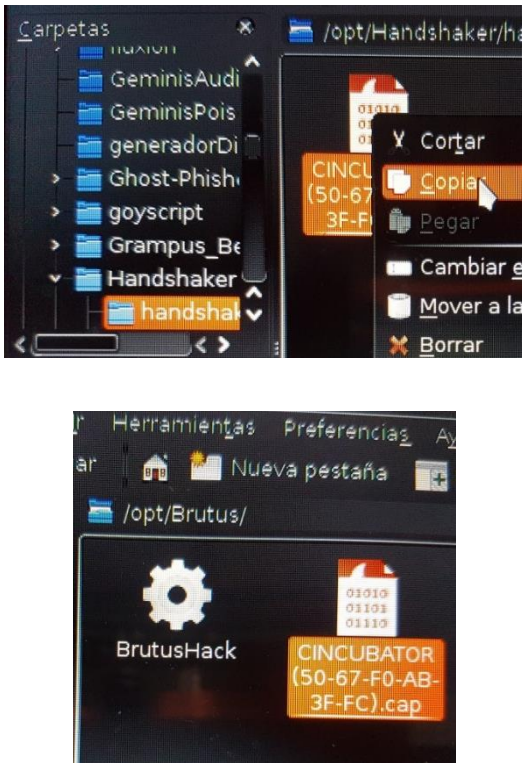

Figura 13.- Copia y pega de dirección del archivo opt/Handshaker

8. Se abre el programa de la lista wifislax > wpa > BrutusHack donde se pega el archivo del Handshaker y se inicia.

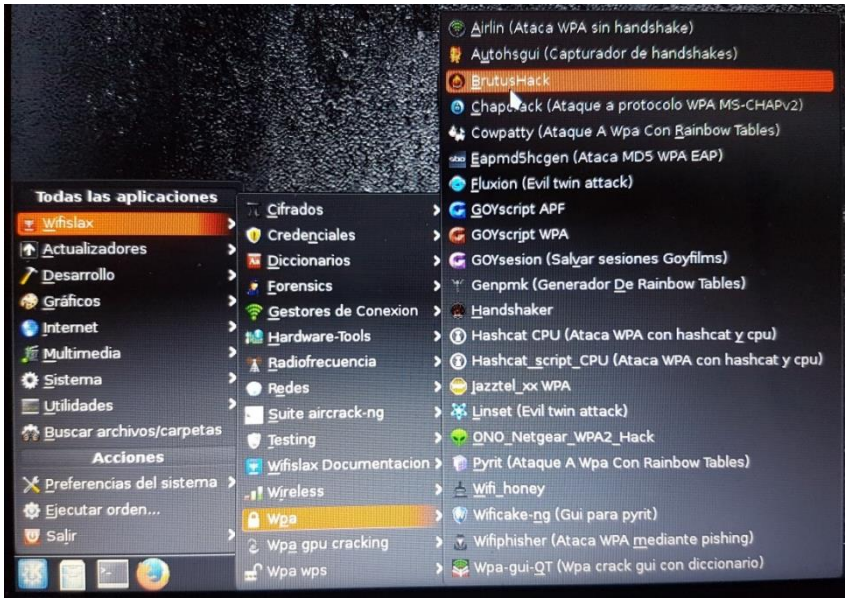

Figura 14.-BrutusHack

9. Nada más iniciar el programa, se pondrá a buscar la clave númerica de la red CINCUBATOR, este proceso suele tardar varias horas.

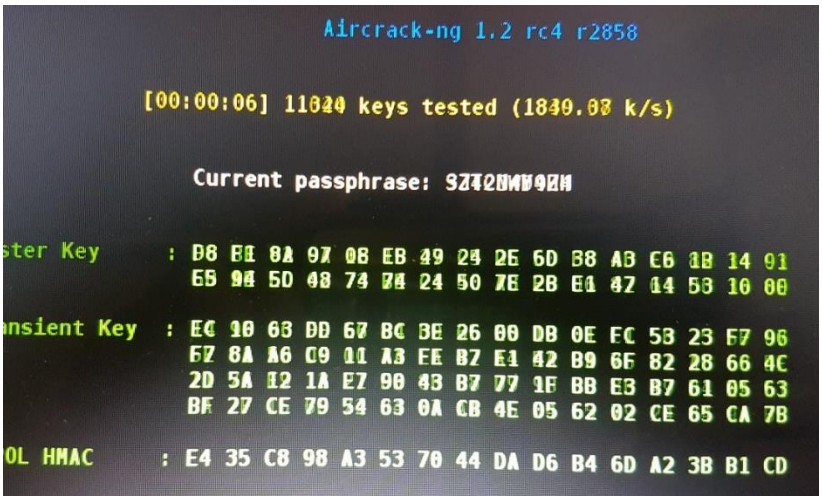

Figura 15.- Clave numérica de la red INCUBATOR

Una vez conseguida, se guarda en un archivo de texto a la carpeta de opt/Brutus, ya solo se cerrará Wifislax e iniciar windows y seguidamente insertar la clave wifi en la red.

10. Como se puede observar se ha insertado la clave de red CINCUBATOR y se puede utilizar la red local del labortorio desde donde se realizará la siguiente fase del proyecto, que consiste en el sondeo de direcciones IP.

Figura 16.- Red INCUBATOR

9.2. PRUEBAS DE SONDEO CON Nmap

En esta prueba se hará el uso de la herramienta Nmap para realizar pruebas de escaneo de puertos de la red local del laboratorio.

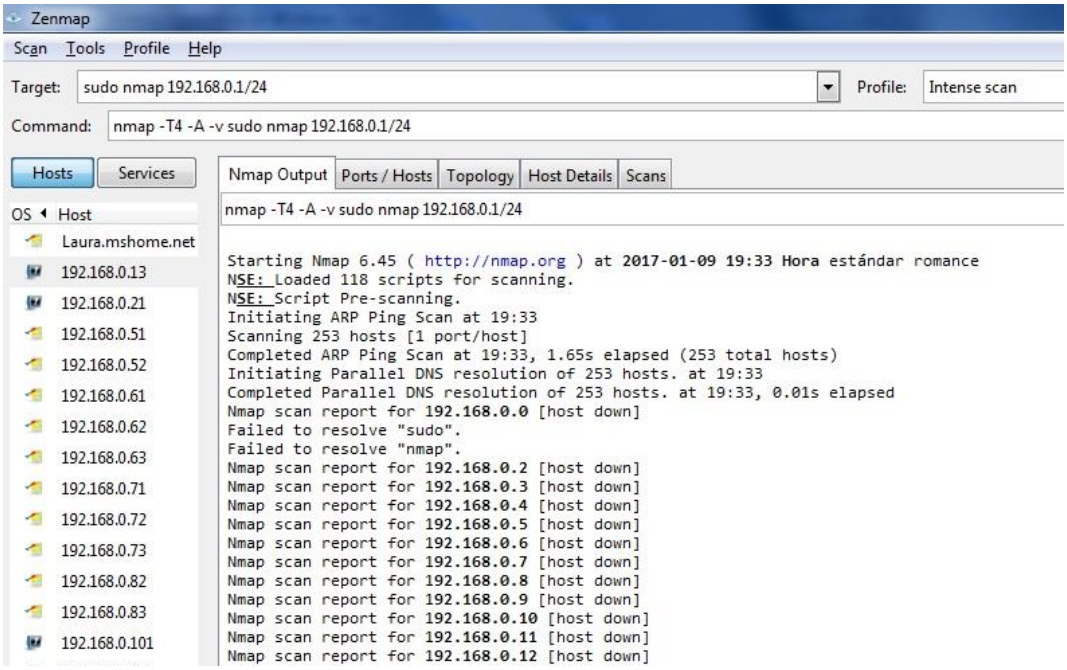

Figura 17.- Nmap

Con la opción sudo nmap se sondea todos los puertos desde 0 a 255, para poder saber que IPs son las que se deben atacar.

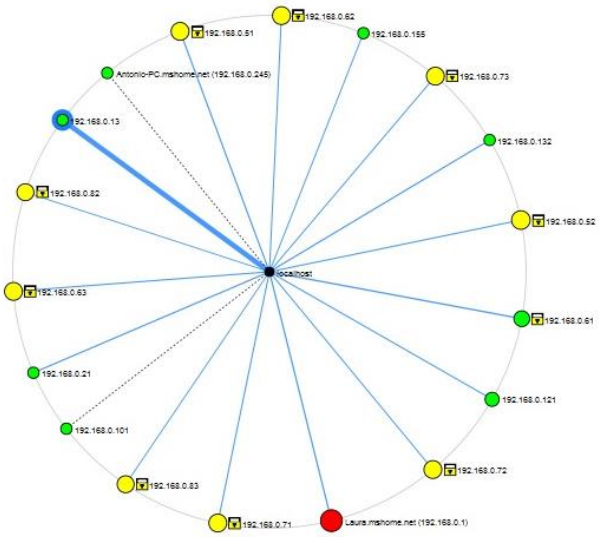

Figura 18.- Sondeo de los puertos

Las IPs que se tienen en cuenta, en esta parte del proyecto son las que contengan información del software que contenga licencia de Siemens.

Una vez terminado el escaneo se puede buscar en la topología de redes las IPs activas en dicha red para saber, de las 255 que se sondean, cuales están en activo.

Como se ve a continuación la dirección 192.168.0.13 y la 21 tienen licencia de Siemens.

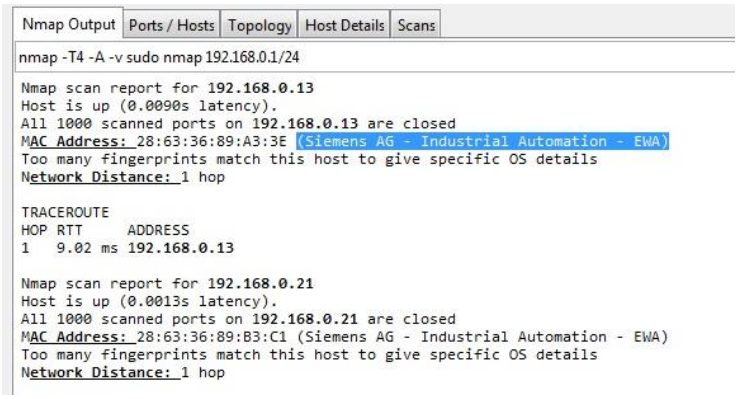

Figura 19.- Direccion IP con licencia Siemens

Al llegar a este punto se procede a atacar la dirección seleccionada desde el software de Tia portal V13.

Con Nmap se pueden realizar tareas muy complejas que están fuera de este proyecto, es este proyecto se ha utilizado esta herramienta con el fin de detectar los paquetes de entrada y salida en una red wifi donde hay ordenadores junto a PLCs para localizar los puertos y atacarlos de manera ética.

9.3. TIA PORTAL V13

Una vez se ha obtenido la dirección que se quiere atacar y se sabe el tipo de PLC solo hace falta abrir el software de Tia Portal V13 y seguir estos pasos.

1. Primero se abre el programa y se crea un nuevo proyecto y se observa la siguiente ventana.

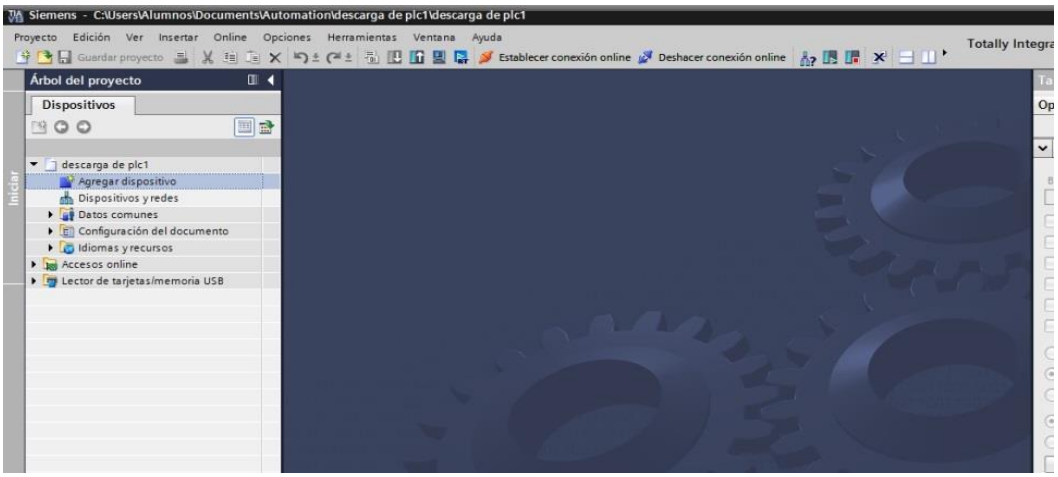

Figura 20.- Tia Portal

2. Se selecciona, en la ventana de la izquierda, el dispositivo tipo Simatic S7 1200 y el CPU 1200. Se desconoce el modelo de CPU.

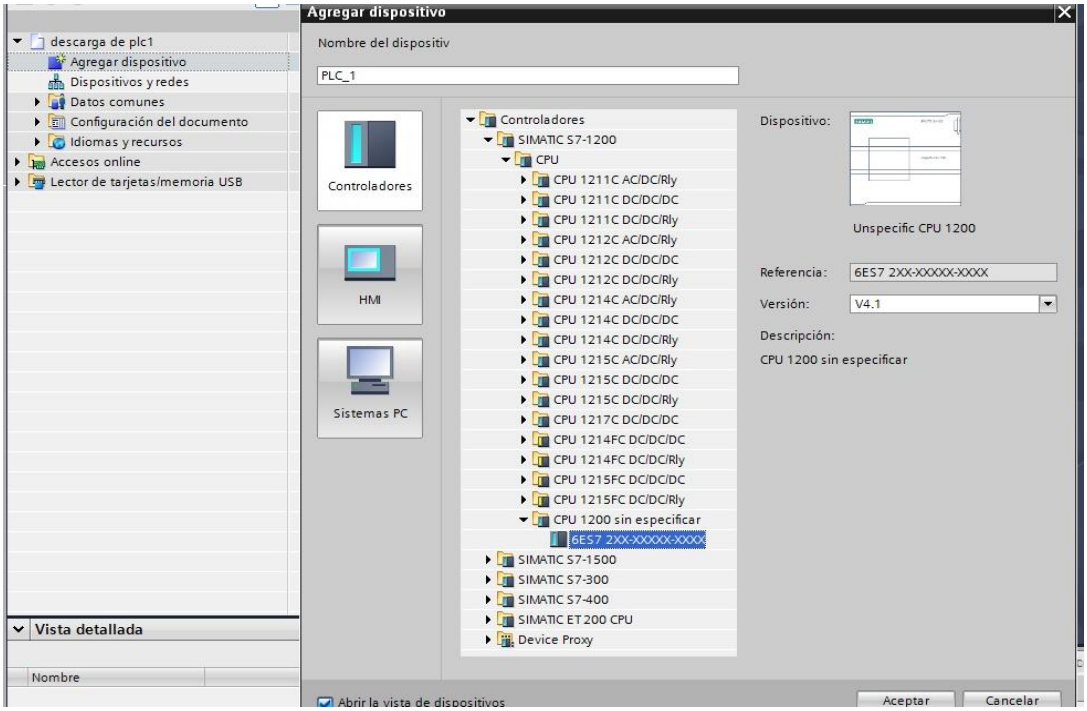

Figura 21.- Selección de dispositivo

3. A continuación, se observa una nueva ventana donde se visualiza el modelo seleccionado anteriormente, S7 1200, sin ningún tipo asignado. Se hace clic encima de la palabra determinar en el bocadillo y se cargará otra ventana.

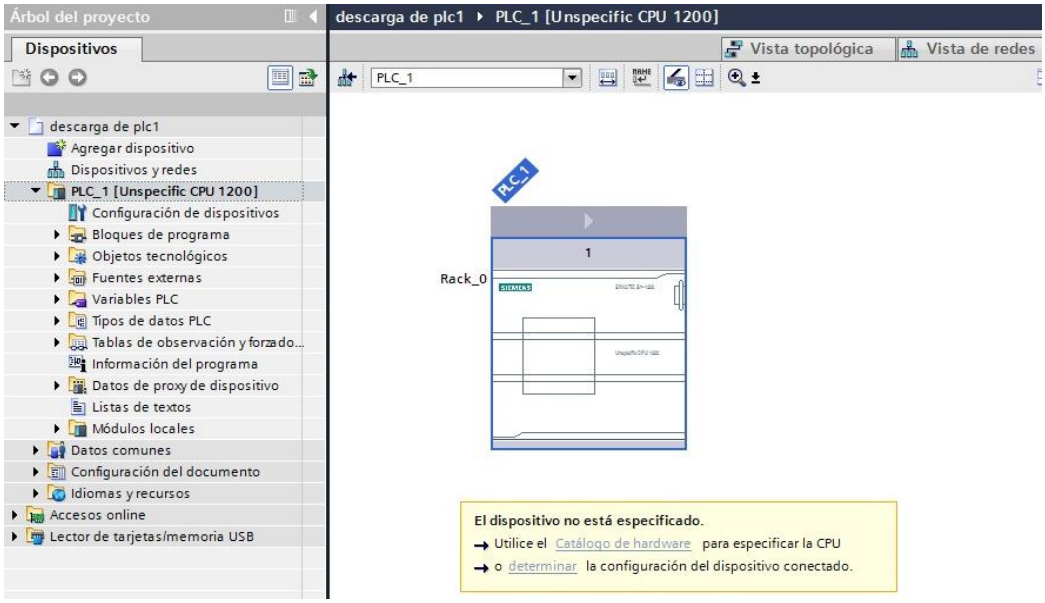

Figura 22.- Dispositivo Siemens seleccionado

4. Una vez cargada la ventana que vemos a continuación, se seleccionará el tipo de interfaz: PN/IE y el interfaz: Adaptador D-Link. Se acepta la operación y se podrá ver las IPs disponibles en red, estas coinciden con los escaneados desde Nmap. Se utiliza la siguiente IP: 192.168.0.13 ya que esta se atribuye a un dispositivo Siemens, Se realiza la deteccion.

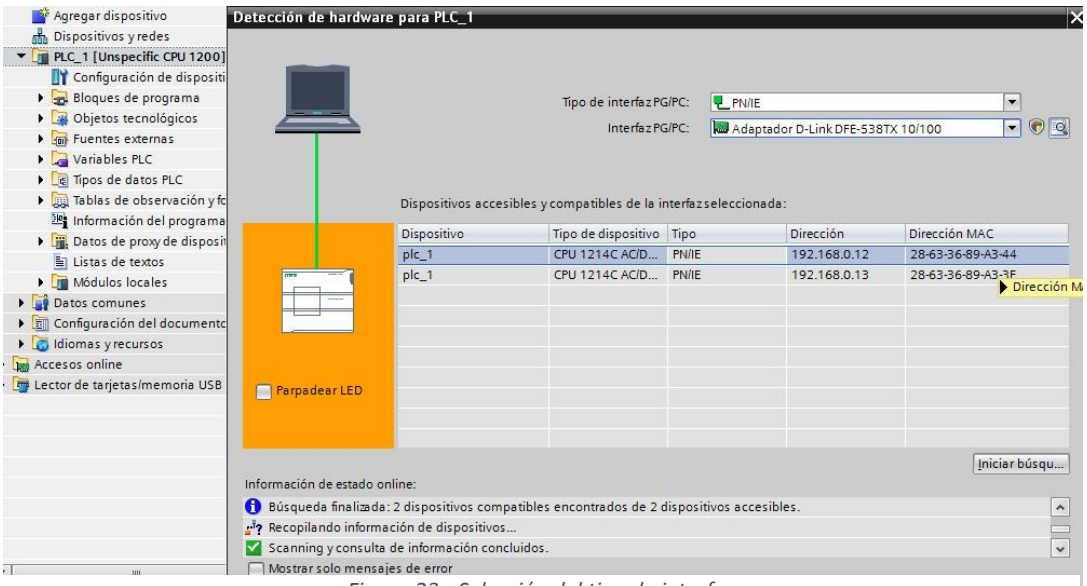

Figura 23.- Selección del tipo de interfaz

5. El enlace dependerá de las conexiones asignadas a la IP, si esta corresponde a una pantalla HMI o a un PC, no se realizará el enlace y habrá que realizar la operación nuevamente con otra dirección. En caso de que la conexión sea con S7 1200 aparecerá la pantalla de la siguiente imagen y se podrá realizar el enlace.

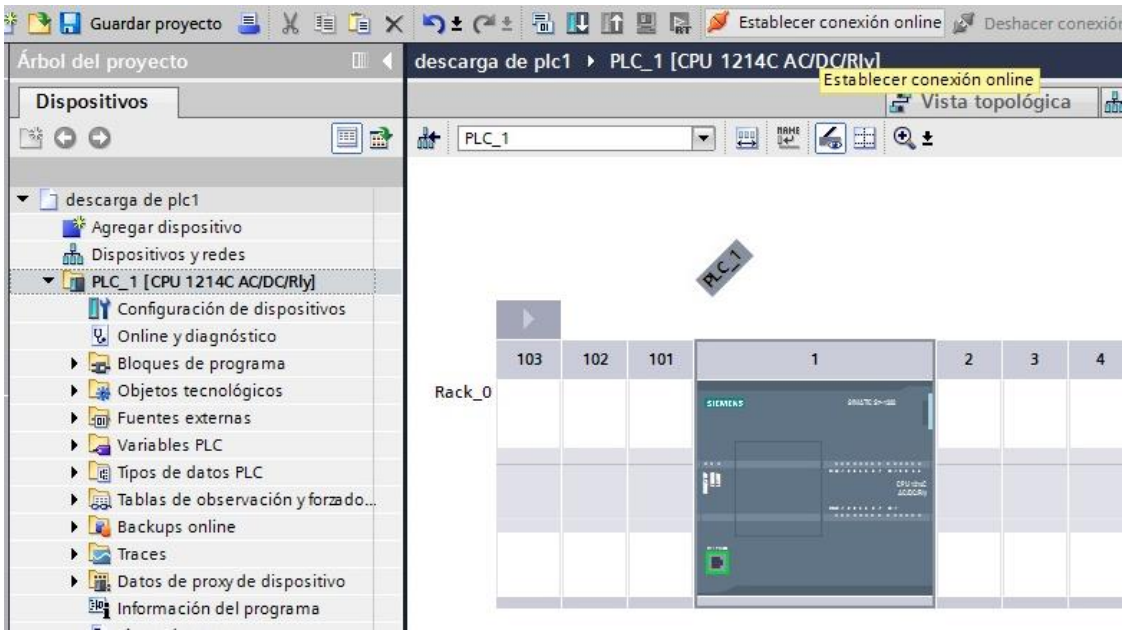

Figura 24.- Conexion con modelo S7 1200

6. Como se ve en la ventana siguiente, el enlace se realizó correctamente, si no hubiese sido posible la conexión se observaría diferentes avisos. Se realiza la carga del dispositivo.

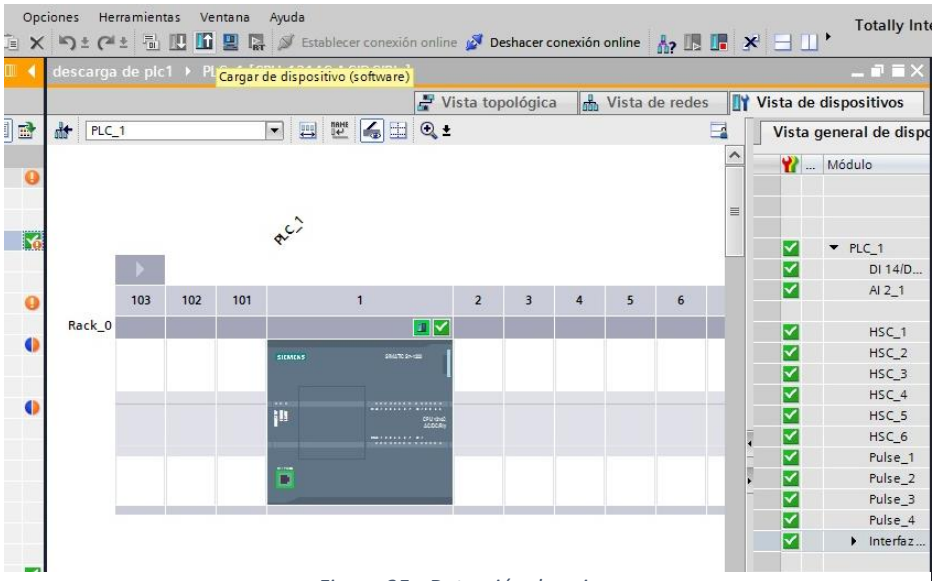

Figura 25.- Detección de avisos

7. En la ventana de Cargar de dispositivo, se observa un mensaje de verificaciones previas en la que se notifica que los datos han sido cambiados por los del proyecto del proyecto, se selecciona la pestaña de Continuar y se hace clic en Cargar de dispositivo.

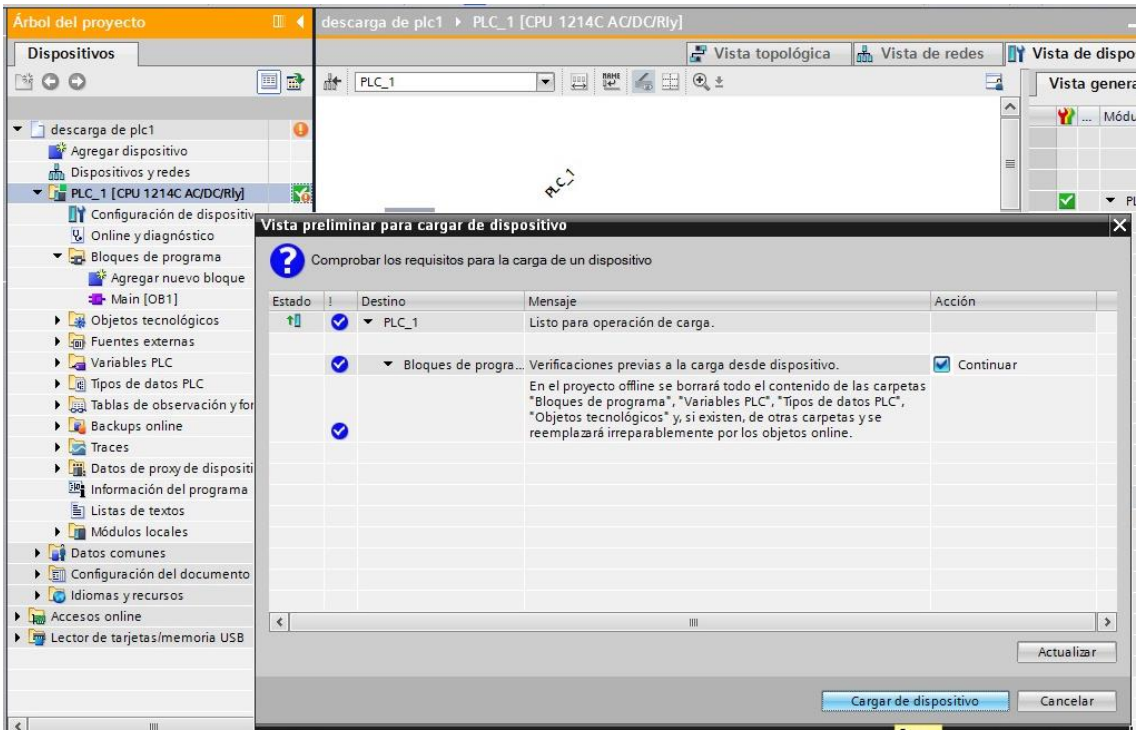

Figura 26.- Mensaje de confirmación

8. Una vez se haya descargado el programa del dispositivo en el proyecto se puede ver en tiempo real lo que está procesando, la tabla de variables, forzar variables e incluso poner la máquina en Stop cambiar el programa y volver a poner en Run cargando el nuevo programa.

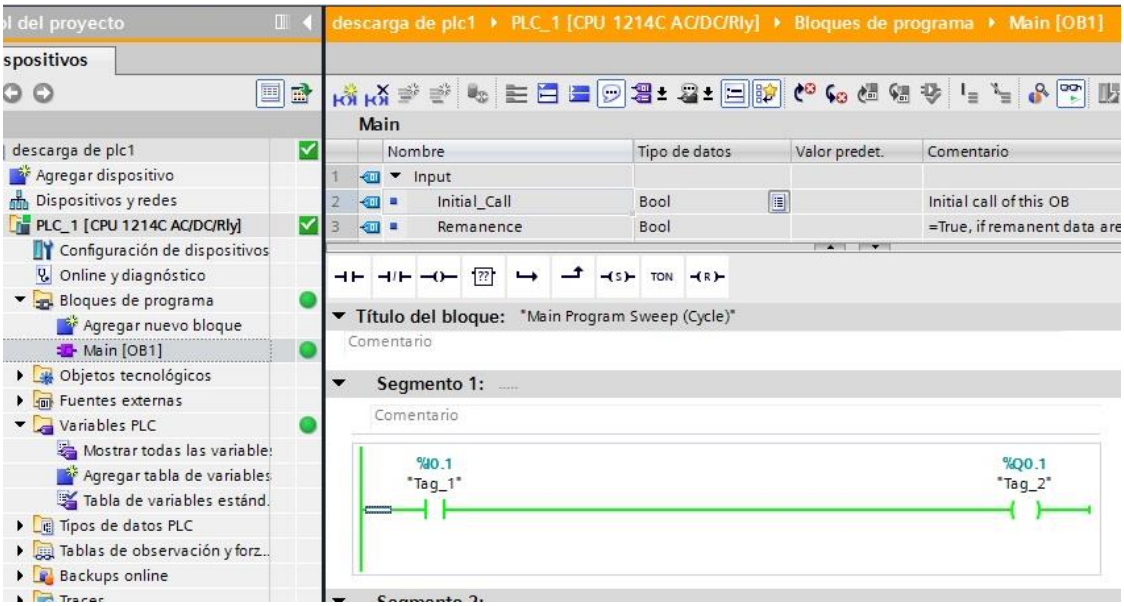

Figura 27.-Control en tiempo real del proceso

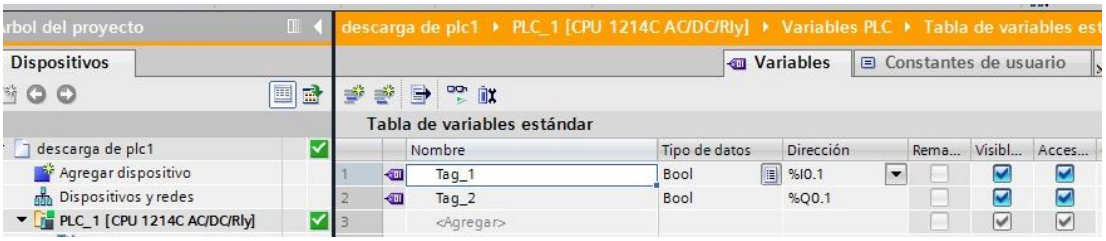

Figura 28.- Tabla de variables del proceso

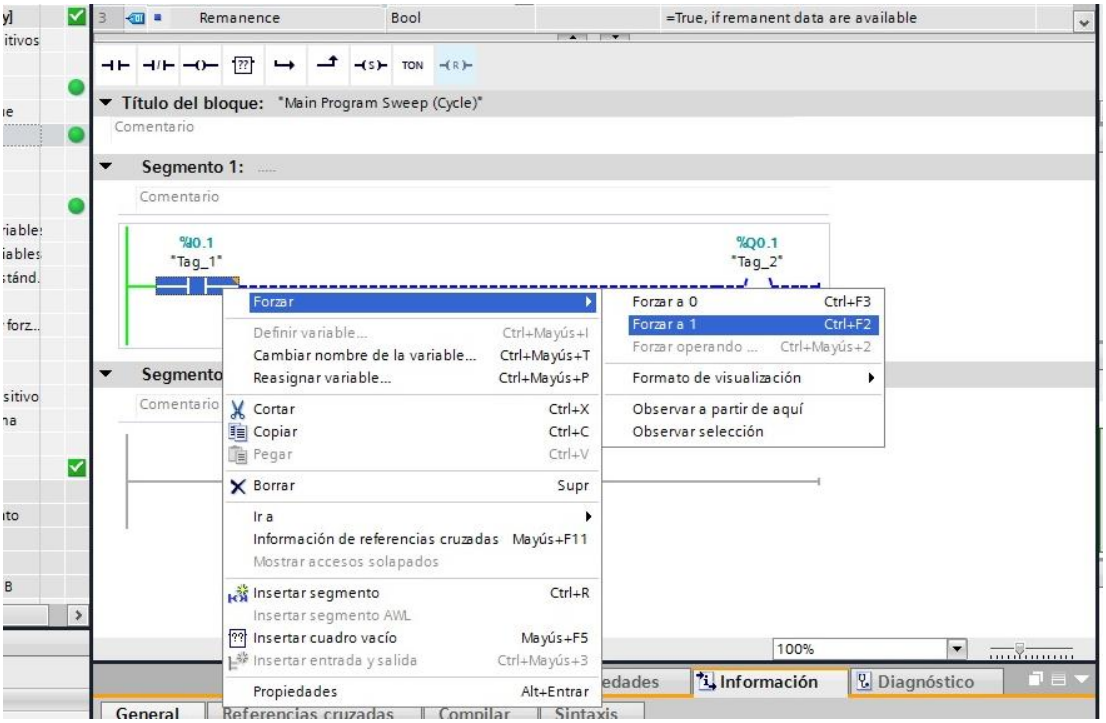

Figura 29.- Forzar variables

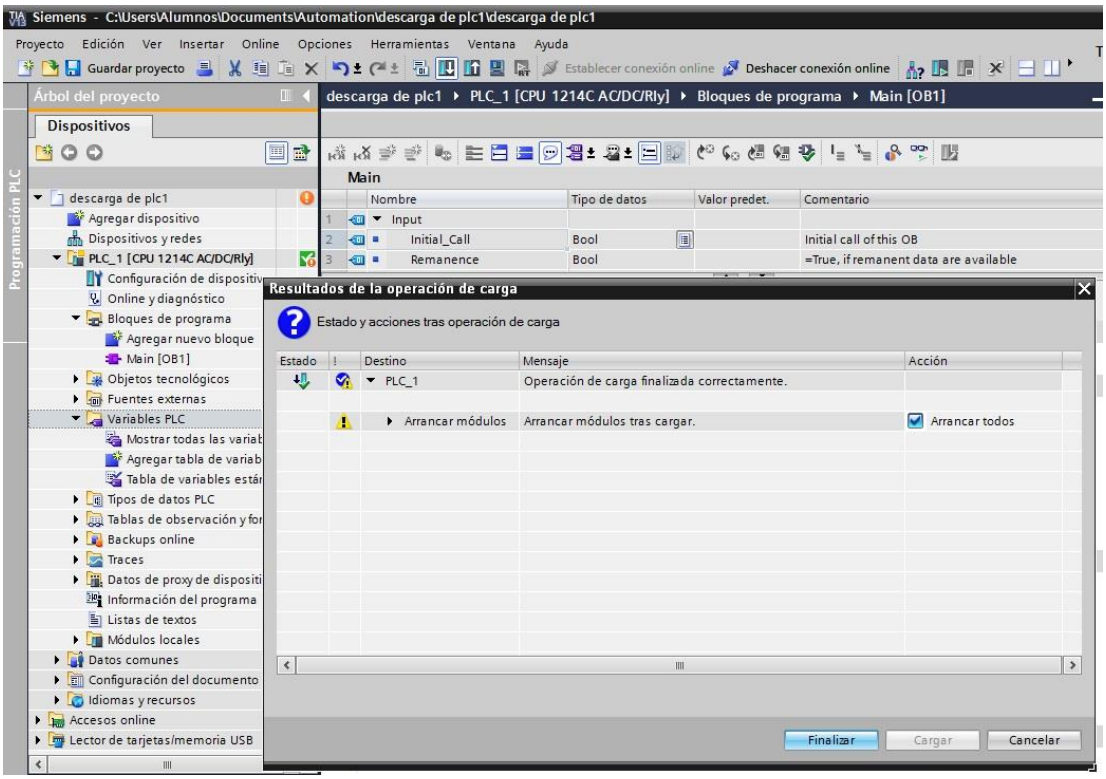

Figura 30.- Modificación del proceso

10. CONCLUSIÓN

Este proyecto, ha sido un gran reto en mi carrera de estudiante, haber tenido la oportunidad de investigar en seguridad en redes de control y aprender algunas técnicas de seguridad a nivel industrial, Hackeo, herramientas, metodologías y sus leyes, hace que pueda ver la estructura de redes de control y la piratería informática en ambientes industriales en primera persona, ya que en la carrera no tenemos la oportunidad de aprender sobre estos temas tabú.

Inicialmente hemos visto que en el área de seguridad tenemos por un lado aquella persona que se preocupa en investigar los fallos de seguridad y al mismo tiempo buscar soluciones para esta misma, y por otro lado tenemos otra persona que se dedica a buscar estos mismos fallos pero en vez de hacer uso de este mismo conocimiento para solucionarlo, se dedica a violar la integridad de la información, robar contraseñas, infiltrar en sistemas etc. La primera persona la llamamos hacker ético y la segunda hacker no ético.

Mi proyecto se basa en investigar a modo genérico y practicar el método básico de intrusión en una red de control y al mismo tiempo explicar lo importante que es implantar elementos de seguridad en redes Wifi para evitar estos fallos.

Para hablar de estos fallos hemos explicado inicialmente las principales vulnerabilidades que existen, donde destacamos algunas de las más conocidas como SQL Inyección, Ingeniería Social, Fuerza Bruta, Redes Wifi , entre otras, aunque estas quedan fuera de nuestro proyecto.

Hemos visto que fue necesario crear metodologías y buenas prácticas para llevar a cabo el trabajo del profesional de seguridad. También se han formado grandes organizaciones responsables de administrar y mantener todas las normas de seguridad informática, la OIS (Organización para la Seguridad en Internet) es la principal organización compuesta por varios fabricantes de productos, donde hacen investigaciones para mejorar la publicación de las vulnerabilidades de sus productos. Con esto estaría reduciendo el riesgo de que estas mismas vulnerabilidades puedan ser encontradas desde fuera y publicadas de forma indebida. En la parte de metodologías hemos visto que existe la OTP que es un proyecto para realizar testes en páginas Web. En nuestro proyecto hemos utilizado un caso práctico haciendo una intrusión en una red Wifi. En esta prueba se puede ver que la vulnerabilidad es una consecuencia directa de fallos en el diseño de los sistemas, limitaciones tecnológicas. No existe ningún sistema libre de

fallos y que sea seguro en su totalidad, pero lo que intentamos es poder minimizar los riesgos intentando garantizar el máximo posible la integridad de la red industrial.

El profesional de hacker ético tiene que conocer un conjunto de herramientas de seguridad y cada una tiene su función propia. Algunas son utilizadas para hacer auditorías de seguridad donde se puede realizar una búsqueda automática de los fallos en la red o sistemas. También existen las que podemos utilizar para realizar pruebas de penetración.

En mi proyecto se puede ver el caso Stuxnet que fue el caso más importante de hackeo de PLC a través de redes de control. Para garantizar una protección de la información y la privacidad de los datos es imprescindible que se intente minimizar al máximo los posibles ataques. Las consecuencias de estos ataques pueden ser desastrosas.

Para ello será necesario que las organizaciones dispongan de un personal capacitado y con experiencia en seguridad informática. Estos profesionales estudiaran la red y aplicaran una estrategia de seguridad para identificar los riesgos con los cuales la red se enfrenta a diario. Es importante que la empresa entienda la importancia de la seguridad informática y la transmita a sus trabajadores. Muchas veces el principal peligro se encuentra en el desconocimiento de los usuarios. Se debe de intentar reducir el riesgo asignando los permisos adecuados a cada usuario, realizando cambios de contraseña cada X tiempo, informar a los usuarios de que desconfíen de cualquier correo desconocido o página Web sospechosa y que antes de pinchar un pen drive externo a la empresa conozcan su procedencia. Estos son algunos de los métodos con los cuales reduciremos considerablemente el riesgo de ataques. No podremos asegurar que será 100% seguro ya que constantemente se encuentran nuevos agujeros de seguridad pero cuanto mayor sea el nivel de seguridad, mayor tendrá que ser el conocimiento de los hackers y menor posibilidad habrá de riesgo para la empresa.

11. BIBLIOGRAFIA

1. Internert Security Systems , Database of Intrusions detected by Network ICE. 2014 *<http://www.iss.net/security_center/advice/Intrusions/2000639/default>*

2. Foro el Hacker.net , 2014 <http://foro.elhacker.net/bugs_y_exploits/lista_sobre_paginas_de_exploits-t31370.0.html>

3. Institute of Information Security, mayo 2014, <http://iisecurity.in/courses/certified-professional-hacker-nxg.html?gclid=CLzf656cvr4CFfMftAodqCgAeQ>

4. Artículo, Servidor SSH 11 oct 2010.<http://www.guia-ubuntu.com/?title=Servidor_ssh>

5. Wikipedia información sobre Ubunbtu , 21 may 2014- *<http://es.wikipedia.org/wiki/Ubuntu>*

6. CAPTCHA: Telling Humans and Computers Apart Automatically -2010 Carnegie Mellon University *<http://www.captcha.net/>*

7. Página Oficial PHP, 2014 *<http://www.php.net/>*

8. The Ethical Hacker Network, Free Online Magazine for the Security Profesional 2014 *<https://www.ethicalhacker.net/>*

9. RevisaPCWORD,Feb15,2012 *<http://www.pcworld.com/article/250045/how_to_become_an_ethical_hacker.html>*

10. ArticulosobreCaptcha,Author:LuisCastro,2014 *<http://aprenderinternet.about.com/od/Glosario/g/Que-Es-Captcha.htm>*

11. Damn Vulnerable Web Application. 2014, *<http://www.dvwa.co.uk/>*

12. Wikipedia, la enciclopedia libre. 2014, <*http://es.wikipedia.org/*>

13. The art of security. <*http://t3rm1t.blogspot.com.es/*>

14. Infierno Hacker. *<http://foro.infiernohacker.com/>*

15. TechRepublic.< *http://www.techrepublic.com/*>

16. PHP: Hypertext Preprocessor.< *http://www.php.net/*>

17. Coding Horror. <*http://www.codinghorror.com/*>

18. EsLoMas. <*http://www.eslomas.com/*>

19. Mundo geek. *<http://mundogeek.net/>*

20. Zoidberg's research lab. *< http://0xzoidberg.wordpress.com/>*

21. M0unttik s0s4. *<http://mounttik.blogspot.com.es/>*

22. Thoughts go here *<http://beautaub.blogspot.com.es/>*

23. Devil's blog on Security < *http://nrupentheking.blogspot.com.es/*>

24. RedInfoCol <*http://www.redinfocol.org/*>

25. C# Corner <http://www.c-sharpcorner.com/>

26. The Code <Project http://www.codeproject.com/>

27. Stackoverflow <http://stackoverflow.com/>

28. Julio Gómez López, Eugenio Villar Fernández, Alfredo Alcayde García. Seguridad en Sistemas Operativos Windows y GNU/Linux (2ª Edición Actualizada**).** Ra-Ma Editorial. ISBN: 978-84-9964- 116-4. 2011.

29. *DE MIGUEL, María del Rosario y Juan Vicente Oltra.* Deontología y aspectos legales de la informática: cuestiones éticas, jurídicas y técnicas básicas. *Valencia: Ed. UPV, 2007. ISBN 978- 84-8363-112-6.7645-7418-3.*

30. *HERNÁNDEZ, Claudio. Hackers:* Los piratas del Chip y de Internet. *1999*

31. *LEVY, Steven. Hackers. Capítulo:* La ética del hacker. *Ed. Penguin, 2001*

32. *MALAGÓN, Constantino.* Hacking ético. *Universidad Nebrija. Madrid.*

33. *PRENAFETA Rodríguez, Javier.* Consecuencias jurídicas de los ataques a sistemas informáticos. *Identificador: 1010087527283. 08-oct-2010 1:07 UTC. Tipo de obra: Literaria, Artículo.*

34. *PRENAFETA Rodríguez, Javier.* Consecuencias jurídicas de los ataques a sistemas informáticos. *19-mar-2005. Tipo de obra: Literaria, Artículo.*

35. *Garcia-Moran,Jean Paul.* Hacking y seguridad en Internet

36. *Mikhailovsky, Andrei.* Hacking Wireless

37. *Ramos,Picouto Fernando.* Hacking Práctico. *Anaya*

38. *Dhanjani, Nitesh,* Generación Hacker. *O'Reilley*

39. *Pérez, Carlos Míguez.*Hacker Edición 2010. *Anaya*

40. Software libre para servicios de información digital / *Jesús Tramullas Saz, Piedad Garrido Picazo, coordinadores*

41. Hacking Wireless 2.0 / *Johnny Cache, Joshua Wright, Vincent Liu*

42. Hacking y seguridad en Internet / *Jean Paul García-Moran ... [et al.]*

43. Hacking práctico / *Fernando Picouto Ramos, Abel Mariano Matas García, Antonio Ángel Ramos Varón*

44. CEH Certified Ethical Hacker Study Guide

45. The RootKit Arsenal - *Reverend Bill Blunden*

46. The Basics of Hacking and Penetration Testing - Ethical Hacking - *Patrick Engebretson*

47. SQL Injection Attacks and Defense - *Justin Claake*

48. Malware Analyst's Cookbook: Tools and Techniques for Fighting Malicious Code- Michael Ligh, Steven Adair, Blake Hartstein , Matthew Richard

49. Coding For Penetration Testers - *Jason Andreess - Ryan Linn*

50.OISSG Open Information Systems Securty Group. 2003 - 2012<*http://www.oissg.org*>

51. GNU Operating System Sponsored by the Free Software Foundation. 2014/05/15 < http://www.gnu.org/>

52. Agencia Estatal Boletín Oficial del Estado. 2014 < *http://www.boe.es/*>

53. Comunidad Linux. 2014 <*http://www.linux.org/*>

54. Tenable network security, provedora de la herramienta Nessus, 2014 <http://www.tenable.com/products/nessus?gclid=CK2xwJGavr4CFScHwwod9VMAZA>

55. Software Enginnering Institute – Carnegie Mellon University. 2014 <*http://www.cert.org/tech_tips/malicious_code_mitigation.html*>

56. W3Schools is optimized for learning, testing and traning 1999-2014 *http://www.w3schools.com/HTML/html_entities.asp.*

57. "Puyosa Piña, H.D. Seguridad informática n sistemas de control y automatización: Estrategias para su aplicación en la industria química. Ingeniería Química, No.462 . pp. 304- 313. 2008.".

12. ANEXOS

12.1. Anexo_1_Legislaciones

Legislaciones

Como los ordenadores se han convertido en las nuevas herramientas que se utilizan para cometer delitos tradicionales y nuevos delitos, las dos entidades han tenido que buscar de manera independiente un nuevo concepto, conocido hoy en día como la ley "**ciberley**", varios países están trabajando para crear medidas para regular los delitos informáticos.

Se han buscado países que están trabajando con la ley ciberley y se encontró que en Estados Unidos se han creado algunos estatutos para los delitos informáticos.

Artículo 1029 de 18 USC (ciberley)

Estatuto del dispositivo de acceso, está relacionado con el fraude y la actividad ilegal que pueden producirse por el uso de dispositivos de acceso falsos que tienen relación con el comercio internacional.

Dispositivo de acceso hace referencia a un tipo de aplicación o pieza de hardware que ha sido creada específicamente para generar credenciales de acceso (passwords, número de tarjeta de crédito, códigos de accesos para servicios telefónicos de larga distancia, números personales de identificación, etc.)

Código Penal Español referente a Delitos Informáticos.

El Código Penal español regula algunos tipos de delitos relacionado con la informática:

Delito	**Código**	**Condena**
La propiedad intelectual.	CP arts. 270-272	Pena de prisión de seis meses a dos años.
La propiedad industrial.	CP arts. 273-277	Pena de seis meses a dos años.
El derecho a la intimidad.	CP arts. 197-201	Pena de un año a cuatro años y multa de multa de 12.000€
Estafas, apropiación indebida.	CP arts. 252-254	Pena de seis meses a cuatro años
Sabotaje informático.	CP arts. 263	Pena de seis meses a 24 meses
Contra la libertad y amenazas.	CP arts. 169	Pena de un año a cinco años
Uso indebido de cualquier terminal de telecomunicación sin consentimiento de su titular.	CP art. 256	Pena de multa de 3 a 12 meses
Publicidad engañosa.	(CP art. 282)	Pena de seis meses a un año.
Falsedades documentales.	CP arts. 390	Pena de tres años a seis años
Provocación sexual y prostitución.	CP arts. 187, 189	Pena de año a cuatro años.

Después de comprobar las leyes que se aplican en EUA frente las que leyes que existen en España se comprueba una diferencia muy grande entre una y otra. En EUA las leyes son mucho más severas.

12.2. Anexo 2 Herramientas Utilizadas

Herramientas

Existe una variedad enorme de herramientas para realización de tests de penetración (Pentesting), el responsable técnico debe escoger la herramienta adecuada para llevar a cabo su tarea, aquí pongo una breve justificación de los programas que se ha utilizado y más adelante se realiza su completa descripción.

Para este proyecto se ha utilizado las siguientes herramientas gratuitas:

Escaneo de puertos

- **Nmap** se trata del rastreador de puertos por excelencia para cualquier profesional del mundo de la seguridad. La principal misión de Nmap es la de permitir a los administradores de sistemas hacer barridos a sus redes y máquinas para determinar qué puertos tienen activos, y así solucionar posibles debilidades en su seguridad.

Motores para pruebas de penetración

- **Wifislax** herramienta diseñada para la auditoria de seguridad y relacionada con la seguridad informática en general. Tienen números escáner de puertos y vulnerabilidades, herramienta para creación y diseño de exploits, sniffers, herramientas de análisis forense y herramientas para la auditoria Wireless. Su licencia está bajo la GNU/Linux.

12.3. Anexo 3 Software y Hardware utilizados

WIFISLAX

WiFiSlax es una distribución GNU/Linux en formato *.iso basada en Slackware con funcionalidades de LiveCD y LiveUSB pensada y diseñada para la auditoría de seguridad y relacionada con la seguridad informática en general.

WiFiSlax incluye una larga lista de herramientas de seguridad y auditoría listas para ser utilizadas, entre las que destacan numerosos escáner de puertos y vulnerabilidades, herramientas para creación y diseño de exploits, sniffers, herramientas de análisis forense y herramientas para la auditoría wireless, además de añadir una serie de útiles lanzadores.

Posee una gran integración de varios controladores de red no oficiales en el kernel de Linux, y da así soporte inmediato para un gran número de tarjetas de red cableadas e inalámbricas.No hace falta enumerar todas las características que tiene ya que se salen

Figura 31.-WiFilax

fuera de este proyecto, pero si podemos decir que es el mejor desencriptador de claves wifi gratuito que existe a día de hoy y que es actualizado cada poco tiempo, además de su fácil instalación y manejo.

En su formato más básico ya cuenta con más de 50 aplicaciones con las que se pueden descifrar claves WEP, WPA, PSK y WPA2.

NMAP

Nmap es un programa de código abierto que sirve para efectuar rastreo de puertos escrito originalmente por Gordon Lyon (más conocido por su alias *Fyodor Vaskovich)* y cuyo desarrollo se encuentra hoy a cargo de una comunidad. Fue creado originalmente para Linux aunque actualmente es multiplataforma. Se usa para evaluar la seguridad de sistemas informáticos, así como para descubrir servicios o servidores en una red informática, para ello Nmap envía unos paquetes definidos a otros equipos y analiza sus respuestas.

Este software posee varias funciones para sondear redes de computadores, incluyendo detección de equipos, servicios y sistemas operativos. Estas funciones son extensibles mediante el uso de scripts para proveer servicios de detección avanzados, detección de vulnerabilidades y otras aplicaciones. Además, durante un escaneo, es capaz de adaptarse a las condiciones de la red incluyendo latencia y congestión de la misma.

CARACTERISTICAS

- Descubrimiento de servidores: Identifica computadoras en una red, por ejemplo listando aquellas que responden ping.
- Identifica puertos abiertos en una computadora objetivo.
- Determina qué servicios está ejecutando la misma.
- Determinar qué sistema operativo y versión utiliza dicha computadora, (esta técnica es también conocida como *fingerprinting*).
- Obtiene algunas características del hardware de red de la máquina objeto de la prueba.

APLICACIONES

Ha llegado a ser una de las herramientas imprescindibles para todo administrador de sistema, y es usado para pruebas de penetración y tareas de seguridad informática en general.

Como muchas herramientas usadas en el campo de la seguridad informática, es también una herramienta muy utilizada para hacking.

Los administradores de sistema pueden utilizarlo para verificar la presencia de posibles aplicaciones no autorizadas ejecutándose en el servidor, así como los crackers pueden usarlo para descubrir objetivos potenciales.

Nmap permite hacer el inventario y el mantenimiento de computadores de una red. Se puede usar entonces para auditar la seguridad de una red, mediante la identificación de todo nuevo servidor que se conecte, es difícilmente detectable, ha sido creado para evadir los Sistema de detección de intrusos (IDS) e interfiere lo menos posible con las operaciones normales de las redes y de las computadoras que son analizadas.

SOFTWARE TIA PORTAL V13

Software para la creación de SCADA que nos ofrece dos vistas diferentes de las herramientas disponibles, distintos portales orientados a tareas organizados según las funciones de las herramientas (vista del portal o vista del proyecto) y un manual detallado de todas las funciones y ayudas. El usuario puede seleccionar la vista que considere más apropiada para trabajar eficientemente. Con un solo clic es posible cambiar entre la vista del portal y la vista del proyecto.

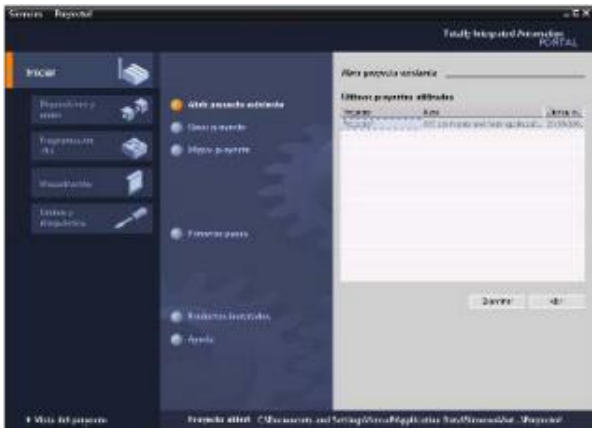

Figura 32.- Tia Portal

- La vista del portal ofrece una vista funcional de las tareas del proyecto y organiza las funciones de las herramientas según las tareas que deban realizarse, p. ej. Configurar los componentes de hardware y las redes. Es posible determinar fácilmente el procedimiento y la tarea que debe seleccionarse.
- La vista del proyecto proporciona acceso a todos los componentes del proyecto. Puesto que todos estos componentes se encuentran en un lugar, es posible acceder fácilmente a todas las áreas del proyecto.

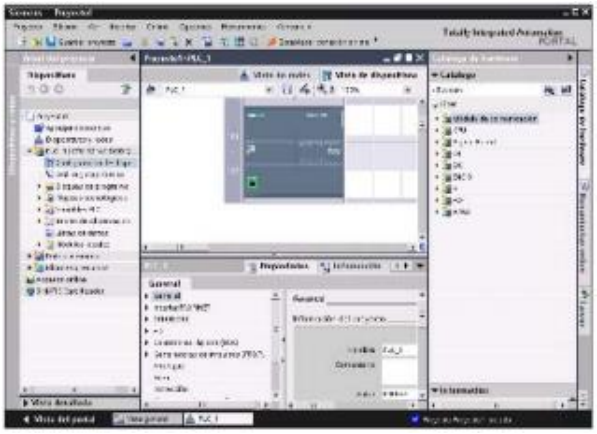

Figura 33.- Acceso a diferentes componentes del proyecto

El proyecto contiene todos los elementos que se han creado o finalizado.

Para poder solucionar las tareas de forma **rápida y eficiente**, STEP 7 Basic proporciona asistencia inteligente donde se necesite:

- En los campos de entrada se ofrece ayuda "roll-out" que facilita la entrada de la información correcta. Por ejemplo, si se introduce un valor incorrecto, aparecerá un texto de aviso en el que se indica el rango de valores válidos.
- Algunos de los tooltips de la interfaz de usuario se abren "en cascada", ofreciendo información adicional. Algunos de estos contienen enlaces a temas específicos del sistema de información.

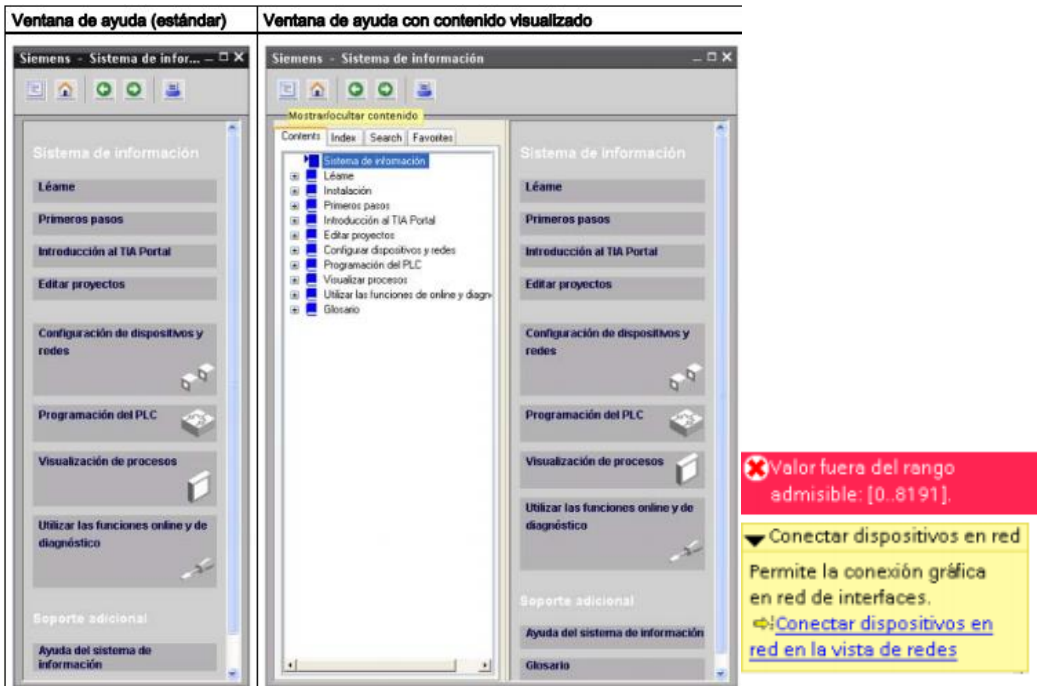

Figura 34.-Enlaces a temas específicos

En los campos de entrada de los diferentes diálogos y Task Cards se ofrece asistencia en forma de un cuadro de texto desplegable que informa proporciona información acerca del rango o los tipos de datos requeridos. Los elementos de la interfaz de usuario ofrecen tooltips que explican la función del elemento. Algunos de los elementos, tales como "Abrir" o "Guardar", no requieren información adicional.

No obstante, algunos de los elementos ofrecen un mecanismo que permite ver una descripción adicional del elemento en cuestión. Esta información adicional se visualiza en un tooltip "en cascada". (Un triángulo negro junto al tooltip indica que hay más información disponible.) El tooltip aparece cuando se sitúa el puntero del ratón sobre un elemento de la interfaz de usuario. Para visualizar información adicional, el puntero del ratón se debe situar sobre el tooltip. Algunos de los tooltips en cascada también ofrecen enlaces a temas específicos del sistema de información. Al hacer clic en el enlace se visualiza el tema en cuestión.

STEP 7 Basic pone a disposición un completo sistema de información online y Ayuda en pantalla, en el que se describen todos los productos SIMATIC que se han instalado. El sistema de información incluye asimismo información de referencia y ejemplos. Para visualizar el sistema de información, seleccione uno de los puntos de acceso siguientes:

- En la vista del portal, seleccione el portal de inicio y haga clic en "Ayuda".
- En la vista del proyecto, elija el comando "Mostrar ayuda" del menú "Ayuda".
- En un tooltip en cascada, haga clic en un enlace para ver más información sobre ese tema.

El sistema de información se abre en una ventana que no oculta las áreas de trabajo. Haga clic en el botón "Mostrar/ocultar contenido" del sistema de información para ver el contenido y desacoplar la ventana de ayuda. Entonces se puede cambiar el tamaño de la ventana de ayuda. Utilice las fichas "Contenido" o "Índice" para buscar un tema o palabra clave en el sistema de información.

PLC S7 1200

Es el Hardware donde hemos realizado la descarga de datos, es un controlador lógico programable (PLC) que ofrece la flexibilidad y capacidad de controlar una gran variedad de dispositivos para las distintas tareas de automatización.

Su diseño compacto, configuración flexible y amplio juego de instrucciones, hacen que muy fácil su uso para controlar una gran variedad de aplicaciones.

La CPU incorpora un microprocesador, una fuente de alimentación integrada, así como circuitos de entrada y salida en una carcasa compacta, conformando así un potente PLC.

Una vez cargado el programa en la CPU, ésta contiene la lógica necesaria para vigilar y controlar los dispositivos de la aplicación. La CPU vigila las entradas y cambia el estado de las salidas según la lógica del programa de usuario, que puede incluir lógica booleana, instrucciones de contaje y temporización, funciones matemáticas complejas, así como comunicación con otros dispositivos inteligentes.

Numerosas funciones de seguridad protegen el acceso tanto a la CPU como al programa de control:

- Toda CPU ofrece protección por contraseña que permite configurar el acceso a sus funciones.
- Es posible utilizar la "protección de know-how" para ocultar el código de un bloque específico. Encontrará más detalles en el capítulo "Principios básicos de programación"

La CPU incorpora un puerto PROFINET para la comunicación en una red PROFINET. Los módulos de comunicación están disponibles para la comunicación en redes RS485 o RS232.

La gama S7-1200 provee módulos de comunicación (CMs) que ofrecen funciones adicionales para el sistema. Hay dos módulos de comunicación, a saber: RS232 y RS485.

- La CPU soporta como máximo 3 módulos de comunicación.
- Todo CM se conecta en lado izquierdo de la CPU (o en lado izquierdo de otro CM)

Para programar este autómata es necesario el software STEP 7 Basic que ofrece un entorno amigable que permite desarrollar, editar y observar la lógica del programa necesaria para controlar la aplicación, incluyendo herramientas para gestionar y configurar todos los dispositivos del proyecto, tales como PLCs y dispositivos HMI. STEP 7 Basic ofrece dos lenguajes de programación (KOP y FUP) que permiten desarrollar el programa de control de la aplicación de forma fácil y eficiente. Asimismo, incluye las herramientas para crear y configurar los dispositivos HMI en el proyecto.

ESPECIFICACIONES

Función	CPU 1211C	CPU 1212C	CPU 1214C
Dimensiones físicas (mm)	90 x 100 x 75		110 x 100 x 75
Memoria de usuario • Memoria de trabajo • Memoria de carga • Memoria remanente	 • 25 KB • 1 MB • 2 KB		 • 50 KB • 2 MB • 2 KB
E/S integradas locales • Digitales • Analógicas	 • 6 entradas/4 salidas • 2 entradas	 • 8 entradas/6 salidas • 2 entradas	 • 14 entradas/10 salidas • 2 entradas
Tamaño de la memoria imagen de proceso	1024 bytes para entradas (I) y 1024 bytes para salidas (Q)		
Área de marcas (M)	4096 bytes		8192 bytes
Ampliación con módulos de señales	Ninguna	2	8
Signal Board	1		
Módulos de comunicación	3 (ampliación en el lado izquierdo)		
Contadores rápidos • Fase simple • Fase en cuadratura	3 • 3 a 100 kHz • 3 a 80 kHZ	4 • 3 a 100 kHz 1 a 30 kHz • 3 a 80 kHz 1 a 20 kHz	6 • 3 a 100 kHz 3 a 30 kHz • 3 a 80 kHz 3 a 20 kHz
Salidas de impulsos	2		
Memory Card	SIMATIC Memory Card (opcional)		
Tiempo de respaldo del reloj de tiempo real	Típico: 10 días / Mínimo: 6 días a 40 °C		
PROFINET	1 puerto de comunicación Ethernet		
Velocidad de ejecución de funciones matemáticas con números reales	18 µs/instrucción		
Velocidad de ejecución booleana	0,1 µs/instrucción		

Módulo		Sólo entradas	Sólo salidas	Entradas y salidas
Módulo de señales (SM)	Digital	8 entradas DC	8 salidas DC 8 salidas de relé	8 entradas DC/8 salidas DC 8 entradas DC/8 salidas de relé
		16 entradas DC	16 salidas DC 16 salidas de relé	16 entradas DC/16 salidas DC 16 entradas DC/16 salidas de relé
	Analógico	4 entradas analógicas 8 entradas analógicas	2 salidas analógicas 4 salidas analógicas	4 entradas analógicas/2 salidas analógicas
Signal Board (SB)	Digital	-	-	2 entradas DC/2 salidas DC
	Analógico	-	1 salida analógica	-
Módulo de comunicación (CM) • RS485 • RS232				

12.4. Anexo 4 Maqueta de Trabajo

Maqueta de Trabajo

Figura 35.- Maqueta de trabajo

Para poder simular el entorno industrial y realización de pruebas hemos usado el laboratorio de robótica con la ayuda de algunas máquinas autómatas.

No he necesitado más que mi portátil con todos los programas descritos anteriormente, una red local conectada a un router wifi, el PLC de la imagen en cuestión y un ordenador del laboratorio conectado por cable al PLC desde el que cargaba el programa de bloques original.