Sistema para la Detección de Intrusos en Plataformas SCADA

Andrés Felipe Sánchez Prisco

Universidad de Antioquia
Faculta de Ingeniería
Medellín, Colombia
2017

Sistema para la Detección de Intrusos en Plataformas SCADA

Andrés Felipe Sánchez Prisco

Tesis presentada como requisito para optar por el título de:
Magister en Ingeniería

Director:
PhD. John Freddy Duitama Muñoz

Grupo de Investigación Ingeniería y Software
Línea de Investigación en Aprendizaje Automático y Ciberseguridad

Universidad de Antioquia
Faculta de Ingeniería
Medellín, Colombia
2017

Dedicatoria

“Que hombre que quiere construir una torre, no se sienta primero y calcula…”

Jesucristo

A mi familia, profesores y amigos…

Agradecimientos

Este proyecto de maestría fue desarrollado bajo la dirección del Profesor PhD. John Freddy Duitama Muñoz y la cobertura del grupo de investigación Ingeniería y Software del Departamento de Sistemas de la Facultad de Ingeniería de la Universidad de Antioquia.

Resumen

Los sistemas SCADA, acrónimo de Supervisory Control And Data Acquisition (Supervisión, Control y Adquisición de Datos), son redes de control que permiten el monitoreo y gestión de procesos industriales de forma remota.

En sus inicios, su prioridad más importante era la disponibilidad de la información de forma bidireccional entre la estación de control y las unidades remotas; no obstante, el creciente escalamiento de los sistemas industriales, así como la conectividad a internet ha llevado a reconsiderar el antiguo paradigma para darle más importancia al tema de la seguridad, con el fin de evitar que un posible ciberataque ponga en peligro el funcionamiento del sistema SCADA. Estos ataques pueden llegar a afectar incluso la industria y poner en juego toda la seguridad de un país.

El presente trabajo de incentivación propuso la creación de un sistema adaptable para la detección de intrusos o IDS (por sus siglas en inglés) en redes SCADA, mediante el uso de técnicas de aprendizaje de máquinas de tipo supervisado, orientadas al análisis de variables de los dispositivos de control. Una máquina de soporte vectorial del tipo “One Class” y un laboratorio de pruebas, permitió la validación del modelo propuesto.

Palabras Clave: IDS, SCADA, ciberseguridad.

Abstract

SCADA systems, an acronym for Supervisory Control And Data Acquisition (supervisory, Control and data acquisition), can be defined as control networks that allow the monitoring and management of industrial processes remotely.

At the beginning, their top priority were the availability of information bidirectionally between the control station and the remote units; however, the growing escalation of industrial systems, as well as internet connectivity has led to reconsider the old paradigm to give more importance to the issue of security, in order to avoid a possible cyber attack endangers the functioning of the SCADA system. These attacks can affect even the industry and put into play all the security of a country.

The present thesis work proposes the creation of an adaptive system for the detection of intruders or IDS (for its acronym in English) on SCADA networks, through the use of machines of type supervised learning techniques, oriented to the analysis of variables of the control devices. A support vector of type "Class One" machine and a test lab, allowed the validation of the proposed model.

Keywords: IDS,SCADA, cybersecurity

Contenido

Índice de Figuras

Índice de Tablas

Introducción

Los sistemas SCADA o sistemas de supervisión, control y adquisición de datos por sus siglas en inglés, se pueden definir en términos simples como redes de controladores y computadoras que se crean con el fin de controlar y monitorear de forma remota todo tipo de procesos industriales [1]. Básicamente, utilizan una capa física bastante similar a los sistemas de información tradicionales, no obstante, se diferencian de éstos en sus prioridades, siendo la disponibilidad, la más importante en los sistemas de control [2].

Debido a que en los últimos años dichos sistemas han sido el blanco de diversos ataques, se hace necesaria la implementación de sistemas de seguridad que permitan la detección de posibles ataques en tiempo real. Se pretende que de esa manera un operario sea alertado y en consecuencia pueda actuar de forma inmediata ante un inminente caso de manipulación indebida de la planta.

En los sistemas SCADA, dos tipos de variables son los que permiten el monitoreo y la manipulación de una planta de forma remota. El primer tipo se les conoce como variables de supervisión y su propósito es transportar información desde la planta hasta el cliente SCADA. Al segundo tipo se les conoce como variables de control y su propósito es poder ejecutar órdenes desde la aplicación cliente. La solución que se propone está dirigida a lograr la detección de intrusos; para ello debe tener la capacidad de reconocer los patrones de comportamiento normal de un sistema SCADA, de tal forma que, al identificar un patrón atípico en el comportamiento de las variables de los dispositivos de control, se pueda reconocer un posible ciberataque.

Aunque varias técnicas y algoritmos han sido reportados en la literatura para la detección de intrusos; la mayoría de ellos solo pueden identificar patrones de ataque previamente conocidos utilizando metodologías basadas en reglas; en otras soluciones que usan algoritmos de aprendizaje de máquinas, estas se encuentran restringidas a un determinado tipo de protocolo de comunicación. Por otra parte, en lo mejor de nuestro conocimiento todos los trabajos realizados se enfocan en el análisis del tráfico de red, ignorando el comportamiento interno de las variables dentro de los dispositivos de control y por ende los efectos mediáticos que estos podrían tener sobre el proceso industrial.

El presente trabajo propone y desarrolla un sistema que usando como descriptores las variables de control y supervisión de un sistema SCADA, permite la detección de posibles intrusos; la estrategia usada permite generar alertas cuando se evidencie un comportamiento anormal de un controlador en tiempo real. Las máquinas de soporte vectorial del tipo "One Class", en adelante OCSVM por sus siglas en inglés, permiten la detección de patrones atípicos una vez han sido entrenadas con un conjunto de variables y características que describen un comportamiento considerado como normal, por lo cual se propone su utilización como método de detección de anomalías [3]. En lo que respecta a los protocolos de comunicación, se propone la utilización del estándar OPC (OLE for Process and Control), el cual, con la ayuda del software Labview, permitirá la interoperabilidad con dispositivos de diferentes marcas, sin la necesidad de estar condiciones a un protocolo específico, como: Modbus, Profibus, etc.

El resto de este trabajo se encuentra distribuido como se muestra a continuación:

En el capítulo 1 se hace el planteamiento del problema, mientras que en el capítulo 2 se encuentra el marco teórico en donde se explican todos los conceptos relacionados en el desarrollo de este proyecto. En el capítulo 3 se hace un análisis de los trabajos relacionados en el estado del Arte. En el capítulo 4, se explica en forma detallada la solución propuesta. A continuación, en el capítulo 5 se evalúa el desempeño del modelo propuesto, para finalmente, en el capítulo 6 obtener las conclusiones y comentarios.

1.Planteamiento del Problema

Un sistema informático seguro debe cuidar de cumplir con los pilares de la ciberseguridad, que son: Confidencialidad, Integridad y Disponibilidad, como se observa en la figura 1[4]. Considerando que en los sistemas SCADA, el paradigma dominante desde sus inicios ha sido la disponibilidad de la información[2], se han descuidado los demás aspectos dando lugar a un escenario en donde muchos de las comunicaciones industriales carecen de encriptación y como si fuere poco los equipos de supervisión y control corren bajo versiones de software obsoletas, altamente vulnerables a todo tipo de malware[8].

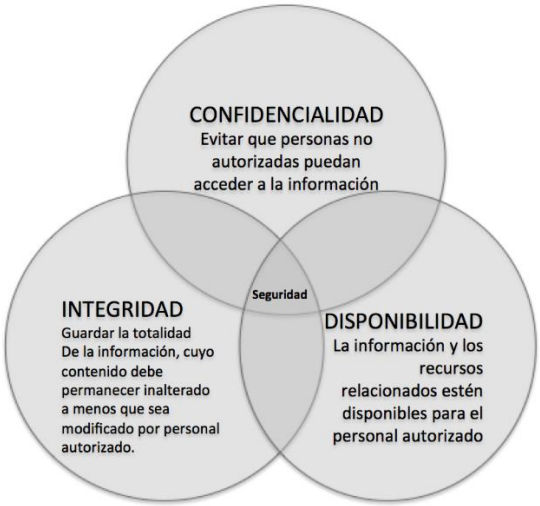


Figura 1.Pilares de la Ciberseguridad

En el pasado, los sistemas SCADA se concebían como entornos totalmente seguros ya que en la mayoría de los casos se encontraban totalmente aislados de las redes de acceso público, haciendo prácticamente imposible un ciber-ataque externo. No obstante, en el año 2010 un gusano informático llamado Stuxnet, atacó un sistema SCADA propietario de Siemens llamado WinCC. A este gusano se le reconoce por tener la capacidad de reprogramar los PLC y ocultar los cambios ejecutados, siendo además capaz de propagarse usando dispositivos de almacenamiento masivo[5]. El virus, no solo dejó en evidencia las vulnerabilidades de los sistemas SCADA, sino que además prendió las alertas a nivel mundial ante un posible ciberataque que pusiera en jaque las infraestructuras críticas (electricidad, industria química,etc) y por ende la seguridad de una nación entera[6]. En el caso particular de Colombia, en el año 2011 el CONPES (Consejo Nacional de Política Económica y Social) determinó los avances en ciberseguridad industrial como tema prioritario para la protección de la infraestructura del país ante una eventual guerra cibernética[7] ;mientras que el reporte anual de ciberataques de DELL puso en evidencia, que solo en el año 2014, el número de ataques a sistemas SCADA se duplicó en comparación con el año 2012[8].

Además de las amenazas que pueden presentarse al interior de los sistemas SCADA, muchos de estos sistemas ya no se encuentran totalmente aislados de las redes públicas como internet; por tanto, conceptos que antes solo tenían validez en las telecomunicaciones tradicionales como seguridad perimetral, firewalls, detectores de intrusos, entre otros, ahora son tenidos en cuenta y son el motivo del desarrollo de múltiples investigaciones a nivel mundial, especialmente en Europa y Norteamérica, de las cuales se hablará más adelante.

Dada su falencia en temas de seguridad, los sistemas SCADA se hacen vulnerables, entre otros tipos de ataques, a ataques del tipo "Suplantación de Identidad". En este tipo de ataques un hacker podría manipular indebidamente las comunicaciones de la red, inyectando en el sistema comandos o paquetes corruptos que los elementos finales de control asumen provenientes de un operador confiable [9]. Si bien es cierto que todo tipo de ataque cibernético produce efectos no deseados en el funcionamiento de un sistema autónomo, los ataques que involucran inyección de comandos son más peligrosos aún, ya que modifican directamente los parámetros de control de uno o varios procesos haciendo que estos colapsen. Teniendo en cuenta que en un sistema SCADA existen variables de supervisión y de control, un ataque que corrompa la información de las variables supervisadas traería como consecuencia información falsa para el operador, lo que implicaría, por ejemplo, que éste no detectaría posibles anomalías; mientras que un ataque que corrompa la información de las variables controlados traería como consecuencia el mal funcionamiento de los actuadores(motores, pistones, resistencias) casi de forma inmediata, pudiendo ocasionar un desastre.

En este punto, un sistema detector de intrusos, IDS por sus siglas en inglés, que solo puede detectar ataques conocidos o que simplemente verifica cambios en el comportamiento del tráfico, no sería capaz de identificar algunos tipos de intrusión, pues para lograrlo requiere de antemano conocer cuál es el comportamiento de las variables de proceso en cada uno de sus diferentes estados. En otras palabras, un sistema IDS ideal debería ser en principio entrenado y a partir de las variables del sistema aprender cuál es su comportamiento ideal, lo que implica la utilización de algoritmos de aprendizaje de máquinas, de forma tal que pueda reconocer patrones de comportamiento anómalos mediante el análisis profundo de las variables independientemente de cual sea la procedencia de los mismos. Por otro lado, la lectura de las variables directamente en el dispositivo controlador, permite determinar incluso la inserción de una posible anomalía, que, aunque no tiene repercusión inmediata podría traer efectos catastróficos a largo plazo.

A través del presente trabajo, se describe el planteamiento, diseño e implementación de un sistema para la detección de intrusos en controladores asociados a una red SCADA que permita, mediante el uso de algoritmos de aprendizaje de máquinas, detectar cuando un

paquete corrupto pretende modificar el normal funcionamiento de un proceso. Esto con el fin de alertar a los operarios sobre un posible ataque y que estos puedan tomar las medidas pertinentes según sea el caso.

2.Marco Teórico

En esta sección se describen los conceptos más importantes en cada uno de los campos del conocimiento que se requieren para el desarrollo la propuesta. Los detalles relacionados con la implementación se explicarán en más detalle en las secciones posteriores.

2.1. Generalidades de los Sistemas SCADA

Los sistemas SCADA se implementan, en la mayoría de los casos, en ambientes industriales con el fin de centralizar la información proveniente de sensores, actuadores y controladores y de esta manera tener control sobre toda una planta de forma remota. Para que esto pueda ser posible se hacen necesarios varios elementos, que finalmente son los encargados de manipular, transformar y transportar la información de modo que los procesos puedan ejecutarse de forma automática. Adicionalmente, los sistemas se configuran con diferentes parámetros de operación definidos por el personal a cargo[10].

En lo que respecta a las disciplinas involucradas en la automatización industrial, se pueden mencionar: El control automático, que es la disciplina que permite el modelamiento de procesos y la programación de los mismos [11]. La instrumentación industrial, que es el área encargada de definir la correcta selección de los componentes, mientras la supervisión de datos es la rama encargada de la gestión de la información y los sistemas de alarmas [12][13]. La jerarquía de los procesos se modela tomando como referencia la pirámide de control, de la cual se hablará a continuación.

2.1.1. La pirámide de Control

La pirámide de Control, define la estructura jerárquica de los sistemas de gestión remota o SCADA. En la figura 2 y 3 [14] se describen las diferentes distribuciones que la componen, así como los diferentes elementos de control que se relacionan en cada una de ellas.

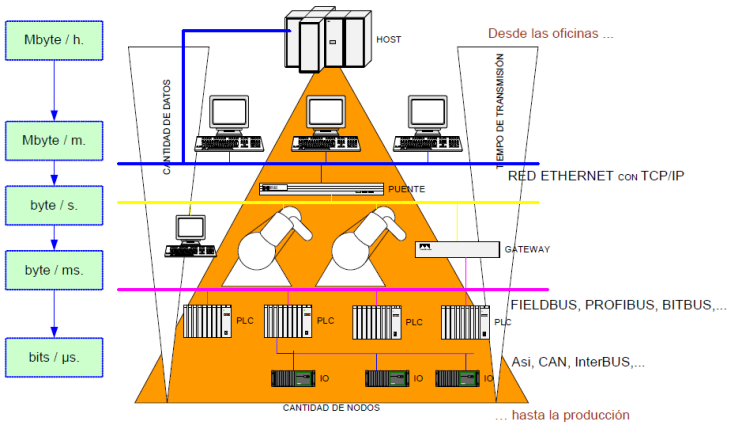


Figura 2.Pirámide de Control-Comunicaciones

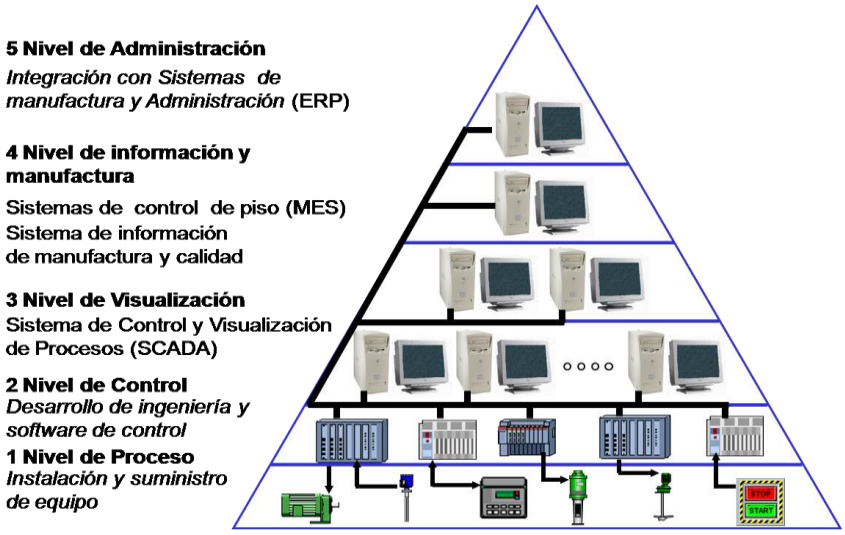


Figura 3.Pirámide de Control-Descripción Niveles

Como se puede observar en las figuras, los sistemas de control y visualización hacen parte del nivel 3, mientras que los dispositivos de control se encuentran ubicados en el nivel 2. El sistema propuesto en este trabajo, se instala entre los niveles 2 y 3 alertando sobre alguna posible anomalía en la operación normal de los procesos valiéndose del análisis de las distintas variables de los controladores involucrados. Es de notar, que en la medida que se aumenta de nivel, el volumen de información crece dramáticamente y por ende la velocidad de respuesta; un sistema de detección eficaz debe por tanto ubicarse en las partes bajas de modo que pueda alertar oportunamente ante una posible anomalía.

2.1.2. Esquema Básico de un Sistema SCADA

El esquema básico de un sistema SCADA se compone de los elementos que se describen en la figura 4.; allí se muestra el esquema de conexiones y la interacción que se dan entre los distintos elementos. Cada uno de los numerales que aparecen en la figura se describen a continuación:

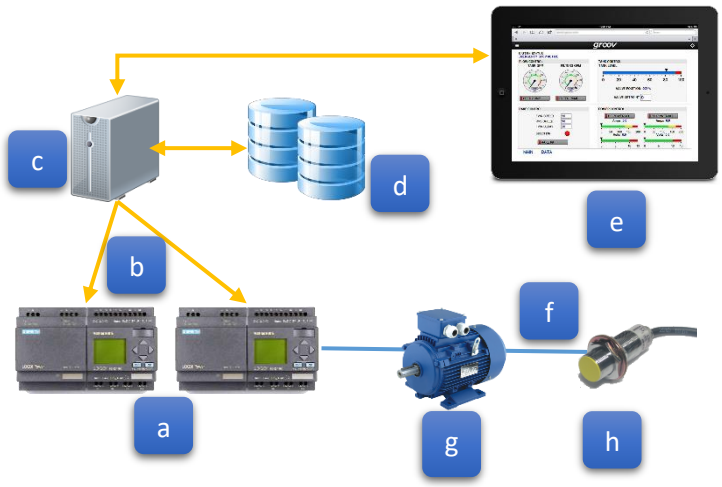


Figura 4.Esquema básico Sistema SCADA

a. Controladores(PLC): Son dispositivos programables que permiten la automatización de los procesos. Los PLC son los encargados de administrar la información proveniente de los sensores y al mismo tiempo ejecutar las acciones de control sobre los dispositivos actuadores; en los últimos años han sido dotados de gran versatilidad, lo que hace posible establecer comunicaciones avanzadas basadas en estándares como: Ethernet, RS485, RS232 entre otros.

b. Comunicaciones: Son aquellos cables y dispositivos que permiten la interacción entre los dispositivos de control y los servidores. Dependiendo de las necesidades o la tecnología utilizada se selecciona el tipo de protocolo de comunicación y los dispositivos de interfaz: cables, routers, entre otros. En los últimos años la mayoría de los automatismos vienen dotados con puerto Ethernet; dicho estándar facilita la interacción entre la etapa de automatización y la redes de información tradicionales.

c. Servidor SCADA: Es el sistema en donde se centraliza toda la información proveniente de los diferentes controladores asociados a una planta. Es un sistema de alta capacidad que interactúa con muchos dispositivos en tiempo real de forma tal, que los dispositivos tipo cliente puedan tener acceso a las diferentes variables de supervisión y al mismo tiempo puedan enviar comandos mediante las variables de control.

d. Bases de Datos: Aunque normalmente están embebidas dentro del servidor SCADA, son éstas las que permiten el almacenamiento de históricos de los diferentes procesos. Su papel es crucial a la hora de hacer análisis de eficiencia en los procesos y reportar posibles anomalías en la ejecución normal de los mismos.

e. Interface Hombre-Máquina(HMI): Se podría decir que son los dispositivos clientes en una red SCADA. A través de estos es posible interactuar con los diferentes procesos en forma visual y amigable para el operario. Se caracterizan por tener animaciones que modelan el comportamiento de un sin número de variables de control y observación, así como los diferentes paneles de alarma, en caso de avería o mal funcionamiento de los procesos.

f. Buses de Campo: Son redes de actuadores y sensores que permiten al dispositivo controlador realizar su gestión, sin la necesidad de implementar un gran número de cables o establecer conexiones punto a punto con cada uno de ellos. En pocas palabras, lo que se busca es simplificar las conexiones entre los dispositivos de sensado, control y acción; de tal suerte que sea mucho más fácil detectar posibles fallas mientras que se garantiza la escalabilidad de los procesos. Comparados con las redes de información, los buses de campo se caracterizan por tener volúmenes de información más pequeños y grandes velocidades de respuesta [15].

g. Dispositivos Actuadores: Son aquellos encargados de ejecutar las acciones de control en los diferentes procesos cuando el dispositivo maestro así lo determine. Dentro de la categoría de dispositivos actuadores podemos encontrar: Motores, válvulas, pistones, contactores, etc.

h. Dispositivos Sensores: Son los dispositivos encargados de convertir cualquier tipo de magnitud física de un proceso en una señal de voltaje o corriente equivalente. A través de ellos los controladores obtienen información del proceso para la toma de decisiones.

Dada la heterogeneidad de los sistemas SCADA y la diversidad de protocolos y dispositivos que pueden utilizarse en su implementación, a continuación se describe la utilidad de los sistemas SCADA basados en el estándar OPC. Este estándar constituye una pieza clave en el planteamiento y desarrollo del trabajo que aquí se expone.

2.2. Sistemas SCADA basados en OPC

2.2.1. Descripción General

Los sistemas SCADA, en términos generales, se podrían clasificar en dos grandes grupos: Los de tipo propietario, cuyas tecnologías y protocolos pertenecen a una firma de automatización específica y los de tipo abierto, cuyos tecnologías y protocolos pueden ser de diferentes firmas, pero aun así garantizan un aspecto muy importante llamado interoperabilidad. La interoperabilidad es la capacidad de poder enlazar en un mismo sistema dispositivos de diferentes firmas garantizando que estos puedan comunicarse entre sí. Para lograr dicho fin se creó OPC (OLE for process and Control), el cual consta de un enlace Cliente/Servidor; mediante este enlace el servidor se comunica con cualquier dispositivo que se encuentre registrado en la *OPC foundation*[16][17](PLC, HMI, controladores, etc). Por su lado, el cliente accede a la información obtenida por el servidor en un protocolo que se ha estandarizado y que permite que los datos de muchos dispositivos pueda ser centralizada en una sola interfaz.

2.2.2. Arquitectura OPC

Los servicios OPC dependiendo de su funcionalidad o características se pueden clasificar de la siguiente manera, tal y como lo ilustra la figura 6[18]:

a. Comunicación Cliente/Servidor OPC: En esta etapa el dispositivo cliente accede a la información contenida en el dispositivo servidor de forma que puede interactuar con el dispositivo final usando como interfaz el estándar OPC. Bajo esta configuración varios clientes pueden tener acceso a un mismo servidor.

b. Servicio OPC-Traducción de Datos: Es el encargado de traducir los diferentes protocolos de las distintas fuentes en un protocolo OPC estándar que pueda ser interpretado por el servidor OPC. En síntesis, sin la intervención de este tipo de servicio sería imposible garantizar la interoperabilidad que se mencionó al comienzo de esta sección.

c. Servicio OPC-Comunicación Fuente de Datos: Es el encargado de interactuar con los dispositivos finales (PLC, HMI, Controlador, etc), para la cual requiere de API´s que le permitan interpretar los diferentes protocolos. Debido al vasto número de protocolos y dispositivos industriales, los servidores cuentan con un número amplio de drivers proporcionados por las firmas asociadas, de modo que pueda establecer comunicación con la fuente primario y posteriormente ser traducida.

Figura 6. SCADA básico basado en OPC

Cada uno de los elementos descritos, permite que los servidores OPC sean aplicables a un número bastante grande de dispositivos; sin embargo, para cada tipo de aplicación se han especificado servidores OPC con diferentes características.

Hasta aquí se han descrito las generalidades del estándar OPC y sus servidores. En la siguiente sección se hará una descripción general del servidor OPC de la empresa National Instruments, mientras que se ilustra cómo establecer una conexión con un dispositivo PLC.

2.2.3. Servidor OPC National Instruments

La compañía estadounidense *National Instruments* fue fundada en el año 1976 por los doctores Dr. James Truchard y Jeff Kodosky[19] y ha sido durante años una de las empresas pioneras en el desarrollo de sistemas de instrumentación virtual mediante la implementación de su exitosa plataforma de programación gráfica llamada Labview.

En los últimos años, la compañía al igual que muchas otras empresas en el campo de la instrumentación y control, ha desarrollado su propio servidor OPC, el cual permita la integración de un gran número de dispositivos con el entorno de programación Labview. Esto permite que en un solo entorno se pueda acceder a las variables de un controlador o proceso, crear entornos amigables con el usuario mediante la utilización de comandos e indicadores estilizados y como si fuera poco realizar todo tipo de procesamiento a los datos recibidos entre los que se incluye: gráficas, operaciones matemáticas y lógicas, entre muchas otras.

La figura 6[20], muestra un sistema SCADA basado en OPC y que ha sido implementado en el software Labview.

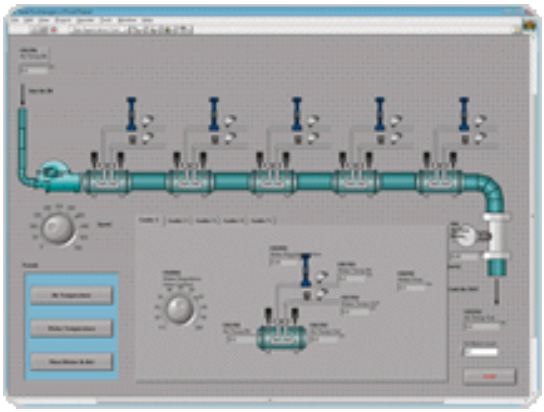


Figura 7. SCADA NI-OPC en Labview

2.3. Controladores Lógicos Programables (PLC)

Los controladores lógicos programables o PLC, por sus siglas en inglés, son sistemas electrónicos dotados con una amplia gama de periféricos y comunicaciones con el fin de automatizar procesos industriales, en donde las condiciones de operación son extremas (altos niveles de temperatura, humedad, ruido eléctrico, etc.). En sus inicios, se valían esencialmente de la activación, o desactivación de sistemas lógicos, operaciones de conteo y temporización; sin embargo, hoy por hoy dichos sistemas han evolucionado, hasta el punto que pueden ejecutar cálculos complejos y establecer comunicaciones en diferentes capas físicas y protocolos.

2.3.1. Tipos de Controladores

En la inmensa gama de dispositivos PLC, se pueden encontrar básicamente 2 tipos de categorías: La primera, es la de los PLC compactos, los cuales se caracterizan por tener pocos puertos de entrada y salida, memoria reducida y, sobre todo, la incapacidad de soportar varios módulos de expansión. En esta categoría, se pueden encontrar controladores como el controlador LOGO de la empresa SIEMENS, cuya apariencia se puede observar en la figura 7[21].

Figura 7. Controlador Compacto de la Empresa SIEMENS

La segunda categoría, es la que más aplicación tiene en entornos industriales y corresponde a los PLC modulares. Esto PLC, a diferencia de los compactos, se caracterizan por admitir muchos módulos de expansión, lo cual, facilita su escalabilidad. Entre otras características, se pueden destacar la estabilidad y confiablidad de dichos controladores, ya que son estos los encargados de gestionar la operación automática de muchos procesos críticos, como son: Centrales hidroeléctricas, industrias químicas, alimentos, entre otras. En la figura 8 [22], se puede observar la apariencia del PLC S7-1200, y su capacidad de expansión.

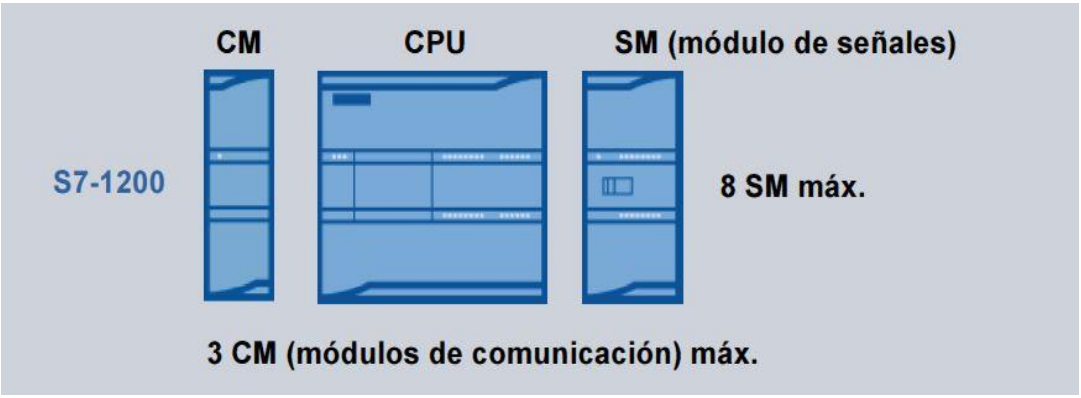


Figura 8. PLC Modular S7-1200 de la Empresa Siemens

En lo que respecta a los módulos de expansión, esto se clasifican como: módulos de entradas digitales, módulos de salidas digitales, módulos de comunicaciones y módulos análogos ya sea de entrada o salida. Los módulos de entradas, son los

encargados de capturar el estado de sensores del tipo ON/OFF. Se utilizan en gran medida para los eventos de arranque o parada de un proceso, ya que son, en la mayoría de los casos, botones los encargados de suministrar información al controlador. Los módulos de salida digitales por su parte, son los responsables de activar o desactivar los dispositivos actuadores. Son bastante útiles para el control de cargas que no requieren un control sofisticado, sino simplemente activarse o desactivarse en determinadas circunstancias. En los últimos años, este tipo de módulos se les ha dotado de nuevas características, como la modulación por ancho de pulso para el control de potencia en cargas mediante la variación del ciclo de dureza en una señal de voltaje periódica o generación de pulsos con una frecuencia determinada.

En lo que respecta a los módulos análogos, es por medio de estos que es posible convertir un amplio rango de variables físicas, en un valor numérico. En síntesis, cualquier señal analógica de voltaje o corriente que representa el estado de un variable física (Temperatura, humedad, presión, etc.) es posteriormente procesada internamente de modo que el controlador pueda tomar decisiones. En el caso de los módulos análogos de salida, se tiene el mismo proceso en sentido contrario, ya que ahora, un valor numérico dado, se convierte en un voltaje o corriente representativa.

Finalmente, los módulos de comunicaciones, permiten el envío de información desde y hasta el PLC. En ese orden de ideas, cada módulo puede operar bajo un determinado número de protocolos y capa física. Es así, que se pueden anexar módulos que transmitan información en forma alámbrica e inalámbrica, con diferentes formatos y velocidades, de modo que otros dispositivos que los soporten pueden interactuar de forma remota con el PLC. Para la implementación de sistemas SCADA robustos, es clave la existencia de buenos módulos de comunicación en los controladores, de modo que se puedan atender eventos de forma remota en procesos donde la cantidad de variables que se observan o controlan es relativamente alta.

2.3.2. Tipos de Variables en Controladores PLC

Uno de los elementos claves para entender cómo podría llevarse a cabo un ataque a un sistema SCADA, y en particular, a los dispositivos PLC, es entender qué tipo de variables básicas son manejadas por éstos y las restricciones que pueden presentar a la hora de tratar de ser manipuladas de forma remota ante un posible ciberataque.

En general, los controladores PLC manejan 3 tipos de variable. y al mismo tiempo, a cada tipo de variable, le corresponden varios tipos de datos. En este caso no se hará una mención muy detallada de los tipos de datos; sin embargo, conviene entender algunas nociones básicas que permiten inferir bajo qué circunstancias una manipulación indebida de las diferentes variables y formatos podría llevar al malfuncionamiento del autómata.

Los 3 tipos de variables, que se pueden encontrar en todo dispositivo PLC son:

a. **Variables de Entrada:** Son aquellas variables que tienen conexión directa a los puertos físicos de entrada en un controlador. Su naturaleza puede ser análoga o digital, según sea el caso. Dicho tipo de variables no puede ser manipulada de forma remota ya que su estado depende directamente de las lecturas obtenidas por el puerto respectivo; no obstante, si se manipula indebidamente los dispositivos conectados a ellas, se podría inducir en el PLC una mala operación.

b. **Variables de Salida:** Son aquellas variables que tienen incidencia directa sobre los dispositivos actuadores que se encuentran enlazados con el PLC de forma física o virtual. Este tipo de variables, a diferencia de las entradas, si puede ser manipulada de forma remota; lo cual puede ser potencialmente riesgoso ante un eventual ciberataque a un proceso industrial.

c. **Variables Internas o Marcas:** Son aquellas variables internas del PLC que le permiten a este realizar operaciones de todo tipo o compartir información de forma virtual con otro tipo de dispositivos o sistemas. Dichas variables son clave en la operación normal de un dispositivo PLC; no obstante, su comportamiento, en la mayoría de los casos, es invisible a menos de que su valor sea visualizado para ser manipulado o vigilado por un operario o centro de gestión. Es clave resaltar que dichas variables, al igual que las salidas, puede ser manipuladas remotamente. Esta posibilidad crea un escenario terriblemente peligroso, debido a que, en teoría, se podría manipular el comportamiento de un dispositivo PLC, sin que los sistemas de monitoreo y alarmas reportaran ninguna clase de incidente. Este tipo particular de ataque, es uno de los que se pretende detectar mediante el sistema IDS al que apunta este trabajo.

La tabla 1[22] resume los tipos de datos; se puede apreciar que el tipo de dato más elemental es de 1 bit, mientras que existen otro tipo de formatos que se agrupan 1, 2, 4 y hasta 8 bytes.

Tipo de datos	Tamaño (bits)	Rango	Ejemplos de entrada de constantes
Bool	1	0 a 1	TRUE, FALSE, 0, 1
Byte	8	16#00 a 16#FF	16#12, 16#AB
Word	16	16#0000 a 16#FFFF	16#ABCD, 16#0001
DWord	32	16#00000000 a 16#FFFFFFFF	16#02468ACE
Char	8	16#00 a 16#FF	'A', 't', '@'
Sint	8	128 a 127	123, -123
Int	16	32.768 a 32.767	123, -123
Dint	32	-2.147.483.648 a 2.147.483.647	123, -123
USInt	8	0 a 255	123
UInt	16	0 a 65.535	123
UDInt	32	0 a 4.294.967.295	123
Real	32	+/-1,18 x 10^{-38} a +/-3,40 x 10^{38}	123,456, -3,4, -1,2E+12, 3,4E-3
LReal	64	+/-2,23 x 10^{-308} a +/-1,79 x 10^{308}	12345.123456789 -1,2E+40
Time	32	T#-24d_20h_31m_23s_648ms a T#24d_20h_31m_23s_647ms Almacenado como: -2,147,483,648 ms a +2,147,483,647 ms	T#5m_30s 5#-2d T#1d_2h_15m_30x_45ms
String	Variable	0 a 254 caracteres en tamaño de byte	'ABC'

Tabla 1. Tipos de Datos Comunes en Dispositivos PLC

Cuando se realiza un programa para un dispositivo PLC, se debe ser precavido a la hora de distribuir la memoria en los distintos tipos, ya que de no ser así se podría presentar un problema de solapamiento de memoria, donde varias variables comparten de forma total o parcial algunas regiones de memoria.

Hasta aquí, se han descrito, a grandes rasgos varios de los principales elementos de un sistema SCADA, por lo que a continuación se explicarán algunas nociones de instrumentación virtual y los sistemas Cliente ligados a los sistemas SCADA.

2.4. Instrumentación Virtual en Labview

El software de programación Labview, como ya mencionó en secciones anteriores consiste en un entorno de programación gráfico que se compone de 2 interfaces llamados: Panel de Control y Diagrama de Bloques como se observa en la figura 9. En la interfaz de diagrama de bloques, el programar ubica todos los elementos de control y visualización, como son: botones, sensores, perillas, graficadores entre otras, mientras que en la interfaz de diagrama de bloques se realiza la interconexión de los módulos y toda la programación asociada a los mismos.

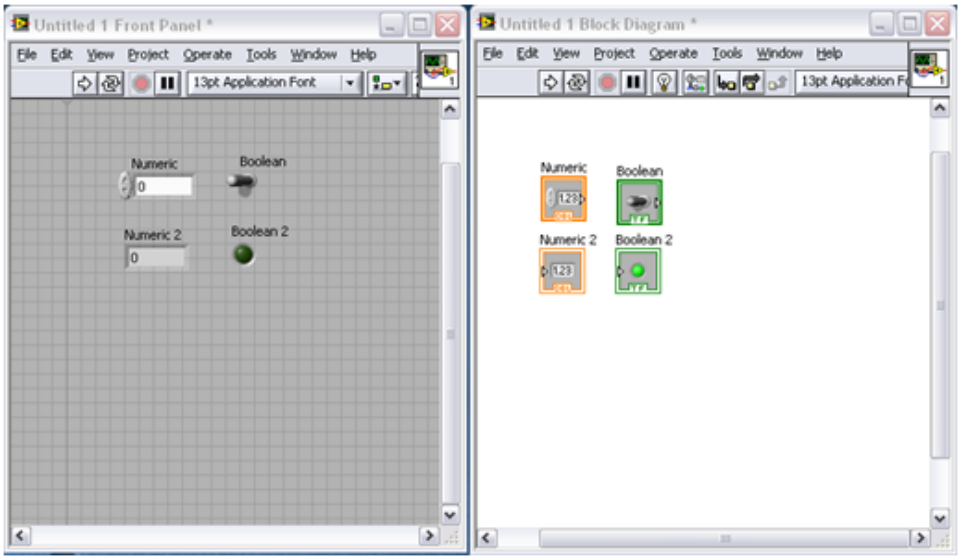


Figura 9. Panel de Control y Diagrama de Bloques Labview

Sin lugar a dudas, este potente entorno de programación gráfica, tiene funcionalidades que van más allá de la instrumentación virtual, permitiendo que en una sola interfaz puede ser implementados todos los componentes necesarios para el desarrollo del detector de intrusos propuesto. En particular, mediante el software Labview es posible la configuración de una interfaz cliente que se enlaza con el servidor OPC, permitiendo que, a través de esta funcionalidad, sea posible no solo la interfaz detectora sino también implementar una aplicación intrusa cuyo objetivo sea tratar de alterar el funcionamiento normal de un proceso determinado.

2.5. Ciberataques a sistemas SCADA

Los sistemas SCADA, al igual que los sistemas informáticos tradicionales, se han convertido en el blanco de todo tipo de ataques cibernéticos que pueden comprometer seriamente la infraestructura crítica de un país, comprometer la productividad de muchas empresas o en el peor de los casos, causar la pérdida de vidas humanas. En la mayoría de los casos, como se verá en el siguiente apartado, los ataques tradicionales a los cuales se ve sometido un sistema informático tradicional aplican para los sistemas SCADA; sin embargo, las repercusiones que acarrea el ataque a un sistema de control en una planta industrial tienen efectos más mediáticos sobre el rendimiento de los sistemas productivos.

De acuerdo con las estadísticas, el incremento en el número de ataques paso de 91,676 en enero de 2012 a 675,186 en enero del año del 2014[23]; esto sin contar el posible de número de ciberataques que no son reportados por las distintas industrias. Por otro lado,

expertos avisan que la tendencia seguirá en aumento, por lo que es clave la implementación de contramedidas.

A continuación, se describen algunos tipos de ataques que pueden alterar el correcto funcionamiento de un sistema SCADA, para finalmente plantear el posible escenario que se pretende detectar mediante el trabajo que aquí se desarrolla.

2.5.1. Tipos de ataque más comunes en sistemas SCADA

En el mundo de los sistemas SCADA, son muchos los tipos de ataques que podrían considerarse. Los más comunes, de acuerdo con expertos en la materia son[24]:

a. Ataques de Criptografía: Muchos sistemas SCADA de prestigiosas marcas carecían de criptografía a la hora de establecer comunicación entre las distintas terminales. Esto, sin lugar a duda, ha sido explotado en diversas ocasiones para obtener las contraseñas de validación de muchos dispositivos PLC y de esa manera tener acceso a la manipulación indebida de los distintos parámetros de configuración y programa [25]. En la figura 10[24], se muestra la captura de la contraseña para un PLC marca Omron, utilizando la herramienta Wireshark.

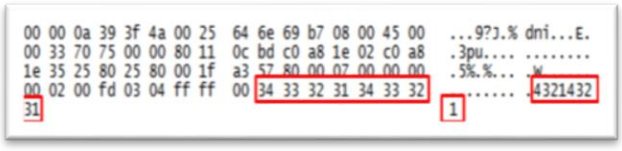


Figura 10. Captura de Contraseña de Seguridad PLC marca Omron

b. Ataque de Corrupción de Paquetes: En este tipo de ataques, los datos que van desde y hacia los controladores, son modificados de manera sutil de tal forma que se altera el correcto funcionamiento de los procesos o se engaña a los operarios reportando información falsa. Sin lugar a dudas, es uno de los tipos de ataques más peligrosos ya que pasa desapercibido para los sistemas de detección tradicionales como: antivirus, firewall, entre otros.

c. Ataques de Fragmentación [24]: Para desestabilizar el funcionamiento del PLC, o simplemente hacer que éste deje de responder a los demás sistemas que se encuentran enlazados, se utilizan ataques de fragmentación; en donde, paquetes malformados, es decir, que no respectan la estructura normal establecida en el protocolo de comunicaciones, son enviados a propósito al controlador. Dicho tipo de ataques, a diferencia de los mencionados anteriormente, si podrían

eventualmente ser detectados por los sistemas tradicionales, dado que la estructura de las comunicaciones es bastante similar en las redes de control y de información.

d. **Ataques de Denegación de Servicio:** En este tipo de ataque, los diferentes componentes de un sistema SCADA, son bombardeados con un inmenso número de paquetes en un corto periodo de tiempo. La consecuencia de esto, es que los dispositivos son incapaces de responder y, por ende, todos los servicios asociados se colapsan.

2.5.2. Escenarios de Ataques a Sistemas SCADA

Una vez se han descrito algunas generalidades de los sistemas SCADA y los posibles tipos de ataque de los que podrían ser víctimas, conviene ilustrar 3 tipos de escenarios que la solución que se describirá en este trabajo, puede detectar. En la figura 11, se muestra la ubicación de atacante, mientras se describen las posibles consecuencias que dicho ataque podría acarrear.

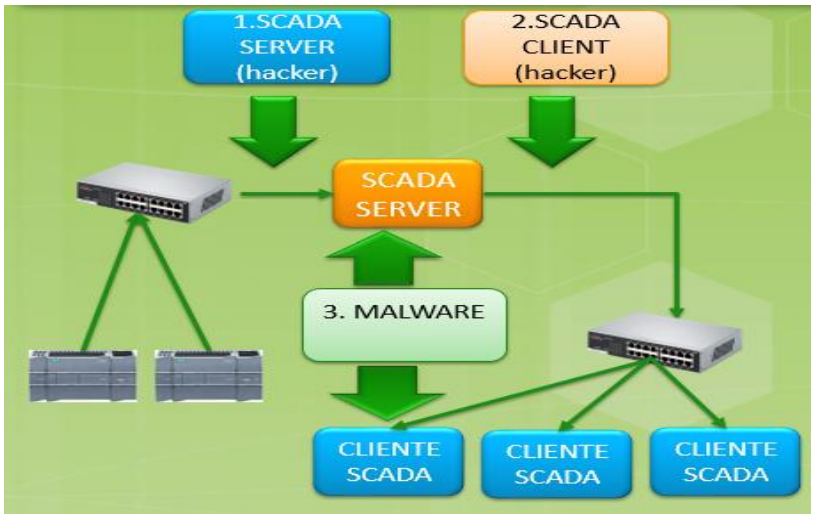


Figura 11. Esquema simplificado de un Sistema SCADA y posibles Amenazas

El primer escenario un intruso suplanta la identidad del servidor, de modo que pueda manipular de forma indebida los parámetros de los controladores o incluso alterar el verdadero valor de las variables observadas antes de que estas lleguen al servidor SCADA. Este tipo de ataque se le conoce en inglés como “Man in the middle” y puede ser potencialmente dañino para la correcta operación de un sistema SCADA.

En el segundo escenario, un intruso toma el control de una plataforma Cliente y por medio de esta manipula de forma indebida los parámetros de control; además el intruso puede filtrar información proveniente de las variables observadas. En comparación con el primer escenario la posibilidad de que un ataque como este se efectúe es mucho mayor dado que la mayoría de las herramientas de seguridad priorizan el servidor.

En el tercer escenario, un "malware" infecta la plataforma cliente o servidor y desde allí manipula de forma indebida los parámetros de control u observación. Este escenario es uno de los más peligrosos, ya que el ataque se efectúa de forma silenciosa y es probable que los mecanismos tradicionales no lo puedan detectar. Si se considera, además, que muchas industrias en Latinoamérica no poseen las últimas actualizaciones de software para sus respectivas plataformas de computación, la probabilidad de una infección de este tipo es supremamente alta.

El sistema que se describe en este trabajo, tiene la capacidad de detectar ataques en los 3 tipos de escenarios mostrados; no obstante, su principal fortaleza está en la detección de ataques tipo "Malware" donde la variación de parámetros puede llegar a ser imperceptible para los sistemas de alerta tradicionales de un sistema SCADA.

2.6. Sistemas de Detección de Intrusos

Los sistemas para la detección de intrusos o IDS, son sistemas encargados de analizar el tráfico, comportamiento de una red o dispositivo con el fin de detectar patrones anómalos, mientras reporta los posibles incidentes.

Los IDS se pueden clasificar en 2 tipos según el lugar en donde se realizará la detección. Los N-IDS son sistemas que permiten examinar el comportamiento anómalo de una red o al menos parte de ella, mientras que los H-IDS permiten evaluar patrones no deseados dentro de un dispositivo final o Host [26]. Para el caso específico de los sistemas SCADA ambas aproximaciones podrían ser posibles; en donde los N-IDS permiten el monitoreo de cada uno de los protocolos de comunicación entre las unidades de control mientras que los H-IDS permiten la evaluación de las variables del proceso dentro del controlador.

Dependiendo de su configuración y el tipo de sistemas que busca monitorear, los sistemas IDS clásicos utilizan diversas técnicas que permiten detectar posibles ciberataques. La primera técnica para la detección de intrusos consiste en detectar paquetes corruptos o en otras palabras, paquetes con información dañina para los componentes de la red. Para la aplicación de esta técnica el sistema debe reconocer el tipo de protocolo utilizado y de esa manera verificar que cada uno de los campos se encuentre dentro de sus rangos normales. Dicha técnica podría ser bastante útil en sistemas SCADA en donde el número de protocolos implementado no sea muy grande permitiendo así que ninguna de las variables del sistema esté por fuera de su rango normal.

La segunda técnica para la detección de intrusos consiste en la identificación de patrones de ataque los cuales se caracterizan por tener un comportamiento no regular. En este caso, el IDS analiza el comportamiento de una red específica al monitorear la ruta de los paquetes y sus correspondientes emisor y receptor. Si eventualmente un paquete o transacción no se comporta de la manera deseada, este se clasifica como un posible ataque hasta la respectiva revisión. Si el paquete registrado efectivamente corresponde a un ataque el sistema IDS se entrena de tal manera que pueda identificar ataques similares. En el caso particular de los sistemas SCADA esta aproximación se puede dar desde 2 frentes independientes o la combinación de los mismos. En el primer caso, el sistema IDS analiza únicamente el comportamiento de las variables de un sistema y con base en ellas podría detectar una eventual amenaza. En el segundo por su parte, se analizan los paquetes de red encargados de controlar el proceso y se verifica que estos se encuentren dentro los rangos deseados. La combinación de ambas aproximaciones es más efectiva ya que puede discernir si el comportamiento anómalo de la planta se debe a mal funcionamiento de la misma o un ataque cibernético [27].

Una aproximación al mundo de los IDS, dirigida específicamente a los sistemas SCADA se encuentra en el artículo de Idaho SCADA-IDS [28], en donde se definen las características que un sistema de detección de intrusos debe poseer dentro de una red de control industrial. De acuerdo con este documento, el sistema debe, en primer lugar, identificar muy bien el tipo de protocolo implementado en la red SCADA de manera que pueda dar cumplimiento con una segunda característica muy importante que es la penetración de paquetes. En esta etapa, el sistema es capaz de detectar información corrupta que ponga en riesgo la planta monitoreada. La tercera característica que se debe cumplir es la identificación de patrones, de modo el sistema IDS tenga la capacidad de adaptarse al comportamiento de la red SCADA y detectar cuando se presente un comportamiento anormal y como última característica este debe ser capaz de identificar estados críticos en la planta como producto de paquetes malintencionados.

Hasta aquí se ha hecho una introducción a cada uno de los tópicos de interés relacionados con sistemas SCADA. En el próximo apartado se describirá en qué consisten los métodos de aprendizaje automático para luego abordar cada uno de los trabajos realizados en el mundo de los IDS-SCADA.

2.7. Algoritmos de Aprendizaje Automático

Los algoritmos de aprendizaje automático son una ramificación de la inteligencia artificial, los cuales tienen como propósito permitir que los sistemas computacionales puedan “aprender”. En términos generales se puede decir que dichos algoritmos tienen la capacidad de reconocer patrones dependiendo de la forma como han sido configurados o gracias a un entrenamiento previo utilizando muestras de ejemplo [20].

2.7.1. Tipos de Algoritmos

Los algoritmos de aprendizaje automático pueden clasificarse en 2 grandes grupos: Los del tipo supervisado y los no supervisados.

Los algoritmos de aprendizaje supervisado son aquellos que requieren ser entrenados con un determinado tipo de muestras, de forma que, posteriormente, puedan clasificar un determinado tipo de dato dentro de las categorías que previamente se habían definido. Bajo este paradigma, existen, por ejemplo, los llamados algoritmos biclase, los cuales requieren 2 tipos de muestra para su entrenamiento de modo que al analizar un nuevo dato se le pueda asignar una categoría dependiendo del patrón de similitud empleado. Usualmente se usan etiquetas como 1 o -1 para identificar la clase a la cual el dato pertenece.

Es importante resaltar, que dentro de los algoritmos del tipo supervisado existen algunos que solo requieren un tipo de muestra para su entrenamiento por lo que se les llama Algoritmos “One Class” o de una clase por su traducción al español. Este tipo de algoritmos poseen la capacidad de aprender el patrón regular de un conjunto de datos en la etapa de entrenamiento, para posteriormente clasificar un nuevo dato indicando si este se ajusta al patrón previamente definido o sí muestra un comportamiento atípico. A este tipo de datos se les conoce como “Outliers”[3] o atípicos.

Los algoritmos de aprendizaje del tipo no supervisado, por otra parte, son aquellos que no requieren una etapa previa de entrenamiento, sino que basados en ciertos criterios de similitud, agrupan los datos en un determinado número de conjuntos que bien puede ser definido previamente o calculado automáticamente [30]. El criterio de similitud puede variar dependiendo de la naturaleza de los datos analizados, siendo uno de los casos más simples la distancia euclidiana que hay entre los datos en mención.

2.7.2. Consideraciones Generales al trabajar con Algoritmos de aprendizaje automático

a. Definición de descriptores

En esta primera etapa, el objetivo es encontrar las características que mejor describen un determinado fenómeno u objeto, de forma que, adoptando una naturaleza numérica, puedan ser ingresadas en un algoritmo de aprendizaje automático con fines de entrenamiento, agrupación o clasificación.

Como criterio general, se debe garantizar que los descriptores sean linealmente independientes entre sí, y que además sus valores cambien significativamente al ser comparados con los de otro ente de naturaleza similar.

b. Escalamiento de los descriptores

Con el fin de garantizar, que los datos que serán ingresados en un algoritmo de aprendizaje tengan rangos similares, previa a la operación de entrenamiento estos se deben escalar con respecto a sus valores máximo y mínimo. Esto debido a que, en un conjunto de datos, si los rangos de los mismos son bastante diferentes, no será posible establecer criterios de similitud, o simplemente no permitirá la convergencia adecuada de las funciones objetivo para determinados algoritmos de aprendizaje. Una de las metodologías más empleadas en el escalamiento de descriptores, se puede ver en la Ecuación 1.

$$x' = \frac{x - min(x)}{max(x) - min(x)}$$

Ecuación 1. Escalamiento de Descriptores

c. Entrenamiento

En esta etapa, el algoritmo recibe los descriptores previamente normalizados o escalados, con el fin de obtener un modelo que permita clasificar los datos que entrarán posteriormente. En este sentido, se puede hablar de 2 tipos de metodología de entrenamiento: *Batch* y *Online* [31].

El método *Batch* requiere un conjunto completo de muestras que describen un determinado objeto de forma tal que el sistema es previamente entrenado y ajustado antes de ser usado en la clasificación. En este tipo de metodología, el sistema entrenado no tiene la posibilidad de evolucionar ya que para esto requiere un nuevo entrenamiento fuera de línea.

El método Online por su parte, propone un modelo de entrenamiento constante en donde el sistema a la par de su tarea de clasificar, reconfigura sus parámetros en el tiempo, evolucionando y adaptándose a las nuevas condiciones que se plantean.

d. Problemas de Entrenamiento

En el proceso de entrenamiento, existe la posibilidad de que un determinado algoritmo se ajuste completamente a los datos de entrenamiento de tal forma que no pueda generalizar su clasificación a otro tipo de datos diferentes a los usados en esta etapa. A este fenómeno se le conoce como sobre-entrenamiento o “Overfiting” en inglés [32].

Para evitarlo, existen varias metodologías en donde lo que se busca es crear subconjuntos en el banco de datos de entrenamiento y de esa manera tener datos de entrenamiento y de prueba. En primer lugar, se entrena el algoritmo con el banco de entrenamiento y una vez se ha hecho esto, se procede a calcular la eficiencia de la clasificación utilizando el bando de prueba.

e. Validación

Como ya se mencionó, en todo proceso de entrenamiento de un algoritmo de aprendizaje de máquinas, es supremamente importante validar los resultados obtenidos de tal forma que se garantice la mayor eficiencia posible en el proceso de clasificación y que dicho comportamiento sea lo más universal posible; es decir, que el algoritmo de clasificación no solo clasifique correctamente los datos que le son conocidos sino también nuevos tipos de datos que muestren comportamientos similares o no.

Una estrategia muy utilizada consiste en separar el conjunto de datos disponibles en 2 grupos; en donde, el primer grupo representado por un 70% de los datos se utiliza para el entrenamiento del algoritmo mientras que el 30% restante se usa para la validación del mismo.

Otro tipo de estrategia que se usa comúnmente para la validación de algoritmos de clasificación se conoce como “Cross-validation” o validación cruzada [3]. En esta estrategia, el número de datos se divide en varios grupos, con el fin de utilizar uno de los grupos en la etapa de validación y el resto en la etapa de entrenamiento. Se hace el mismo proceso con cada uno de los grupos conformados y se espera que el patrón de comportamiento para cada uno de los escenarios sea similar. En caso de no ser así se concluye que el modelo planteado no es universal y por tanto deben reconfigurarse los parámetros de entrenamiento.

2.7.3. Algoritmos OCSVM

Las máquinas de soporte vectorial del tipo “One Class” representan un caso especial de la clasificación biclase, en donde el algoritmo de clasificación solo se entrena con un tipo de datos, permitiendo la posterior detección de patrones atípicos que no se ajustan al modelo previamente establecido.

Por defecto, las máquinas de soporte vectorial son entrenadas por 2 tipos de muestras en donde a cada una de ellas se les asigna una etiqueta para su posterior identificación. Típicamente se asignan un 1 o -1 para diferenciar las clases; sin embargo, en las OCSVM el entrenamiento solo se realiza con un tipo de datos que bien podría ser identificado con un 1. Ya en la etapa de clasificación, el comportamiento de ambas metodologías es prácticamente el mismo, con la diferencia de que para las SVM tradicionales las etiquetas arrojadas en la clasificación corresponden a las clases previamente definidas mientras que para la OCSVM un 1 representa que el nuevo

dato se ajusta al modelo proporcionado o -1 que dicho dato tiene un comportamiento diferente al esperado. El hecho que las OCSVM puedan ser entrenadas con un solo tipo de datos, permite que puedan ser utilizadas ampliamente en la detección de atípicos o "Outliers" como, por ejemplo, en la detección de posibles manipulaciones indebidas a las variables de un sistema SCADA [3].

Al igual que en las SVM tradicionales, el algoritmo calcula la frontera de decisión basado en algunos datos llamados vectores de soporte. Para el caso de la OCSVM, la frontera es calculada utilizando únicamente un tipo de datos tal y como lo muestra la figura 2.

El algoritmo OCSVM primero mapea los datos de entrada en una escala dimensional más alta mediante una función Kernel y luego de forma iterativa encuentra el plano de separación marginal que permita la mejor separación de los datos con respecto al origen. De esta manera, el hiperplano o frontera de decisión, se describe mediante la función de clasificación mostrada en la ecuación 2:

$$f(x) =< x, w > +b$$

Ecuación 2. Ecuación Clasificación SVM

En donde el parámetro w es el vector normal y b el offset de la función. Mientras la función f(x) sea mayor que cero el dato será clasificado como normal; en caso contrario se clasificará como "outlier". Un esquema de la frontera de decisión, basada en los vectores de soporte elegidos se muestra en la figura 12.

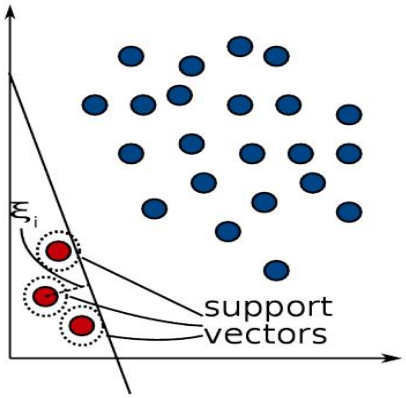


Figura 12.Vectores de soporte en clasificación OCSVM para determinar la frontera de decisión.

a. Funciones Kernel

En lo que respecta a la función Kernel generalmente se utilizan 4 tipos dependiendo del tipo de separación que se quiera lograr. Los tipos de Kernel más comunes son: Lineal, polinómico, sigmoide y Gaussiano con base Radial o (RBF)[2]. Para cumplir con el objetivo propuesto en este trabajo se utilizará el Kernel del tipo RBF, el cual

se rige por la ecuación 3, ya que para las OCSVM es el más común y está soportado en la librería LIBSVM [33] de la cual se hablará más adelante.

$$K(x_i, x_j) = \exp(-\sigma||x_i - x_j||^2)$$

Ecuación 3. Ecuación Kernel RBF

b. Parámetros de la OCSVM

Para la configuración del clasificador OCSVM es necesaria la configuración de 2 parámetros importantes. Uno de ellos se conoce como *n* y es el que define qué tan ajustada se encuentra la frontera de decisión con respecto a los vectores de soporte. Generalmente este parámetro se encuentra entre 0 y 1. El otro parámetro es σ y es el encargado de determinar qué tan ancha es la campana Gaussiana en el Kernel RBF. Mientras más pequeño sea σ mayor será el ancho de la campana y viceversa, por tanto es necesario garantizar que este valor no sea muy grande ni muy pequeño de tal suerte que el sistema no quede sobreentrenado o la eficiencia de la clasificación sea muy baja [34].

c. Librería LIBSVM

La librería conocida como LIBSVM es una poderosa herramienta que permita la implementación de máquinas de soporte vectorial en distintos ambientes de programación (Labview, python, java, etc) con la ventaja de que puede ser configurada para operar como una OCSVM. En la figura 13 se muestran las diferentes herramientas de las que dispone la librería una vez instalada en el software Labview.

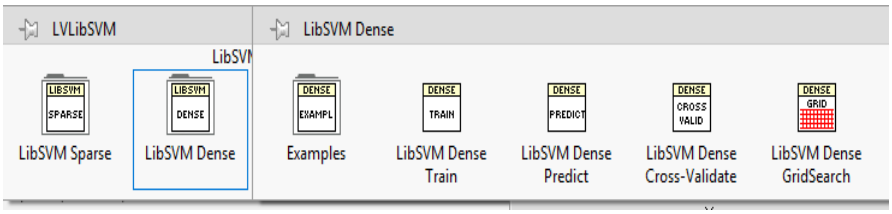


Figura 13. Toolkit LIBSVM en Labview

Hasta aquí se han abordado todos los conceptos relevantes para el desarrollo de este trabajo. A continuación, se hará un análisis de todos los trabajos relacionados para luego destacar, las ventajas de este trabajo con respecto al desarrollado por los pares.

3.Estado del Arte

En comparación con otras líneas de investigación, los trabajos relacionados con el mundo de la ciberseguridad asociada a los sistemas SCADA no son muy abundantes; sin embargo, y con el fin de delimitar el alcance de este proyecto, se ha hecho un análisis exploratorio inicial para luego analizar en mayor detalle los trabajos relacionados con el mundo de los IDS en entornos industriales. En particular, se dio especial énfasis a aquellos que abordan metodologías adaptativas o que incluyeron algoritmos de aprendizaje de máquinas.

3.1. Exploración Inicial

Como ya se ha mencionado, el tema de la ciberseguridad aplicado a los sistemas SCADA es relativamente nuevo si se compara con su contraparte en los sistemas de información.

Un documento regulatorio de los Estados Unidos, llamado NIST [35] y cuya versión más reciente data del año 2014, define básicamente 5 puntos básicos que deben ser tenidos en cuenta a la hora de diseñar un marco de seguridad para tecnologías cibernéticas.

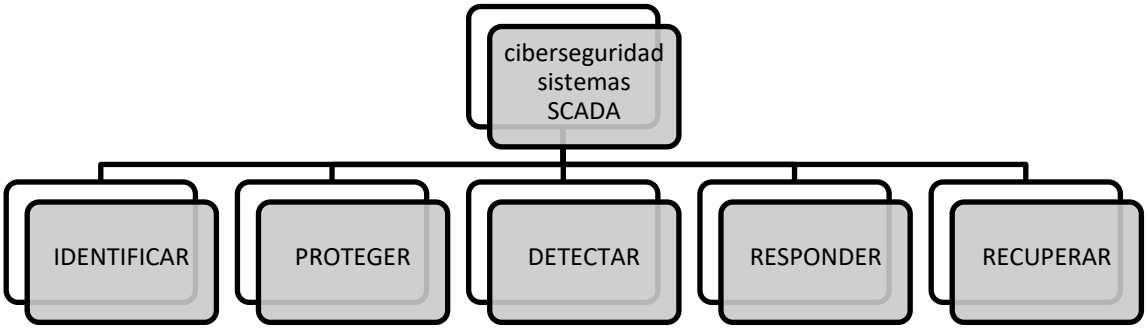


Figura 14. Esquema Ciberseguridad Sistemas SCADA

Tal como lo muestra la figura 14, en primer lugar, aparece la etapa de la identificación, que hace referencia a la búsqueda de posibles peligros o vulnerabilidades presentes en las redes SCADA. En este contexto se hallan interesantes trabajos como el de Paukatong, T.[36] en donde se describen los posibles ataques de los cuales puede ser víctima una red de distribución de energía. En particular, se describen 2 posibles ataques a redes SCADA que son: ARP spoofing y SQL attack. Entre otros trabajos interesantes que avanzan en esta misma línea se puede observar un documento extenso de la división de seguridad de

los Estados Unidos [37], en el cual se mencionan los ataques más comunes de los cuales puede ser víctima una red SCADA. En este último documento se añaden otros como "Man in the middle"; ataque que, dadas sus características, es potencialmente dañino en ámbitos industriales. Sumado a lo anterior, Ting. Wang[38] describe el desarrollo de una herramienta de fuzzing para identificar posibles anomalías o vulnerabilidades en sistemas SCADA basados en el estándar OPC. Los resultados obtenidos se enfocan en la falta de encriptación de dicho estándar en sus versiones iniciales, así como las dificultades que pueden presentarse en el módulo de comunicaciones gestionado desde el sistema operativo Windows.

En segundo lugar, se tiene *la etapa de protección* en donde evalúan las iniciativas que buscan mejorar las medidas de protección de los sistemas SCADA. Bajo esa pretensión se han desarrollado interesantes trabajos como el de Disso J.P[39], en el que se describe la utilización de honeypots como medida de protección para redes industriales; el trabajo de Sommestad, T.[40], en el que basados en estadísticas muestran los niveles de protección de los actuales sistemas SCADA en comparación con las normativas vigentes al respecto.

En *la etapa de detección* se evalúan las posibles herramientas o medidas que se pueden considerar a la hora de detectar posibles ataques o intervenciones maliciosas dentro de la red. Es aquí donde cobran fuerza conceptos como seguridad perimetral y en particular los IDS o sistemas para detección de intrusos por sus siglas en inglés. Dado que este ítem hace referencia directa a la definición del problema, los trabajos relacionados en esta área se analizarán con más profundidad en apartados posteriores.

Como cuarta y quinta aparecen Recuperar *y Responder*, y es en estos dos tópicos donde se especifican los protocolos o medidas a seguir en caso de que un ciberataque haya tenido éxito. Para estos ítems en particular, no se encontraron trabajos de tipo académico pues son las normas internacionales quienes determinan qué hacer en esos casos.

3.2. Antecedentes

Para el problema de la detección de intrusos dentro de los sistemas SCADA se han planteado diversas soluciones cumpliendo en mayor o menor medida con las características que se supone debe cumplir una solución que combina IDS-SCADA.

En primer lugar, se describe el trabajo de Yang [42] que utiliza un IDS llamado Snort para la detección de posibles incidentes de seguridad. Snort es reconocido como un sistema libre que es usado con regularidad en las redes informáticas tradicionales para la detección de intrusos; sin embargo, considerando que muchos protocolos industriales trabajan en la misma capa física que las redes informáticas, esto es Ethernet, es posible implementar Snort para la detección en SCADA. Snort es programado con un conjunto de reglas, las cuales al ser violadas generan alertas. Esta configuración tiene como

problema que el sistema sigue siendo susceptible a ataques que no hayan sido pre-programados.

Siguiendo el mismo paradigma Littler[45] busca determinar cuándo una red SCADA es penetrada ilegalmente mediante un software llamada ITACA. ITACA no solo es capaz de escuchar los paquetes que viajan dentro de la red bajo el protocolo IEC 60870-5-104(protocolo para la industria energética), sino que puede analizar el tráfico. Para la detección de paquetes o comandos no deseados se programa el software ITACA con un conjunto de reglas que permiten hallar paquetes corruptos con una eficacia del 100% de acuerdo con los resultados publicados.

Otra aproximación es la de Carcano [43], en donde, por medio de la evaluación de las variables críticas de un sistema SCADA es posible identificar cuando está siendo atacado. El principio de funcionamiento se basa en la idea de que si alguien quiere alterar el normal funcionamiento de una planta debe alterar el valor de las variables de control, de tal suerte que todo el sistema entre en un estado de riesgo. Para discernir entre un posible ataque y el mal funcionamiento de la planta, Carcano utiliza el estudio de los paquetes en busca de códigos u órdenes maliciosas.

Siguiendo la línea de los trabajos de tipo descriptivo, Schuster [44] describe un posible método para la implementación de un sistema IDS distribuido, en donde no solo se puedan monitorear una parte específica del proceso sino varias partes al mismo tiempo. La metodología que se plantea utiliza el análisis de tráfico para determinar cuándo un intruso entre en escena y parte de la premisa que dentro de una red SCADA las rutas de comunicación tienden a ser siempre las mismas. Específicamente, la aproximación busca detectar ataques donde los sistemas SCADA son muy vulnerables; es decir, ataques como ‘Man in the middle’ o ‘Denial of the service’. En este tipo de ataques es necesario que el atacante altere el normal tráfico de la red de control.

Kim [46] por su parte, en otro trabajo de tipo descriptivo, propone la creación de un sistema IDS encargado de analizar patrones de comportamiento para detectar una posible amenaza. En su esquema el IDS-SCADA ha de ser dinámico con el fin de estar preparado para identificar nuevos ataques; es decir, debe tener la capacidad de reconocer patrones anómalos a pesar de que éstos no se encuentren registrados en su base de datos. En este sentido es donde cobran relevancia los métodos basados en aprendizaje de máquinas.

El trabajo de Yang [47], dentro de la línea de los trabajos experimentales, hace una excelente aproximación al IDS-SCADA planteado por el laboratorio de Idaho[12]. Este IDS es un sistema multi-atributo, en donde para detectar una posible intrusión el sistema se cuenta con tres metodologías diferentes operando simultáneamente. Un analizador de protocolos, sumado a un sistema de reglas y a un detector de correlación, que permiten obtener una eficacia hasta del 100% según los resultados publicados, con un tiempo de respuesta que no supera los 256 microsegundos. El sistema soporta más de 2 protocolos relacionados con las comunicaciones de la industria eléctrica, donde los trabajos mencionados anteriormente se cubrían en su mayoría tan solo en uno y además tiene

capacidad de detectar hasta 3 ataques que son: Man in the middle Attack, Dos, ARP Spoofing.

Hasta el momento, si bien es cierto que los trabajos realizados cumplen con la función de detectar intervenciones maliciosas, no se ha mencionado ningún trabajo de tipo experimental que permita la detección de nuevos ataques, o que se adapte a nuevas condiciones de operación.

En esta última parte se describen los proyectos basados en aprendizaje de máquinas, los cuales aprovechan la habilidad de ciertos algoritmos para clasificar o agrupar información con el fin de identificar posibles amenazas o paquetes corruptos.

En [48] y [49] se encuentran 2 distintas implementaciones por parte del mismo autor para el análisis de paquetes utilizando máquinas de vectores de soporte de una clase y algoritmos de agrupamiento como es K-medias (K-means). Básicamente, este tipo de IDS requiere de dos procesos para operar correctamente: El primero consiste en el entrenamiento del sistema, en donde se le entrega al IDS una gran cantidad de muestras que no se consideren corruptas con el fin de que las pueda reconocer y clasificar. Una vez se ha hecho esto, en la segunda etapa, el IDS es capaz de seleccionar los paquetes corruptos de los no corruptos con la gran ventaja de que no necesariamente necesita conocer las características de la nueva amenaza.

El método OCSVM, al ser variación del método original SVM biclase, solo se requiere un tipo de muestra para el entrenamiento; no obstante, para análisis en tiempo real es bastante efectivo ya que requiere menor de tiempo de ejecución. La mejora al sistema inicial propuesto en [48] se describe en [49] al añadir el algoritmo de agrupamiento K-medias al IDS inicial. K-medias agrupa los paquetes de red del sistema SCADA según sus características comunes caracterizando entre estos los paquetes corruptos. La correlación de los resultados obtenidos por los algoritmos de agrupamiento es en últimas la que permitió al autor mejorar la eficiencia del IDS propuesto. Según los resultados publicados esta aproximación alcanza casi el 100% de eficiencia; sin embargo, si se considera que solo se tuvieron en cuenta 2 características de entrenamiento (tamaño de paquetes y tasa de actualización), no es posible dar absoluta credibilidad a este resultado partiendo de la poca cantidad descriptores.

Finalmente, tenemos el trabajo de Nader[50], el cual utilizando una máquina de soporte del tipo One Class, propone un modelo para la detección de ciberataques en un sistema de distribución de agua. Utilizando un la distancia de Mahalanobis, se propone incrementar la eficiencia de la detección con respecto a otras metodologías basadas también en algoritmos supervisados. En su trabajo reporta una eficiencia del 100% para el modelo propuesto.

Hasta aquí se ha presentado una descripción somera de los diferentes trabajos realizados en el área SCADA-IDS; por tanto, en el siguiente capítulo se hará un análisis global en donde se comparan los diferentes puntos en común de cada uno de los proyectos.

3.3. Discusión y Análisis

En este apartado se discutirán las ventajas y desventajas de los trabajos realizados, así como los aspectos que no fueron trabajados y que fueron tenidos en cuenta para el desarrollo de este trabajo.

La tabla 2 incluye aspectos claves de análisis para los trabajos más representativos en estudio. La vigencia del trabajo, basados en la fecha de la publicación, permite determinar si el trabajo tiene aplicación en los sistemas SCADA actuales; el país donde se desarrolla el proyecto nos permite para conocer la procedencia de las fuentes y mirar en donde se han realizado la mayor de los trabajos; el tipo de herramientas utilizados y la calidad de los resultados obtenidos nos permite conocer las aproximaciones realizadas y su resultado.

ID	Proyecto	AÑO	País	herramientas	protocolo	Técnica	Ataque analizado	Eficiencia	Trabajos Futuros
[41]	Idaho National Laboratory Supervisory Control and Data Acquisition Intrusion Detection System (SCADA IDS)	2008	EE.UU	No se especifica	Aplica para todos los protocolos.	-Flujo de comandos y de datos. -Análisis profundo de paquetes. -Identificación de protocolos.	Man in the middle, corrupción de paquetes.	No aplica.	El trabajo es del tipo descriptivo, por tanto no se plantean trabajos futuros. Se especifican condiciones deseables para un IDS para sistemas SCADA.
[42]	Snort IDS for SCADA networks	2009	Australia	Snort Wireshark Etherape	Modbus, dnp3	Se utiliza Snort como IDS en una red conformada por unidades virtuales y reales.	No se especifica	Sin evaluar.	Implementación de honeypot para identificar otros posibles ataques.
[43]	A Multidimensional Critical State Analysis for Detecting Intrusions in SCADA Systems	2011	Italia	C#	Modbus	Análisis de estados Críticos	Corrupción de paquetes.e intervenciones externas	99%	No se describen trabajos futuros.
[44]	A Distributed Intrusion Detection System for Industrial Automation Networks	2012	alemania		Profinet	Detección de intrusos en sistemas SCADA mediante el monitoreo del tráfico entre controladores bajo el estándar profinet	Denial of service y Man in the middle attack.	No aplica	Realización de pruebas teniendo en cuenta el estándar y los ataques mencionados.
[45]	RULE-BASED INTRUSION DETECTION SYSTEM FOR SCADA NETWORKS	2013	Reino Unido	ITACA(software de sniffing y análisis de tráfico de red IP)	IEC 60870-5-104 protocol	Signature based.-model based para detección de intrusos	Corrupción de paquetes	100%	No se especifica en el artículo, pero en el año 2014 se escribe un nuevo artículo abordando una nueva técnica llamada IDS multiatributo para sistemas SCADA.
[46]	Network Anomaly Detection for m-connected SCADA Networks	2013	Corea del sur	No se especifica	No se especifica	network-based pattern reference. El IDS propuesto genera reglas dinamicas para detectar un posible ataque.	No se especifica	No se espefica	El artículo tiene un enfoco descriptivo, por tanto solo aborda la posible eficacia de la técnica descrita.

[47]	Multiattribute SCADA-Specific Intrusion Detection System for Power Networks	2014	Reino Unido	ITACA (C/C++)	IEC 60870-5 series, DNP3, propietary protocols, entre otros.	- Protocol-Based Whitelists (PBWs) - Behavior-Based Rules (BBRs) -Correlation detector	-Man in the middle Attack -Dos -ARP Spoofing	100% para una rata de 180kb/s y en un tiempo menor a 254 us.	
[48]	Intrusion Detection in SCADA systems using Machine Learning Techniques	2014	Reino Unido	CoCkpitCI	DNS, FTP, MDNS, MODBUS, MODBUS, TCP Y UDP.	One-Class Support Vector Machine	Man in the middle, syn flood	98% Y 99%	Se prevé añadir detalles como reputación de la fuente, identificación de protocolos, mejorar eficiencia entre otros.
[49]	OCSVM model combined with K-means recursive clustering for intrusion detection in SCADA systems	2015	Reino Unido	CoCkpitCI	DNS, FTP, MDNS, MODBUS, MODBUS, TCP Y UDP.	One-Class Support Vector Machine y K-means recursive clustering	Corrupción de paquetes.	No se espcifica.	Analizar el funcionamiento del método para diferentes tipos de ataques.
50	Detection of cyberattacks in a water distribution system using machine learning techniques	2016	Francia	Planta de distribución de Agua.	No especifica	One-Class Support Vector Machine modificada	Corrupción de paquetes	100%	Analizar criterio de distancia diferentes para mejorar la eficiencia.

Tabla 2. Análisis de Trabajos Relacionados con Sistemas IDS para SCADA

En la información mostrada en la tabla 1 se evidencia que la mayoría de los trabajos en el área SCADA-IDS se han realizado en Europa, con una participación menor de otras naciones como Corea del Sur y Australia. Queda claro que en el continente americano, con excepción de los EE.UU aún no se la ha dado la relevancia suficiente al tema de la ciberseguridad de los sistemas SCADA y más en particular los IDS aplicados a los mismos. Por otro parte, la fecha de las publicaciones da cuenta de las incursiones en este campo son relativamente recientes, siendo la más antigua en el año 2008, en donde apenas se esbozaba este tipo de temas desde un enfoque netamente teórico. Considerando que la publicación más reciente corresponde al año 2016 se puede concluir que todavía hay demasiados aspectos a investigar en el mundo de los SCADA-IDS.

Otro aspecto bastante importante que se puede deducir de los trabajos realizados es el tipo de herramientas que se utilizan para la implementación de los sistemas mismos. Es importante resaltar que la mayoría de los sistemas SCADA son de tipo propietario; por ende, los protocolos de comunicación asociados a los mismos no siempre están abiertos al público en general. En el caso de los trabajos mencionados en la tabla 2, la mayoría de las herramientas utilizadas son “open source”, indicando que la gran parte de los análisis se hicieron filtrando paquetes y decodificando las tramas de sistemas propietarios o en su defecto se realizaron bajo estándares abiertos. Protocolos como Modbus, DNP3, TCP, UDP son los protocolos más utilizados dentro de los distintos trabajos, ya que son estándares cuya configuración está disponible al conocimiento público. En lo que a los lenguajes de programación se refiere, el lenguaje de programación más utilizado es C, debido a que es un lenguaje más cercano a la máquina que permite la ejecución de los algoritmos de una forma más rápida y eficaz.

Ahora bien, el análisis de las técnicas empleadas para la detección de intrusos es, sin lugar a dudas, uno de los puntos críticos a comparar en cada uno de los trabajos. Si bien, los resultados publicados en términos de eficiencia son alentadores, es bueno recordar que algunos trabajos no añaden ningún tipo de innovación a la concepción clásica de un IDS. La determinación de intrusos por medio de reglas preestablecidas [42][45], aunque eficaces para ataques conocidos, no muestran un cambio de paradigma que permita la detección de ataques no conocidos. En este sentido, propuestas más novedosas como son los detectores de estado crítico [43], máquinas de aprendizaje [48][49][50] y sistemas multi-atributos[47] proponen soluciones que permiten al IDS prever nuevos escenarios de ataque.

Por último, pero no menos importante, encontramos los tipos de ataque que el sistema IDS-SCADA debería ser capaz de detectar. La gran parte de los artículos analizados están de acuerdo en que detectar ataques como "man in the middle" o ARP Spoofin debe ser uno de las prioridades de los SCADA-IDS, ya que las redes de control en general son bastante susceptibles a los mismo; sin embargo, en ninguno de los trabajos se propone un análisis correlacional del comportamiento de las variables de los controladores, con el fin de reconocer paquetes corruptos como producto de dichos ataques. Dicho reconocimiento permite que el IDS centre su atención en el contenido y no en la procedencia, ya que, como es sabido, en un ataque de suplantación de identidad, el supuesto origen del paquete es engañoso.

3.4. Áreas de Investigación

Después del análisis de los trabajos realizados en el área de la detección de intrusos, se evidenciaron las siguientes áreas de investigación, que permitieron el desarrollo del presente trabajo.

- *Detección de manipulación no deseada en variables de control y supervisión mediante técnicas para la detección de "Outliers".*
  En todos los trabajos que se han analizado hasta ahora, ninguno hace una distinción entre las variables de supervisión y control. Este hecho es bastante crucial, ya que manipular indebidamente una variable de control en una planta, pueda traer consecuencias inmediatas que afectan el buen funcionamiento de la misma.

- *Sistemas IDS para la detección de ciberataques en sistemas SCADA considerando mayor número de descriptores (características de entrenamiento para un sistema basado en aprendizaje automático).*
  En los trabajos analizados, la cantidad de variables que se tuvieron en cuenta para los análisis son relativamente pequeñas. Este caso en particular, es más significativo para los IDS basados en algoritmos de aprendizaje en donde solo se utilizaron 2 descriptores como criterio de clasificación [48][49].

- *Detección de fallas en sistemas SCADA mediante el uso de aprendizaje de máquinas.*
  Los sistemas para la detección de fallos en sistemas SCADA utilizan normalmente como criterio de evaluación umbrales, los cuales, al ser sobrepasados por una variable indican una posible avería o mal funcionamiento de la planta; sin embargo, si el número de variables a analizar es muy grande este tipo de metodologías no se hace viable. El uso de algoritmos de aprendizaje es más eficaz, no solo porque sería posible analizar la correlación de un gran número de variables al mismo tiempo; sino que a su vez nos da la posibilidad de crear un modelo adaptativo donde no se requiere de una modificación manual de cada uno de los umbrales de operación.

- *Análisis en dispositivos de Control y no en el tráfico*
  En ninguno de los sistemas de detección estudiados se analizó el comportamiento de las variables de los controladores, lo cual es crucial a la hora de detectar ataques cuya acción sea paulatina o que modifiquen parámetros que normalmente no son monitoreados por los sistemas SCADA, como es el caso de las variables tipo marca en los PLC.

- *Estándar de interoperabilidad para las comunicaciones*
  Para todos los casos estudiados, el análisis solamente aplicaba para dispositivos que estuvieran comunicándose bajo un determinado protocolo. El uso de un estándar de interoperabilidad como OPC, permite el acceso a las variables de un inmenso número de dispositivos de control, bajo una gran cantidad de interfaces (Ethernet, MPI, RS485, serial, etc).

4.Planteamiento de la Solución

En este capítulo se hará una descripción detallada del sistema IDS-SCADA que se ha creado considerando los problemas de investigación hallados en el análisis del estado del arte. A continuación, se mostrarán las bondades y características del sistema, así como la descripción del laboratorio de pruebas que se creó con el fin de evaluar qué tan eficiente es el sistema en contextos reales.

4.1. Alcance de la Solución Propuesta

Con el fin de proveer una solución eficaz que permita la detección de paquetes corruptos independientemente de cual sea su origen o destino y poder determinar las amenazas en las variables de entrada, salida y marcas de forma separada en los controladores de una red SCADA, se propone la creación de un sistema para la detección de intrusos en tiempo real; el cual, mediante el uso de análisis de variables y algoritmos de aprendizaje automático, permitirá identificar cuando las variables del proceso se han salido de los rangos aceptables o están siendo manipuladas con el fin de estropear el buen funcionamiento de sistema de control.

Para cumplir con dicho propósito se diseñó un sistema IDS, el cual se puede analizar considerando la topología de red mostrada en la figura 15.

Como se puede observar, el sistema IDS que se diseñó se comporta como un servidor SCADA redundante, mediante el cual es posible monitorear las variables críticas de los dispositivos PLC en una red SCADA. En el caso de un eventual ataque o comportamiento anormal de los dispositivos de control, el sistema IDS reporta la alarma mediante la activación de una señal, mientras que simultáneamente analiza el paquete anómalo para determinar en qué tipo de variable del controlador se presentó dicho comportamiento; de esta manera, el sistema propuesto no solo sirve para determinar ataques, sino que a su vez puede servir como sistema de alarma de un proceso sin la necesidad del uso de umbrales o la detección de averías en el controlador, sensores o actuadores. En apartados posteriores se explicará en mayor detallé cada una de las características y ventajas previamente mencionadas.

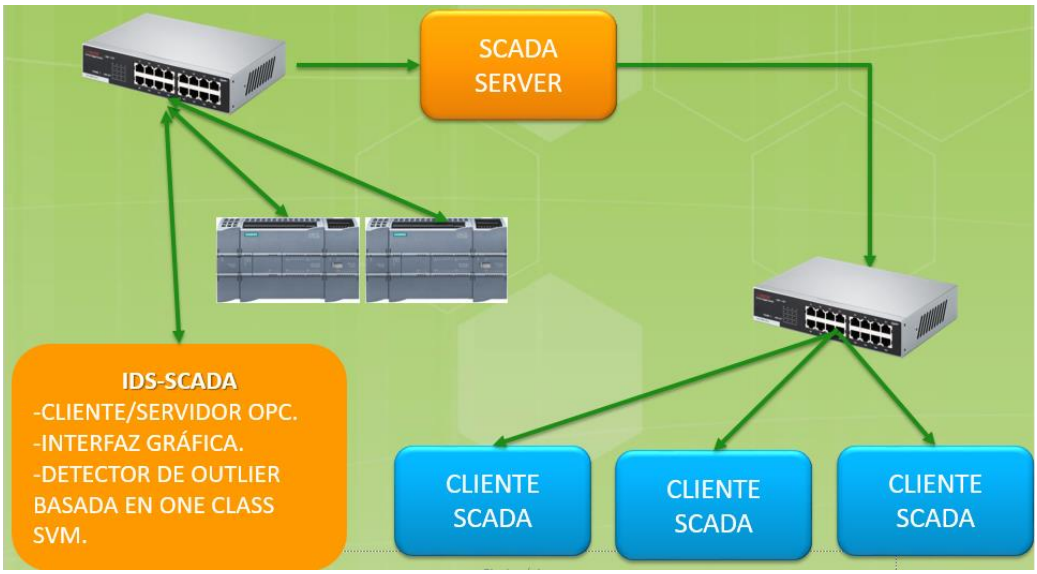


Figura 15. Topología de Red para el IDS-SCADA

4.2. Características y Ventajas del IDS-SCADA Propuesto

Si se compara el IDS-SCADA propuesto, con los trabajos relacionados en el estado del arte del capítulo anterior, podemos encontrar notables diferencias y ventajas. A continuación, se enumeran algunas de ellas, resaltando sus virtudes con respecto a otras soluciones.

a. Estándar de interoperabilidad para el análisis de paquetes

A diferencia de los trabajos analizados, en donde el análisis se realiza para un conjunto de protocolos dado, el IDS-SCADA propuesto analiza directamente el contenido de las variables de los dispositivos de control, valiéndose del estándar de interoperabilidad OPC. La ventaja de utilizar este estándar radica en que, por medio de un servidor, es posible interactuar con una amplia gama de dispositivos que no necesariamente son controladores PLC; esto último permite expandir la solución a otro tipo de automatismos, mientras que facilita el análisis de la información ya que no se requiere conocer la estructura específica de un protocolo para tener acceso a los campos en donde se encuentra la información relevante para la operación del IDS-SCADA.

b. Identificación de variables atacadas

En ninguno de los trabajos anteriores se describe una metodología para la identificación del tipo de variables que está siendo manipulada indebidamente o cuyo comportamiento no se ajusta al modelo de entrenamiento. Esto sin lugar a dudas representa una ventaja enorme del sistema IDS propuesto, ya que no solo es capaz de identificar un comportamiento no deseado, sino que a su vez puede determinar en qué tipo de variables se presentó.

La identificación del tipo de variable afectada, permite la identificación de los siguientes escenarios:

- **Falla de los dispositivos de sensado:** Si la anomalía se presenta en las variables de entrada de los controladores, es bastante probable que la anomalía se deba a fallos en los dispositivos sensores; los cuales, son los encargados de llevar la información del proceso, al PLC en forma de señales eléctricas. En el escenario planteado, el sistema IDS, se comporta como una plataforma de supervisión para la detección de averías.

- **Detección de cambios de firmware sutiles:** Se toma como referencia uno de los casos más famosos en el mundo de la ciberseguridad de los sistemas SCADA, el gusano Stuxnet[50]. Este ataque modificaba los parámetros de operación de aproximadamente 1000 máquinas en una central nuclear en irán, sin ser detectado oportunamente. En nuestra solución se dotó al sistema IDS con la capacidad de analizar el comportamiento de las variables internas o marcas en los PLC con el fin de detectar posibles ataques, que en la mayoría de los casos resultan invisibles para los mecanismos de supervisión tradicionales.

  Como se sabe, los sistemas de supervisión basados en umbrales basan su análisis en detectar cuando una variable en específico se sale de una determinada región de operación. El problema es que algunas regiones útiles de la memoria no son analizadas, por lo que un potencial ataque puede ser implantado sin que tengas efectos mediáticos sino progresivos. Por otra parte, al no utilizarse modelos que correlacionen el comportamiento de los distintos parámetros como un todo, podría darse el caso que un determinado número de parámetros se encuentre de forma individual en una región ideal de operación, pero que su conjunto sea dañino para el correcto funcionamiento de un proceso.

- **Cliente o Servidor SCADA comprometido:** Una detección en el comportamiento anómalo en una de las interfaces de salida en el PLC, puede ser indicio de la mala operación del mismo, pero también de un posible ataque desde una aplicación cliente o servidor SCADA.

c. **Implementación en Labview:** Debido a que el entorno de programación de Labview, fue diseñado con el fin de ofrecer ambientes de visualización para la instrumentación virtual y por supuesto, aplicable a los sistemas SCADA, fue posible reunir en una sola interfaz de programación: la captura, visualización y análisis asociado al IDS-SCADA. El programa cuenta además con su propio servidor OPC, con lo que se facilitó bastante la comunicación y el acceso a las diferentes variables del PLC del banco de pruebas.

d. **Gestión de Alarmas Automático:** Muchos de los sistemas SCADA requieren de la programación de alarmas de forma manual, lo cual resulta especialmente tedioso cuando se trata de miles de variables que deben ser monitoreadas simultáneamente. El IDS-SCADA propuesto puede también ser utilizado como gestor de alarmas automático, mediante el análisis de los datos de entrenamiento y el algoritmo para la detección de "outliers"; de esta manera es posible determinar los valores no válidos para las variables de un proceso sin la necesidad de programarlos manualmente.

e. **Análisis de los controladores de forma independiente:** En todos los trabajos relacionados, el análisis de la información se realizaba para el conjunto de todos los datos de un sistema SCADA y no para cada controlador en particular. El IDS-SCADA diseñado permite el análisis de los controladores de forma individual, lo cual facilita la identificación de anomalías en forma más detallada. Alguien podría objetar que dicho análisis podría ser bastante complejo considerando la gran cantidad de controladores disponibles en algunas plantas; no obstante, no se hace necesario analizar cada uno de los controladores, sino solo aquellos que son más críticos, ante una posible amenaza. En dicho caso, para las plantas que ejecutan sistemas de control distribuido, no se hace necesario el análisis de cada uno de los PLC esclavo, sino prioritariamente los dispositivos maestros. En lo que respecta a las variables a ser analizadas, tampoco se requiere el total de ellas, sino solamente las más críticas.

4.3. Esquema de Detección IDS-SCADA Propuesto

El sistema IDS-SCADA propuesto se desarrolló en 4 etapas mediante las cuales es posible el entrenamiento del sistema y la posterior detección de anomalías en tiempo real. El esquema de la solución se puede observar en la figura 16, y a continuación se describe en detalle cada una de las etapas.

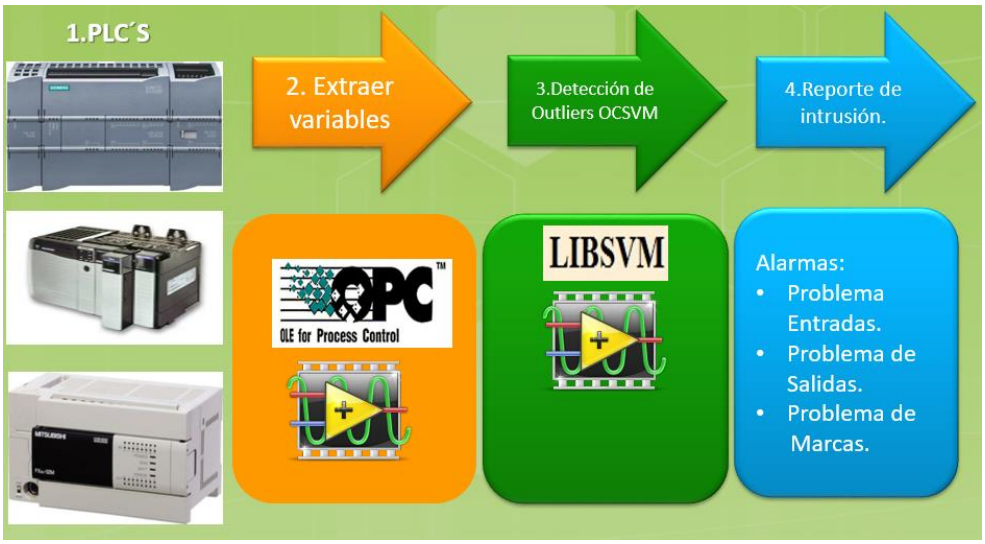


Figura 16. Esquema de Detección IDS-SCADA Propuesto

4.3.1. Dispositivos PLC

Como se mencionó en el apartado anterior, el IDS-SCADA propuesto analiza cada uno de los controladores de forma independiente; sin embargo, si existe una relación de dependencia entre varios PLC, es posible que estos se analicen como conjunto y no como entes particulares debido a que muchas de sus variables están interconectadas y los procesos que se ejecutan dependen los unos de los otros.

Los dispositivos PLC se analizan de forma individual porque el modelo de detección, al basarse en una máquina de soporte vectorial, es por naturaleza correlacional; es decir, es necesario que al comportamiento de una variable específica le corresponda un determinado patrón en el resto de las variables en análisis. Si los PLC analizados no guardan ninguna relación en el comportamiento de sus variables, el modelo obtenido no sería eficiente para la detección de posibles anomalías debido a la falta de interdependencia de sus variables, mientras que a su vez se dificultaría la identificación de aquellas cuyo comportamiento es anormal.

Otra de las condiciones que deben cumplir los dispositivos de control que van a ser analizados es que estos deben ser compatibles con el servidor OPC utilizado. En la página oficial de la *National Instruments* [51] se muestran todos los dispositivos compatibles con el servidor NI OPC; este es el servidor que se utilizó para enlazar los datos con la plataforma Labview.

Finalmente, es clave entender que la cantidad de dispositivos analizados depende directamente de la complejidad de los procesos controlados por ellos, del número de variables asociados, y sobre todo de la potencia de la máquina en la que se ejecuta la aplicación IDS-SCADA.

4.3.2. Extracción de Variables

La extracción de variables se logra mediante la conexión del programa Labview y el PLC mediante el servidor NI OPC. Las variables críticas en el análisis se muestran en la tabla 3.

Tipo de Variable	Tipo de Registro	Descripción
Entrada	Entradas Digitales	Se lee el byte completo de cada uno de los registros de entrada.
	Entradas Análogas	Se lee las direcciones asociadas a la conversión análoga digital. Normalmente dicha información se almacena en una variable de 2 bytes.
Marcas	Definición de Umbrales	Son aquellas marcas que se utilizan para definir los límites de un proceso. Se consideran críticas porque su manipulación puede alterar el ciclo normal de los procesos.
	Estados de Operación	Determinan el estado actual de los procesos. Se les considera críticas porque su manipulación indebida puede inducir comportamientos erráticos del autómata a largo plazo.
	Variables asociadas a bases de Datos	Se utilizan ampliamente para la configuración de tramas que serán enviadas a dispositivos remotos como variadores de velocidad, entre otros. Además, pueden ser de gran utilidad para la configuración de recetas en el caso en que una misma planta ejecute diferentes rutinas. Se les considera críticas debido a que su manipulación indebida puede alterar las comunicaciones del PLC con otros dispositivos, o inducir a comportamientos inadecuados por la manipulación

		de los parámetros asociados a las bases de datos.
Salida	Salidas Digitales	Se lee el byte completo de cada uno de los registros de salida.
	Salidas Análogas	Se lee las direcciones asociadas a la conversión digital analógica. Normalmente dicha información se almacena en una variable de 2 bytes.
	Salidas Pulsadas	El comportamiento de dichas salidas, se asemeja al de las variables analógicas, por lo que solo basta tener acceso a la dirección de control. Normalmente dicha información se almacena en una variable de 2 bytes.

Tabla 3. Variables Críticas del PLC

Una vez las variables han sido enlazadas al instrumento virtual, estas deben ser normalizadas; luego se almacenan todas las posibles combinaciones que pueden darse en la ejecución normal de un proceso. Con el archivo que contiene los datos de entrenamiento se procede a entrenar la máquina OCSVM que luego será implementado en la detección de anomalías en tiempo real.

4.3.3. Bloque de Identificación

Para la detección de intrusiones en tiempo real se planteó la metodología mostrada en la figura 17, en donde los datos ingresan al bloque, se realiza la predicción por parte de la máquina de soporte vectorial y finalmente un criterio de distancia euclidiana permite determinar en qué tipo de variable o variables se presentó la anomalía.

Por medio de la implementación de la ecuación 4, es posible calcular la desviación en cada uno de las características que está siendo analizadas, en donde el dato en análisis corresponde a x, la cantidad de características es k y los datos de entrenamiento corresponden a v.

$$d = \sqrt{\sum_{0}^{k} (v_k - x)^2}$$

Ecuación 4. Distancia Euclidiana de un Dato con Respecto a los Datos de Entrenamiento

Una vez calculadas todas las distancias de un dato específico con respecto al conjunto de muestras de entrenamiento la mínima distancia en condiciones normales debería de ser pequeño o idealmente igual a cero; en caso de no ser así, se analiza el vector de distancia mínimo y se observa en cuál de sus componentes la desviación fue mayor. Este último procedimiento es el que da cuenta del tipo de anomalía que se dio en el proceso.

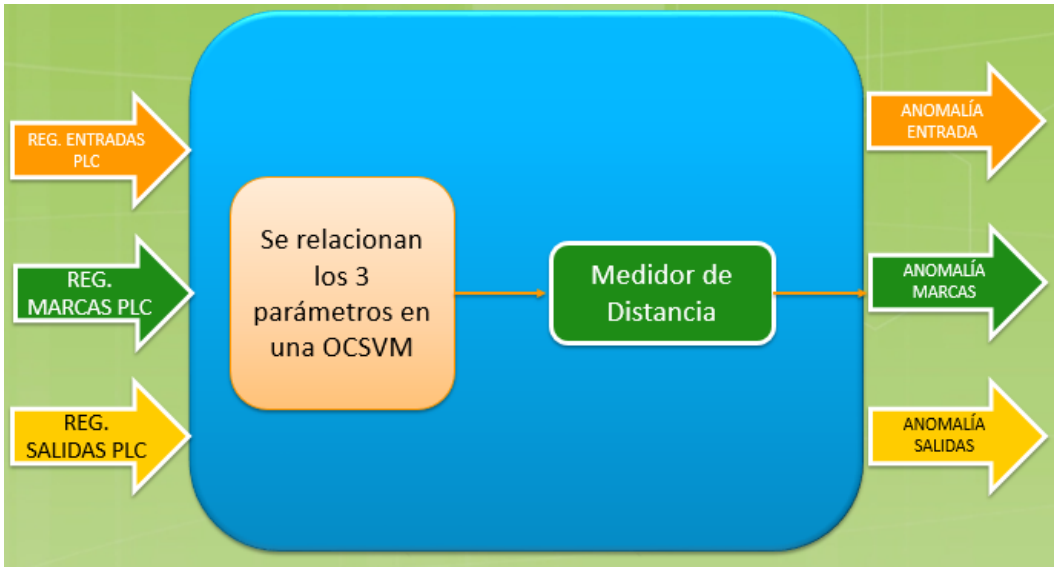


Figura 17. Bloque de Detección

4.3.4. Gestión de Alarmas

En el panel de visualización del sistema IDS-SCADA propuesto, una serie de indicadores luminosos, se activan en cada uno de los siguientes casos:

- **Indicador General:** Se enciende cuando una anomalía es detectada por el algoritmo OCSVM. Dicho indicador no ofrece información acerca del tipo de anomalía, pero en el flujo del programa es el encargado de activar el medidor de distancia crítico.

- **Indicadores Locales:** Son indicadores luminosos asociados a cada variable en análisis, que permiten determinar cuando esta se encuentra fuera de los rangos aceptables.

4.4. Caso de Uso: Laboratorio SCADA

El diseño e implementación de un laboratorio SCADA permitió validar el modelo que hasta ahora se ha expuesto. Como se puede observar en la figura 18, el laboratorio SCADA consta de los siguientes elementos: Un proceso que se describirá en detalle más adelante, un PLC S7-1200 con salidas tipo transistor, una pantalla HMI de 6 pulgadas a color, Un servidor Intruso y un IDS-SCADA.

Figura 18. Laboratorio SCADA

4.4.1. Proceso

El proceso que se ha implementado consiste en una grúa con movimientos horizontal y vertical. La grúa permite el desplazamiento de un objeto en 3 estaciones que son: estación base desde donde el objeto parte y es nuevamente traído una vez el proceso termina; estación de calefacción, en donde el objeto es calentado por un tiempo determinado; y la estación de enfriamiento donde el objeto se expone a un ventilador.

Este tipo de proceso se seleccionó por las siguientes razones: En principio, y para todos los trabajos revisados, no existe un tipo de proceso estándar que se haya certificado para validar la eficiencia de un sistema para la detección de intrusos en redes SCADA. Para todos los casos bajo estudio los procesos analizados resultaron ser variables, siendo en muchos de los casos ambientes totalmente simulados y no plantas reales.

En este caso en particular, las primeras aproximaciones al laboratorio SCADA se plantearon basadas en la simulación de un proceso con un microcontrolador que modificaba los puertos de entrada del PLC; sin embargo, este modelo resulta ser demasiado ideal y no considera los tiempos de respuesta de los sensores, así como posibles anomalías en los dispositivos de suicheo. Por otro parte, los sistemas electromecánicos reales son susceptibles al ruido que en determinadas ocasiones puede ocasionar una mala operación de los algoritmos de aprendizaje de máquinas. Esta es precisamente una de las ventajas de las máquinas de soporte vectorial en donde la clasificación no se ve alterada de forma crítica por el ruido de las muestras. La figura 19 muestra el esquema planteado inicialmente.

Figura 19. Laboratorio SCADA con Proceso Simulado con Microcontrolador

Por otra parte, a nivel de los procesos industriales podemos encontrar procesos que podríamos llamar concurrentes, en donde un cambio en los parámetros de control trae consecuencias inmediatas sobre este. Normalmente dichos procesos se programan usando metodologías combinacionales; pero para efectos de validación de este trabajo no resultaban prácticos debido a que uno de las virtudes del IDS-SCADA propuesto es precisamente advertir sobre anomalías que no tienen efectos mediáticos sino paulatinos. Siguiendo la línea argumental, se diseñó e implementó un proceso que

pudiera programarse de forma secuencial, aplicando la programación secuencial para PLC basada en las redes de Petri [52]; en este tipo de proceso y bajo ciertas restricciones pueden sembrarse semillas de ataque, que sean imperceptibles para un sistema SCADA real, pero potencialmente dañinas.

Finalmente, la necesidad de un proceso a escala era determinante, ya que en los procesos industriales a nivel macro, la manipulación indebida de los mismos conducirían irremediablemente a grandes daños y escenarios sumamente peligrosos; un modelo a escala facilita la recreación de escenarios de ataque o anomalías, sin grandes compromisos a nivel económico, de infraestructura o lesiones personales.

a. Elementos del proceso

La Figura 20, muestra un esquema del proceso, mientras se describen cada uno de sus elementos, funcionalidades, conexiones y restricciones en la tabla 4.

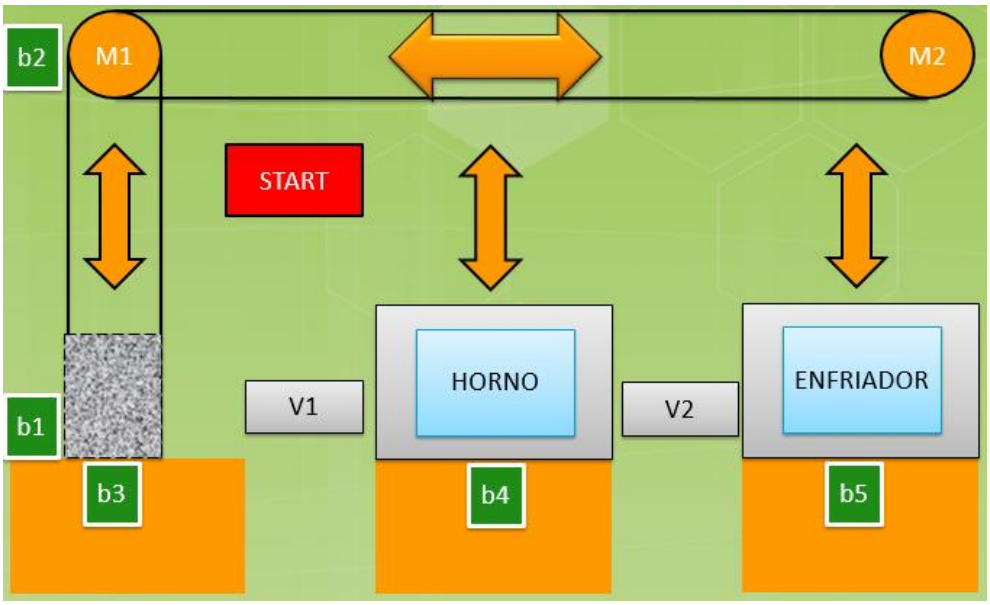


Figura 20. Esquema del Proceso de Validación

Elemento	Funcionalidad	Conexión	Restricciones
Motor 1	Permite el desplazamiento vertical del objeto en todo su recorrido. El voltaje de operación es de 12v y un consumo de corriente inferior a los 50mA.	Se controla mediante una señal PWM generada por el PLC.	-Los límites del movimiento son advertidos por la activación de los pulsadores B1 y B2. -El rango aceptable de variación en el ciclo de dureza para la velocidad del motor está entre 90 y 100%. Para ciclos de dureza

			menores se eleva el consumo de corriente.
Motor 2	Permite el desplazamiento Horizontal del objeto en todo el recorrido. Su voltaje de operación y corriente de consumo son iguales al motor 1.	Se controla mediante una señal PWM generada por el PLC.	-Las paradas en los movimientos horizontales está controladas por la activación de los sensores B3-B5. -El rango aceptable de variación en el ciclo de dureza para la velocidad del motor está entre 50 y 60%. Para ciclos de dureza mayores se pierde el sincronismo con los sensores.
Sensores B1 y B2	Son interruptores de final de carrera, que indican al PLC cuando el objeto ha alcanzado su extremo máximo o mínimo.	Se conectan directamente a los puertos digitales de entrada del PLC.	-De acuerdo con el modelo del proceso, en ninguna circunstancia deberían ambos estar activos al mismo tiempo. Si dicha situación se da, es porque alguno de los interruptores se encuentra averiado.
Sensores B3-b5	Son interruptores magnéticos encargados de sensar la posición horizontal de la grúa.	Se conectan directamente a los puertos digitales de entrada del PLC.	-De acuerdo con el modelo del proceso, solo uno de estos sensores puede estar activado al tiempo.
Relevos 1-4	Permiten el control del sentido de giro de los motores respectivos. Todos ellos tienen un voltaje de activación de 24 Voltios.	Se conectan directamente a las salidas tipo transistor del PLC.	-En ningún caso, la pareja de relevos que controla un motor, deberá estar activada simultáneamente.

Relevos 5-6	Permiten la activación del Horno y la ventilación. Todos ellos tienen un voltaje de activación de 24 voltios.	Se conectan directamente a las salidas tipo transistor del PLC.	-En ningún caso la pareja de relevos deberá estar activada simultáneamente.

Tabla 4. Descripción Elementos del Proceso

b. Interfaz de Potencia

La interfaz de potencia consiste en un circuito diseñado con el fin de facilitar las conexiones entre el proceso y el PLC. Para tal fin, dicha tarjeta está dotada de relevos para la conmutación de los actuadores previamente descritos, así como transistores de potencia que permiten la conmutación a alta velocidad de señales con el fin de variar la velocidad de los motores.

La tarjeta cuenta además con un acondicionamiento de señal para sensor de temperatura, conexiones para la polarización a 24 V y un regulador a 12 V para el control de los motores del proceso en un rango seguro.

4.4.2. PLC S7-1200

El núcleo del análisis para la detección de intrusos se encuentra en el monitoreo de cada una de las variables asociadas a los controladores, en este caso el modelo s7-1200 con salidas tipo transistor [53].

a. Justificación

Para el laboratorio de pruebas, se utilizó el PLC S7-1200 debido a que este cuenta, entre muchas otras cosas, con varias interfaces que facilitan enormemente su programación y diagnóstico.

Entre sus grandes ventajas, podemos mencionar su interfaz Ethernet, la cual permite comunicarse con otros dispositivos valiéndose del protocolo TCP/IP; esto facilita enormemente la interacción con el servidor OPC, mientras garantiza una alta velocidad en las comunicaciones. Por otro lado, la integración de módulos para la modulación en PWM y puertos análogos ligados directamente a la CPU, facilita enormemente las conexiones y la configuración en el entorno de programación TIA Portal [54] de Siemens.

b. Declaración de Variables

En el proceso de programación de un PLC, conviene declarar las regiones de memoria que se utilizarán para cada variable, asignándoles su respectivo nombre. El incorrecto

uso de la memoria disponible, induce a problemas de funcionamiento conocidos como solapamiento de memoria. En la tabla 5, se ilustran las respectivas regiones de memoria asignadas a cada elemento del proceso.

Variable	Tipo	Tipo de Dato	Dirección	Byte
B1	Entrada	bool	I0.0	IB0
B2	Entrada	bool	I0.1	
B3	Entrada	bool	I0.2	
B4	Entrada	bool	I0.3	
B5	Entrada	bool	I0.4	
START	Entrada	bool	I0.5	
PWM1	Salida	Bool	Q0.0	
HORNO	Salida	Bool	Q0.1	
PWM2	Salida	Bool	Q0.2	QB0
UP	Salida	bool	Q0.3	
DOWN	Salida	bool	Q0.4	
RIGHT	Salida	bool	Q0.5	
LEFT	Salida	bool	Q0.6	
VENTILADOR	Salida	bool	Q0.7	
E1	Marca	bool	M0.0	MB0
E2	Marca	bool	M0.1	
E3	Marca	bool	M0.2	
E4	Marca	bool	M0.3	
E5	Marca	bool	M0.4	
E6	Marca	bool	M0.5	
E7	Marca	bool	M0.6	
E8	Marca	bool	M0.7	
E9	Marca	bool	M1.0	MB1
E10	Marca	bool	M1.1	
E11	Marca	bool	M1.2	
Velocidad1	Salida	Word	QW2	QW2
Velocidad2	Salida	Word	QW4	QW4

Tabla 5. Variables PLC para banco de Pruebas

c. Secuencia de Estados

Para la programación del proceso en el PLC, se utilizó la metodología secuencial, valiéndose de estados que indicaran que tipo de operaciones debían realizarse en cada uno de ellos y evaluando las condiciones que debían cumplirse para saltar al estado siguiente. La figura 21 muestra el diagrama de estados detallado del proceso, mientras que en el Anexo A se encuentra disponible el programa en lenguaje Ladder que fue cargado en el PLC, el cual permite hacer un barrido paramétrico y de esa manera obtener todas las posibles combinaciones que servirán como datos de entrenamiento.

Figura 21. Diagrama de estados del Proceso

4.4.3. HMI

La pantalla HMI es la encargada de permitir la visualización de los diferentes estados del proceso, así como la configuración de las velocidades de los motores. Las flechas y graficas parpadean o cambian de color para describir el tipo de movimiento que se está realizando, indicar si algún sensor se encuentra o no activado y si el horno o el ventilador se encuentran en operación. En la figura 22, se muestra el diseño de la pantalla de control, con sus respectivas animaciones y comandos.

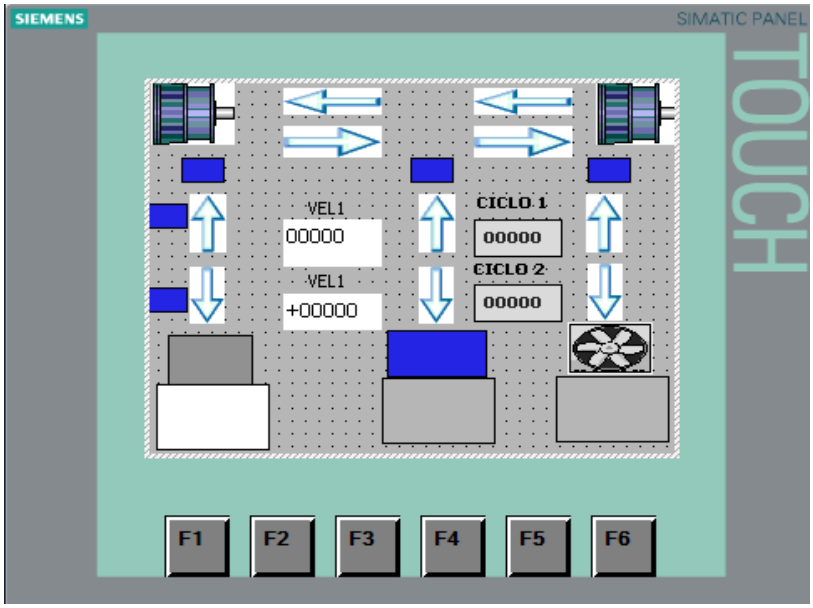


Figura 22. Panel HMI del Proceso

4.4.4. IDS-SCADA

Es el software de detección creado en el entorno Labview, el cual consta de los siguientes elementos: Captura de datos, detección de anomalías en tiempo real y finalmente un graficador de puntos donde se observa el comportamiento de los datos de entrenamiento en un ambiente tridimensional. En la figura 23 se muestra el panel de control diseñado para el IDS, y a continuación se describirá en detalle el funcionamiento de cada una de sus componentes.

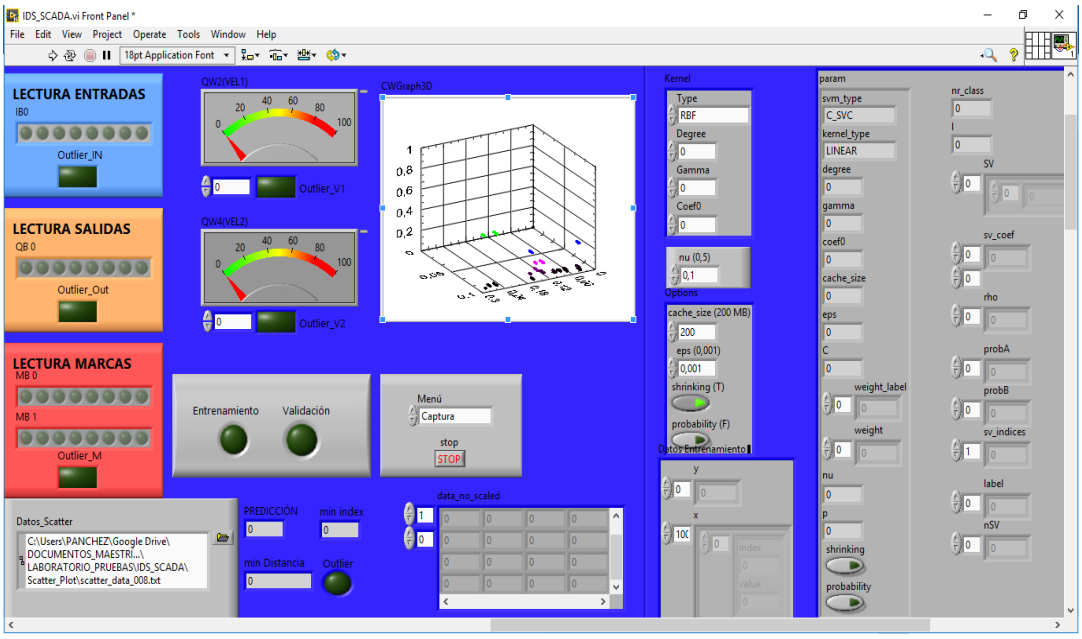


Figura 23. Panel de Control IDS-SCADA

a. Selección de Descriptores

Antes de entrar a describir cada uno de los elementos del sistema de detección, conviene definir el número de descriptores que se utilizaron para la detección de posibles anomalías en el proceso.

Como se había mencionado previamente, los descriptores son las características asociadas a cada dato y son estas las que permiten la clasificación y entrenamiento por parte del algoritmo clasificador, en este caso la OCSVM. Típicamente los descriptores, son las características que mayor información entregan acerca del ente que se encuentra bajo análisis, por lo que, basados en las restricciones del proceso, se escogieron las variables que aparecen en la tabla 6.

Descriptor	Dirección (byte)	Valor Mínimo Posible	Valor Mínimo Proceso	Valor Máximo Proceso	Valor Máximo Posible
Entradas	IB0	0	0	64	255
Salidas	QB0	0	0	128	255
Marcas	MB0-MB1	0	0	32768	65535
Velocidad1	QW2	0	90	99	100
Velocidad2	QW4	0	50	59	100

Tabla 6. Descriptores del Proceso

El proceso de normalización para cada descriptor se llevó a cabo utilizando la ecuación 1, en donde los valores máximo y mínimo corresponden a los valores del proceso y no los valores posibles.

Considerando los descriptores de la tabla 6, la estructura de cada dato que será analizado, una vez se haya normalizado, corresponde a un arreglo de 5 posiciones, el cual tiene la estructura mostrada en la figura 24.

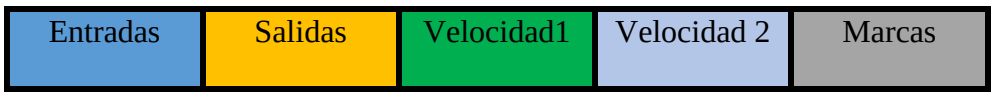


Figura 24. Estructura de Datos

En el caso particular de los datos de entrenamiento, estos tienen una etiqueta adicional, la cual, siempre será '1' indicando que dicho dato corresponde a un patrón de comportamiento normal del proceso en análisis. En la figura 35 se puede observar la estructura de los datos de entrenamiento.

Figura 25. Estructura de Datos de Entrenamiento

b. Captura de Datos

Para la captura de los datos la interfaz cuenta con una serie de indicadores, los cuales están enlazados directamente con los diferentes registros del PLC mediante el servidor OPC. En la figura 26 se muestra la respectiva configuración de las variables en el servidor OPC para luego ser leídas por el IDS-SCADA.

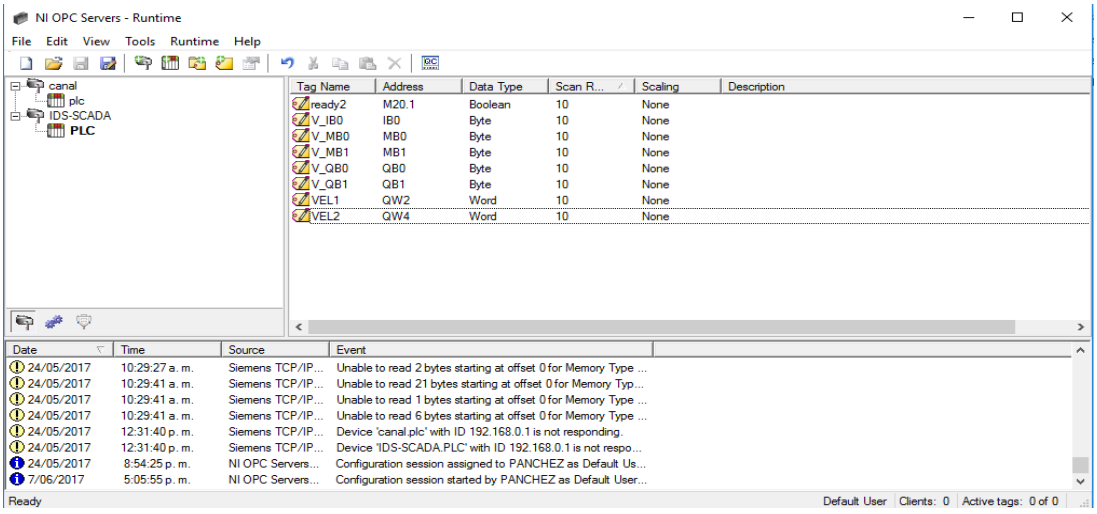


Figura 26. Configuración de Variables en el Servidor

La velocidad de muestreo para cada una de las variables analizadas es de 10ms, con lo que se garantiza una relativa alta respuesta, considerando que el proceso bajo análisis es relativamente lento.

En la etapa de entrenamiento un conjunto de datos debe ser almacenado para posteriormente ser utilizados para el entrenamiento de la máquina de soporte vectorial. En dicho caso se utiliza la sección de programa disponible en los Anexos B y C, donde las variables en análisis se escalan y se almacenan en un archivo con formato. Adicionalmente, el sistema ofrece la posibilidad de almacenar los resultados de las predicciones, de forma que puedan validarse los resultados.

c. Visualización

La etapa de visualización, va ligada directamente al panel de control y en ella se pueden apreciar básicamente 2 elementos fundamentales:

- **Visor de eventos:** Se encuentra conformado por animaciones tipo led y de aguja que permiten la visualización de los diferentes estados y registros internos asociados a las variables del PLC. Dicha etapa de visualización permite que adicionalmente a la detección el sistema pueda servir como sistema SCADA auxiliar.

  En la parte inferior, asociada a cada variable monitoreada, un indicador led se ilumina cuando alguna de las variables exhibe un comportamiento anormal. La figura 27, presenta una visión más detallada de esta etapa.

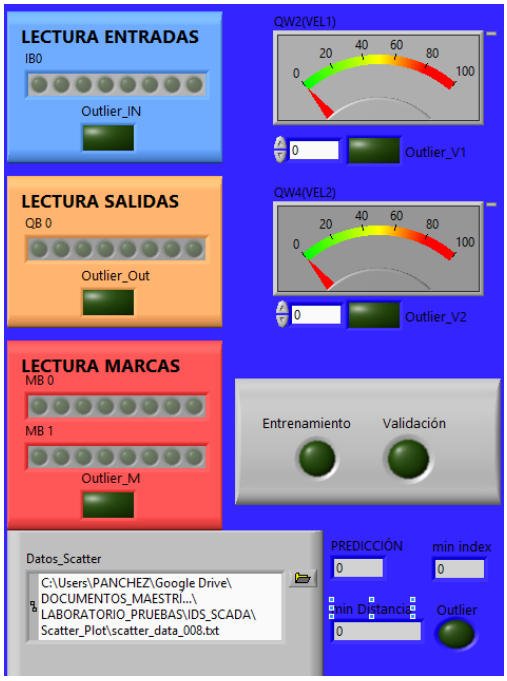


Figura 27.Visor de Eventos IDS-SCADA

- **Visor de distribución de Datos:** Mediante un grafícador 3D, es posible observar la distribución de las variables de entrada, salida y marcas en el espacio. Esto es bastante útil a la hora de determinar qué tipo de función

Kernel es más eficiente a la hora de separar los estados normales y anormales del proceso. En la figura 28 se puede observar el aspecto del graficador en el panel de control, mientras que en los Anexos D y E se aprecian los diagramas de bloques respectivos.

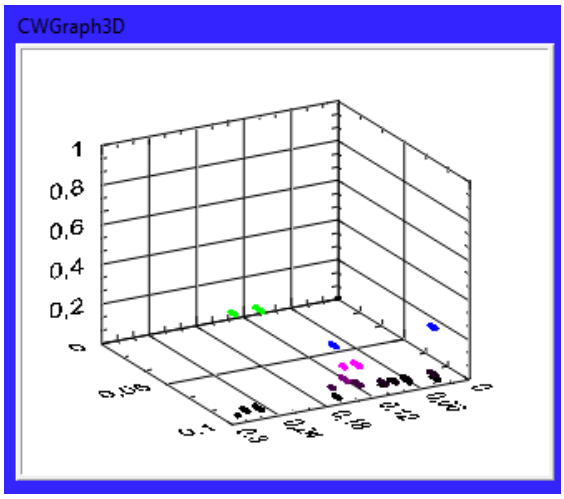


Figura 28.Visor 3D para datos de Entrenamiento

d. Predicción

La etapa de predicción, se fundamenta en la implementación de la OCSVM en el entorno Labview y la implementación del algoritmo medidor de distancias críticas. Como se sabe, por apartados anteriores, antes de poder predecir una posible anomalía en el sistema es necesario un entrenamiento de la máquina de soporte vectorial; el entrenamiento permite discriminar si los datos analizados corresponden o no al patrón de comportamiento típico. El resultado de este proceso de entrenamiento da lugar finalmente a un modelo, que es utilizado por el bloque de predicción para el análisis en tiempo real.

La figura 29, muestra los diferentes elementos de configuración que se encuentran disponibles en el panel del IDS-SCADA, mientras se describen algunos de los parámetros, controles e indicadores más importantes.

- **Configuración de Kernel:** Por medio de este conjunto de comandos es posible determinar el tipo de Kernel que va a ser empleado en la predicción, mientras se configuran los parámetros asociados al mismo. Para el caso particular de las OCSVM, el tipo de Kernel más empleado es el RBF; los parámetros críticos en este caso son: *nu y Gamma,* los cuales definen que tan cerca o lejos se encuentra la frontera de decisión con respecto a los vectores de soporte obtenidos en la etapa de entrenamiento. Para otro tipo de Kernels

es posible seleccionar parámetros como grado del polinomio, el coeficiente inicial entre otros.

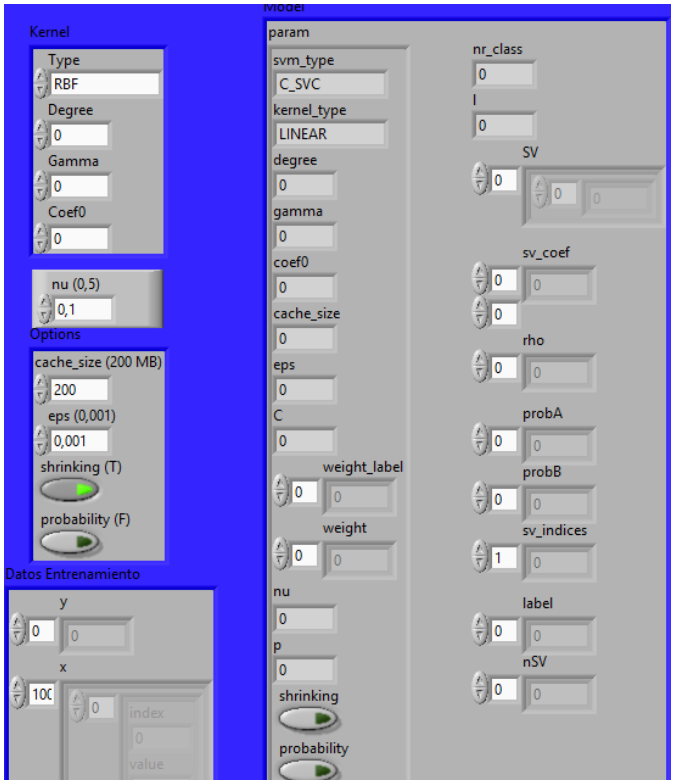


Figura 29.Panel de Configuración OCSVM en el IDS-SCADA

- **Datos de Entrenamiento:** Permite observar los datos de entrenamiento mediante la visualización del vector *x,* así como las etiquetas respectivas disponibles en el vector *y*.

- **Modelo:** En el panel del modelo se pueden observar los distintos parámetros que se obtienen una vez el sistema ha sido entrenado y que son entregados al bloque de predicción para la clasificación de datos en tiempo real. El dato más relevante en este caso corresponde a los vectores de soporte que se obtuvieron, ya que son estos los que determinan la frontera de decisión.

En lo que respecta al diagrama de bloques respectivos, la figura 30 muestra la configuración del bloque de entrenamiento; mientras la figura 31 muestra las conexiones asociadas al bloque de predicción. Como se observa, en la etapa de entrenamiento se requiere acceder a un archivo de texto plano donde se encuentran los datos que posteriormente son formateados, de modo que sean compatibles con la estructura interna del bloque de predicción.

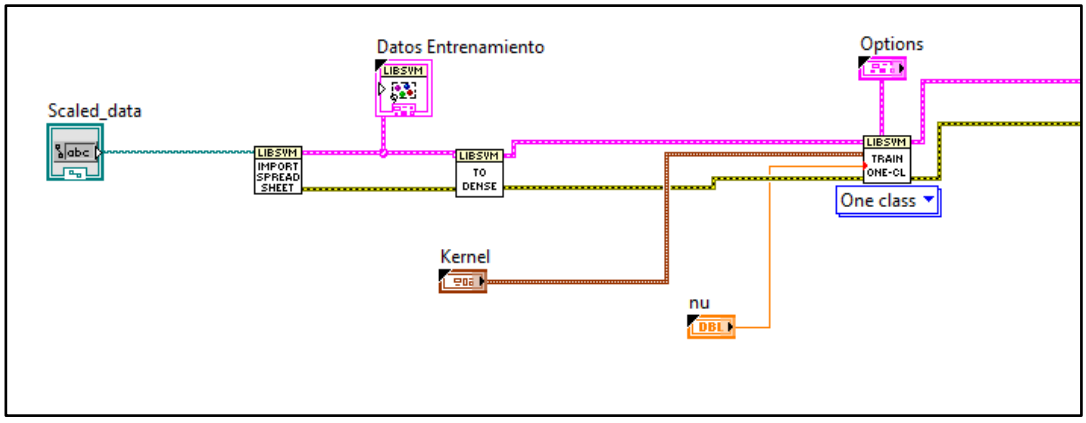


Figura 30.Configuración Bloque de Entrenamiento LIBSVM en Labview

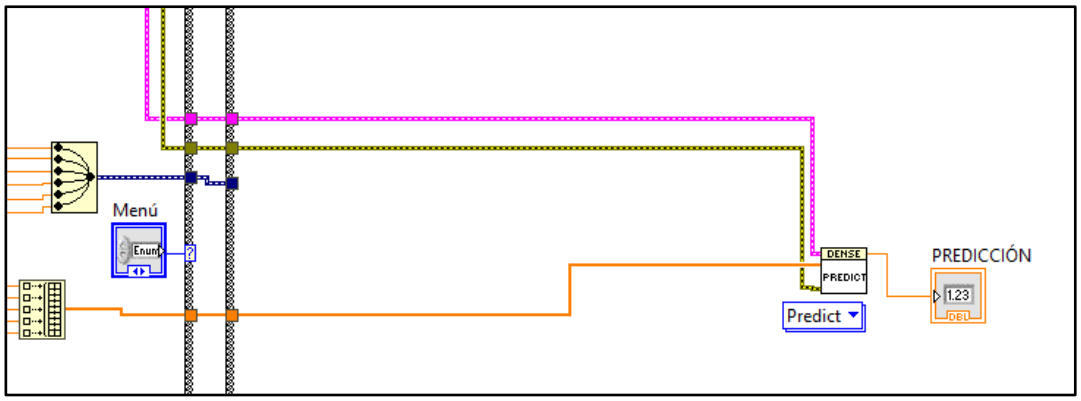


Figura 31.Conexión Bloque de Predicción

Finalmente, el programa para el cálculo de distancia crítico se encuentra disponible en el Apéndice F. En este programa en la primera etapa se hace el cálculo de distancia mediante la implementación de ciclos y operaciones entre arreglos; mientras que en la segunda etapa se utiliza la distancia calculada para determinar el tipo de variable o variables que ha resultado afectada. La distancia mínima que ha de ser superada para registrar una anomalía se definió como '1' analizando los variables sin ser escaladas; sin embargo, este valor puede ser alterado dependiendo de las necesidades del proceso.

e. Servidor SCADA Intruso

Con el ánimo de introducir errores en el funcionamiento del PLC y la planta de forma remota, se creó una aplicación que permite la escritura o modificación de los parámetros críticos de operación en tiempo real. En síntesis, la aplicación intrusión se basa en un servidor SCADA que posee características muy similares al panel del

IDS-SCADA, con la diferencia que este tiene la posibilidad de escribir y no solo de leer.

Con el fin evitar que los cambios realizados afecten el proceso de forma continua, escenario que no es deseable, puesto que las modificaciones deben aparecer en momentos puntuales, se dotó a cada uno de los controles con un botón que habilita la escritura solo cuando este es presionado; esto permite que las modificaciones se inserten en el PLC y se pueda observar que efectos producen a corto o largo plazo, especialmente cuando se modifican las marcas de estados. Un ejemplo de lo dicho anteriormente, sería la activación de una marca sin respetar la secuencia de encendido que se establece en el programa. Esto podría ocasionar que las salidas se activen de forma incoherente, haciendo que el proceso se torne caótico.

La figura 32 muestra el panel de control de la interfaz intrusa y en el Anexo I se encuentra disponible el diagrama de bloques respectivo.

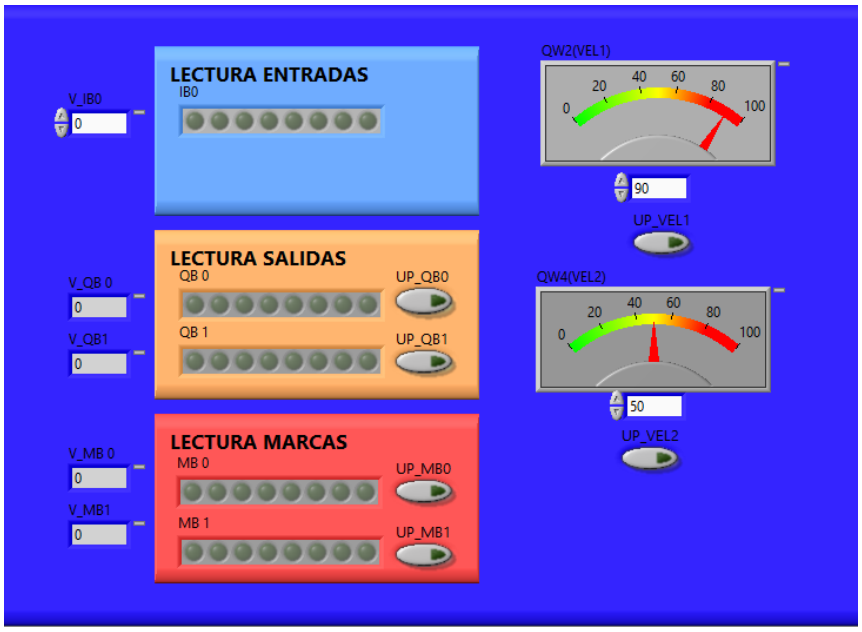


Figura 32.Panel de Control Servidor Intruso

5.Validación y Resultados Finales

Hasta ahora se ha hecho una descripción de la propuesta y los elementos que se usaron para la verificación de la eficiencia de la misma. En este capítulo se hará un análisis detallado de todo el proceso de validación y cuáles fueron los resultados obtenidos durante el mismo.

Para la validación de los resultados, se utilizó la metodología propuesta en la tabla 7, en donde además de las actividades realizadas se hace una breve descripción de las mismas.

No	Actividad	Descripción
1	Captura de datos	Se captura una muestra de datos significativa, en donde esté representado todo el patrón de comportamiento normal del proceso.
2	Eliminación de datos reincidentes	Se eliminan los datos repetidos, dejando solo las combinaciones exclusivas de los distintos parámetros
3	Escalamiento de los datos	Se escalan los diferentes descriptores considerando los criterios descritos en los capítulos anteriores.
4	Gráfica de los datos	Se hace una gráfica de los datos en un espacio tridimensional para observar su patrón de comportamiento.
5	Selección del Kernel	Se escoge el Kernel que ofrezca la mejor alternativa para la separación de los datos.
6	Selección de valores para los hiperparámetros	Utilizando una metodología de validación de entrenamiento, se hace un barrido de parámetros para encontrar el modelo más óptimo.
7	Captura de muestras de validación	Se obtienen n cantidad de muestras para validación
8	Análisis de resultados finales	Se calcula la eficiencia del IDS-SCADA

Tabla 7. Metodología de Validación

5.1. Captura de Datos

En aras de obtener un conjunto de datos de entrenamiento que fuese lo suficientemente grande para representar el comportamiento normal del proceso analizado se configuró en la aplicación IDS una velocidad de muestreo de 10 milisegundos; de esta manera es posible además de capturar los valores representativos de cada estado, obtener información de los valores correspondientes a las transiciones. Dichas transiciones corresponden, para el caso del proceso analizado, a las activaciones y desactivaciones de los sensores indicando cambios en la secuencia operativa.

En total se obtuvieron más de 30000 muestras iniciales, que luego serían procesadas para entrenar la OCSVM. El periodo de captura fue de aproximadamente 6 horas; tiempo en el que se realizaron aproximadamente 100 ciclos del proceso.

5.2. Eliminación de Datos Reincidentes

Debido a que ciertos algoritmos de aprendizaje de máquinas no operan correctamente a causa de la no independencia lineal de las filas en la matriz de descriptores, la muestra inicial de los datos que había sido capturada fue filtrada para eliminar los datos reincidentes o repetidos. Una vez los datos capturados fueron filtrados el conjunto de datos para entrenamiento se redujo a un total de 1042. En la figura 33 se observa una porción de los datos no escalados en un visor de arreglos de Labview.

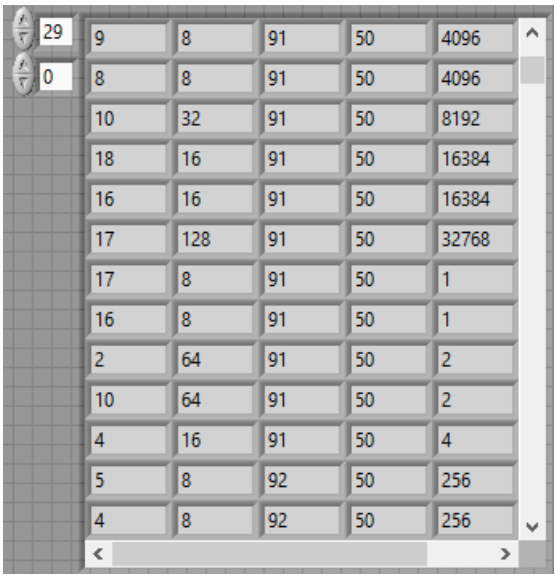


Figura 33.Arreglo de Datos de entrenamiento no Escalados

5.3. Escalamiento de los Datos

Los datos fueron normalizados aplicando los criterios explicados en capítulos anteriores con el fin de dar el mismo peso a los descriptores y tener una mejor predicción. En la figura 34 se puede observar una porción de los mismos en un visor de arreglos de Labview.

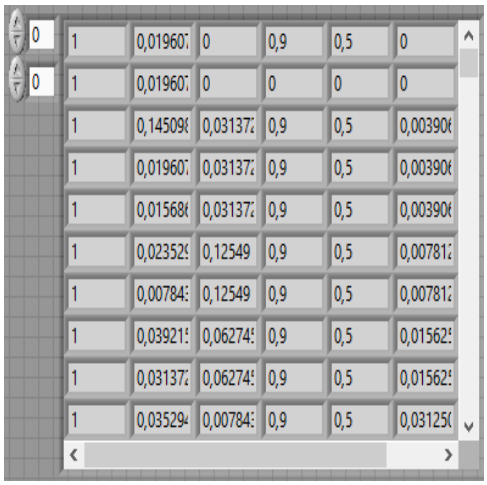


Figura 34.Visor de Datos de Entrenamiento Escalados

5.4. Grafica de los datos

Para un arreglo de 5 dimensiones que corresponde a la cantidad de descriptores disponibles se hace imposible obtener un gráfico de forma directa. Para sortear dicha situación, se agruparon los diferentes descriptores en los 3 tipos de datos soportados por el PLC, es decir, como entradas, salidas y marcas.

Para una mejor observación de los datos, se utilizó la función *scatter3* del programa Matlab. La vista 3D se observa en la figura 35.

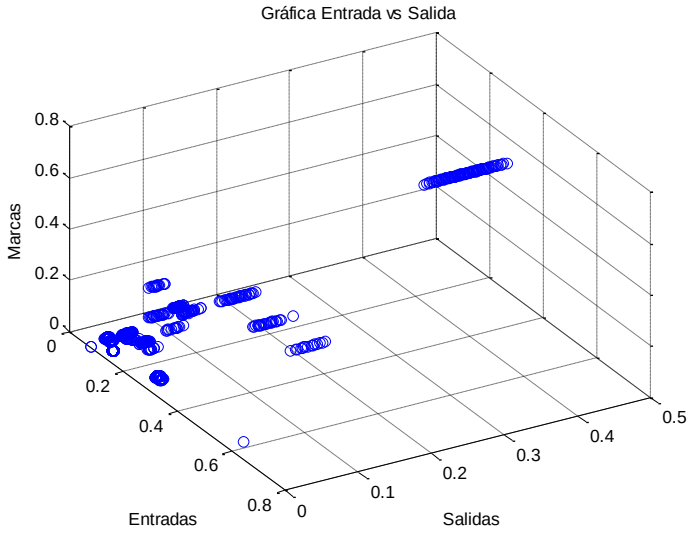


Figura 35.Vista 3D de los datos

En un primer análisis se puede observar que los datos aparecen bastante concentrados en diferentes regiones. Esto era de esperarse, considerando el barrido de los parámetros y que además se está modelando un proceso secuencial.

Se puede observar también que las variables tipo Marca y tipo Salida presentan cambios abruptos haciendo que el conjunto de los datos se desplace. Esto se debe a que dichas variables incrementan en potencias de 2; por lo que a cada cambio de estado o activación de una salida de orden mayor le corresponde un número cada vez más grande. Dicha característica es ventajosa para la detección de intrusiones debido a que una pequeña variación en los parámetros repercute en cambios abruptos que faciliten la detección de anomalías.

En las gráficas 36 al 38 se puede tener una vista más detallada de cada uno de los planos.

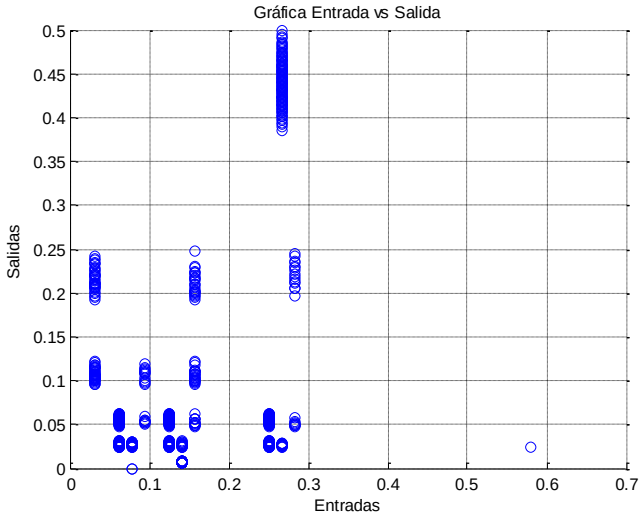


Figura 36.Gráfica de Entradas vs Salidas

En la figura 36 se nota claramente la variación abrupta de los datos como consecuencia del cambio de las salidas.

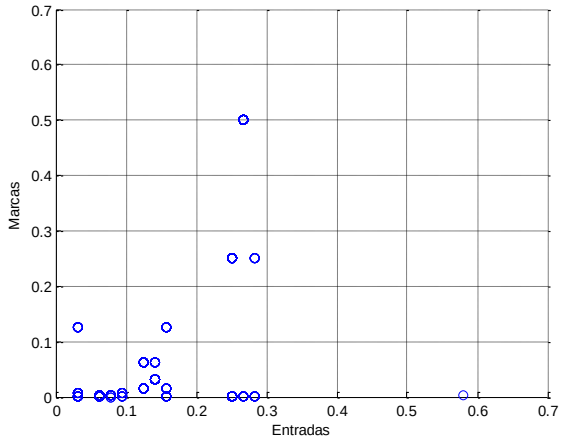


Figura 37.Gráfica de Entradas vs Marcas

Para el caso de la figura 37 aplica el mismo análisis de figura 36; se observa que la variación de las variables tipo Marca es bastante grande comparada con las variables tipo Entrada.

Finalmente, tenemos la gráfica 38 donde se observa las grandes variaciones que se dan en la relación Marcas vs Salidas.

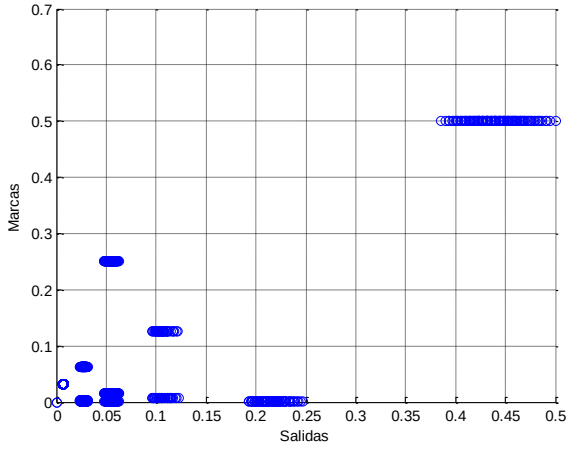


Figura 38.Gráficas de Marcas vs Salidas

5.5. Selección de Función Kernel

Como se mencionó en el capítulo 2, la función Kernel sirve para proyectar los datos en análisis en otra dimensión con el fin de poder mejorar la clasificación de los mismos.

Para el caso en estudio, los datos no muestran un comportamiento lineal, por lo que de entrada se rechaza el Kernel lineal como una posible alternativa de separación.

En lo que respectiva a los Kernel restantes, de acuerdo con la literatura reportada [3][55] la mejor opción es el Kernel RBF para detección de anomalías; este permite la clasificación de datos con distribuciones complejas como el caso analizado. La figura 39 muestra un ejemplo de clasificación para la detección de anomalías basada en este tipo de Kernel, en donde los datos regulares o normales se encuentran agrupados en regiones diferentes y aun así el algoritmo fue capaz de calcular varias fronteras de detección.

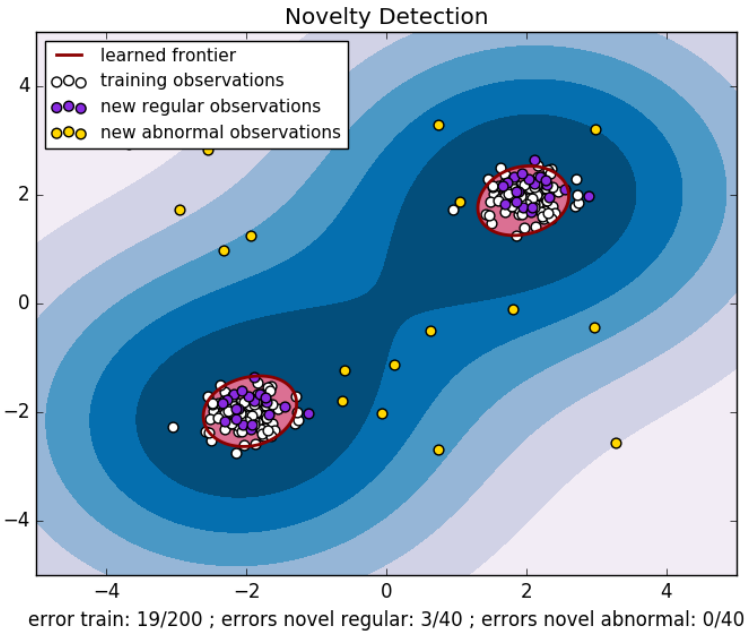


Figura 39.Ejemplo separación de Muestras con Kernel RBF[55]

Retomando la ecuación 3, se puede observar que uno de los parámetros críticos para este tipo de Kernel es el parámetro *Gamma*, mientras que el parámetro *nu* va ligado directamente al modelo de la OCSVM para la clasificación monoclase. La forma como se seleccionaron dichos parámetros, es el tema de la siguiente sección.

5.6. Selección de parámetros

5.6.1. Descripción General

Los parámetros *nu* y *Gamma* determinan que tan lejos o cerca se encuentra la frontera de decisión con respecto a los datos de entrenamiento. Su rango de valores oscila entre 0 y 1; un valor muy pequeño de ambos significa que la frontera de decisión es muy cercana a los datos o que la cantidad de vectores de soporte seleccionados resulta ser mayor. Para el caso concreto de este trabajo, se optó por seleccionar valores relativamente pequeños de ambos valores de forma que pudieran detectarse datos anómalos cuyas variaciones fueran muy pequeñas con respecto a los datos de entrenamiento.

Con el fin de determinar la mejor combinación de ambos se diseñó una aplicación que hace un barrido de ambos parámetros mediante la implementación de dos ciclos For anidados, en donde el ciclo más externo modifica el valor de *Gamma* y el ciclo más interno modifica el valor de *nu*.

Por otro lado, para poder calcular la eficiencia de los parámetros seleccionados se utilizó la metodología "Cross-Validation". Esta metodología divide el conjunto de datos de entrenamiento en N subconjuntos disyuntos; Uno de los grupos se usa para la validación y el resto se utiliza para el entrenamiento. Los resultados obtenidos en esta etapa se analizan finalmente mediante una matriz de confusión, la cual informa la eficiencia total de la clasificación para cada conjunto de parámetros.

En la figura 40, se puede observar el panel de control de la aplicación diseñada y en el anexo H se encuentra disponible el diagrama de bloques respectivo.

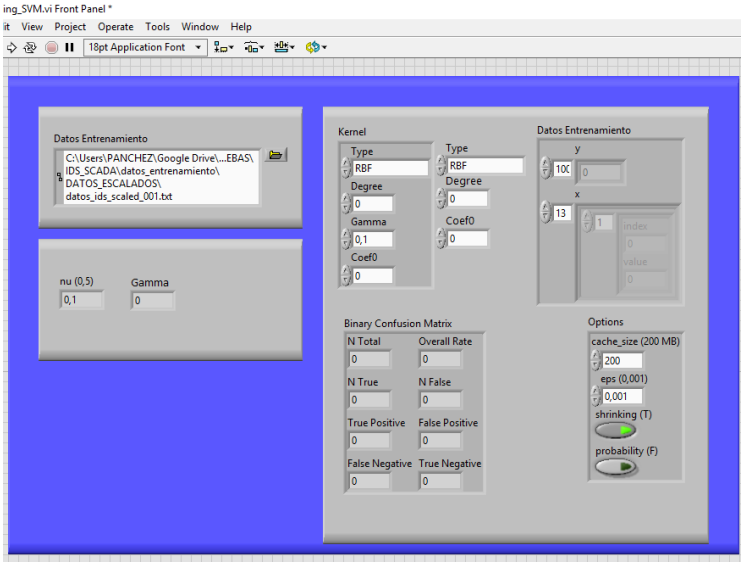


Figura 40.Aplicación para Calculo de Parámetros y Eficiencia de Entrenamiento

5.6.2. Ejecución de Algoritmo y Análisis

Para la obtención de los parámetros se configuró el algoritmo de modo que pudieran generar de forma automática 10 subconjuntos. De estos 10 subconjuntos, uno de ellos se utiliza para la validación, mientras el 90 % restante se asigna al entrenamiento de la OCSVM.

Para cada uno de los ciclos se programaron en total 20 iteraciones, dando como resultado una matriz de 400 filas, en donde se evalúan los siguientes parámetros:

- Número total de datos que se utilizaron.
- El número total de muestras que se etiquetaron como verdaderas y que para este caso corresponden al 100% por ser un algoritmo que solo se entrena con este tipo de datos.
- El número total de muestra que se etiquetaron como falsas que para todos los casos debe ser 0.
- La eficiencia total calculada.
- El número total de Verdaderos positivos que corresponden a la cantidad de muestras que se etiquetaron como positivas y resultaron ser positivas.
- El número total de Falsos Positivos que se detectaron. En la etapa de entrenamiento este valor no aplica, debido que en el análisis todas las muestras se etiquetaron como positivas y no existe la posibilidad de que una muestra negativa se clasifique como positiva.
- El número total de Falsos negativos que corresponden a las muestras que se etiquetaron como positivas y se clasificaron como negativas.
- El número de Verdaderos negativos tampoco aplica en el análisis debido a que no existen etiquetas de este tipo en los datos de entrenamiento.

Una porción de los resultados obtenidos, donde se puede observar la máxima eficiencia calculada con los respectivos valores de *nu* y *gamma* se muestra en la tabla 8.

nu	Gamma	Total M.	TP	TN	Eficiencia	True P.	False P	False N	True N.
0,1	0,07	104	104	0	0,895393	0,895393	#NV	0,104607	#NV
0,11	0,07	104	104	0	0,886756	0,886756	#NV	0,113244	#NV
0,12	0,07	104	104	0	0,880038	0,880038	#NV	0,119962	#NV
0,13	0,07	104	104	0	0,867562	0,867562	#NV	0,132438	#NV
0,14	0,07	104	104	0	0,856046	0,856046	#NV	0,143954	#NV
0,15	0,07	104	104	0	0,845489	0,845489	#NV	0,154511	#NV
0,16	0,07	104	104	0	0,837812	0,837812	#NV	0,162188	#NV

0,17	0,07	104	104	0	0,827255	0,827255	#NV	0,172745	#NV
0,18	0,07	104	104	0	0,817658	0,817658	#NV	0,182342	#NV
0,19	0,07	104	104	0	0,81094	0,81094	#NV	0,18906	#NV
0,2	0,07	104	104	0	0,799424	0,799424	#NV	0,200576	#NV
0,01	0,08	104	104	0	0,991363	0,991363	#NV	0,008637	#NV
0,02	0,08	104	104	0	0,977927	0,977927	#NV	0,022073	#NV
0,03	0,08	104	104	0	0,96929	0,96929	#NV	0,03071	#NV
0,04	0,08	104	104	0	0,954894	0,954894	#NV	0,045106	#NV
0,05	0,08	104	104	0	0,949136	0,949136	#NV	0,050864	#NV
0,06	0,08	104	104	0	0,93762	0,93762	#NV	0,06238	#NV
0,07	0,08	104	104	0	0,924184	0,924184	#NV	0,075816	#NV
0,08	0,08	104	104	0	0,918426	0,918426	#NV	0,081574	#NV
0,09	0,08	104	104	0	0,90499	0,90499	#NV	0,09501	#NV
0,1	0,08	104	104	0	0,898273	0,898273	#NV	0,101727	#NV
0,11	0,08	104	104	0	0,890595	0,890595	#NV	0,109405	#NV
0,12	0,08	104	104	0	0,878119	0,878119	#NV	0,121881	#NV

Tabla 8. Resultados etapa de Entrenamiento

De todas las combinaciones posibles la mejor combinación de parámetros para una eficiencia del **0,991363** se da cuando: nu igual 0,01y gamma igual a 0.08. Dichos parámetros se utilizaron en la etapa de detección en tiempo real para verificar el buen comportamiento del sistema.

5.7. Captura de Muestras de Validación

Con el fin de generar una matriz de datos que pudiera ser analizada posteriormente por el IDS-SCADA y de esa manera validar su eficiencia, se generaron un total de 145 muestras a través de la captura de datos en tiempo real de la manipulación indebida de parámetros por parte del servidor intruso.

En total dichos datos se componen de: 89 muestras que se les etiquetó manualmente como anormales debido a que inducían comportamientos erráticos en la planta y 56 que permitían su operación normal. Una porción de los datos se puede visualizar en la tabla 9, donde la primera columna identifica un paquete corrupto con un ‘1’ y un paquete normal con un ‘0’.

Etiqueta	Entradas	Salidas	Velocidad1	Velocidad 2	Marcas
1	0,125	0,439216	1,010526	0,963636	0,03125
0	0,15625	0,062745	0,947368	0,909091	0,03125
0	0,140625	0,031373	1,042105	0,909091	0,125
1	0,0625	0,031373	0,936842	0,909091	0,007812

1	0,0625	0,282353	0,989474	0,963636	0,03125
0	0,0625	0,031373	0,989474	0,963636	0,007812
1	0,140625	0,007843	0,936842	0,909091	0,0625
1	0,0625	0,031373	0,947368	0	0,007812
1	0,28125	0	0,947368	0,909091	0
1	0	0,12549	0,989474	0,963636	0
0	0,28125	0,062745	0,957895	0,909091	0,5
0	0,078125	0	0	0	0
1	0,125	0	0,957895	0,909091	0
1	0,28125	0	0,989474	0,963636	0

Tabla 9. Porción de Datos de Validación

5.8. Análisis de Resultados Finales

Para la validación final de los datos obtenidos se diseñó una aplicación en Labview, que permitiera barrer el conjunto de datos de validación y finalmente entregar los resultados en una matriz de confusión. El panel de control se muestra en la figura 41.

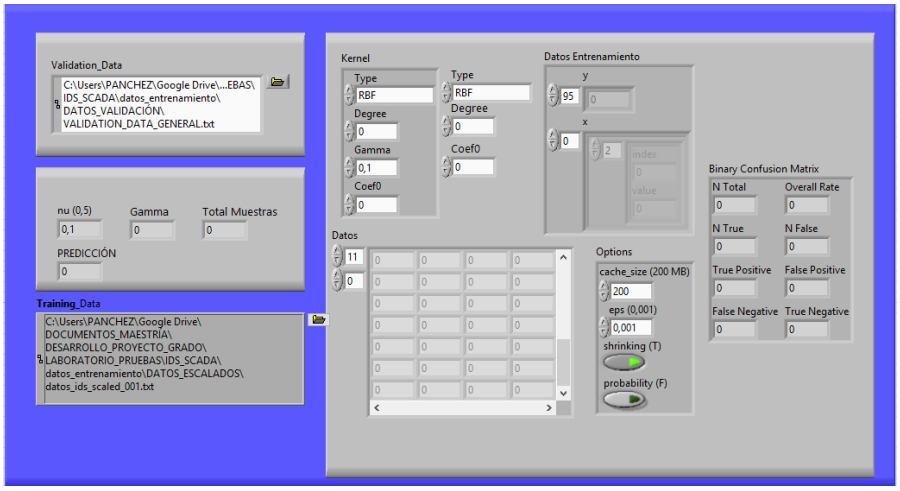


Figura 41. Panel de Control Aplicación de Validación

Utilizando los parámetros calculados en la etapa de entrenamiento se pudo observar que el modelo de predicción es bastante eficiente si se consideran los resultados entregados por la matriz de confusión disponibles en la figura 42.

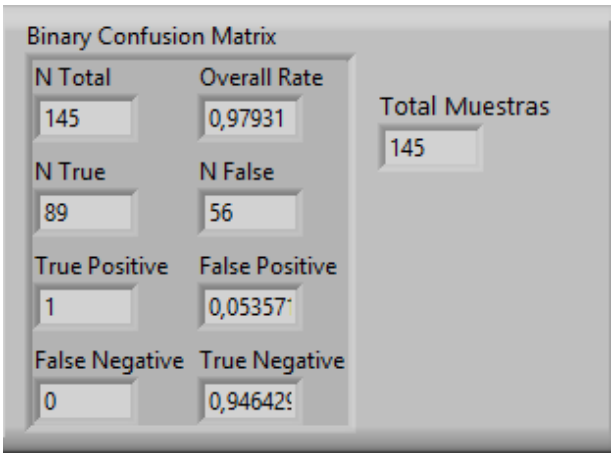


Figura 42. Matriz de Confusión de Validación

Analizando los resultados de la matriz se puede observar que el total de los datos etiquetados como intrusos fue clasificado correctamente; esto se puede evidenciar en la tasa de verdaderos positivos que para este caso corresponde al 100%; sin embargo, el 5% de los datos que se etiquetaron como normales fueron clasificados erróneamente como se evidencia en la tasa de falsos positivos. La eficiencia total del modelo de validación se calculó en 97.9% aplicando la ecuación x, lo cual cumple satisfactoriamente con las expectativas del modelo.

$$Eficiencia(\%) = \frac{(Tasa\ Verdaderos\ Positivas + Tasa\ Verdaderos\ Negativos)}{2}$$

Ecuación 5. Ecuación de Eficiencia

En la figura 43 se observa un gráfico comparativo de las distancias calculadas normalizadas para cada una de las variables con respecto a la predicción obtenida. Este análisis permite determinar qué tan lejos se encontraba el dato de validación con respecto a las muestras de entrenamiento en la clasificación.

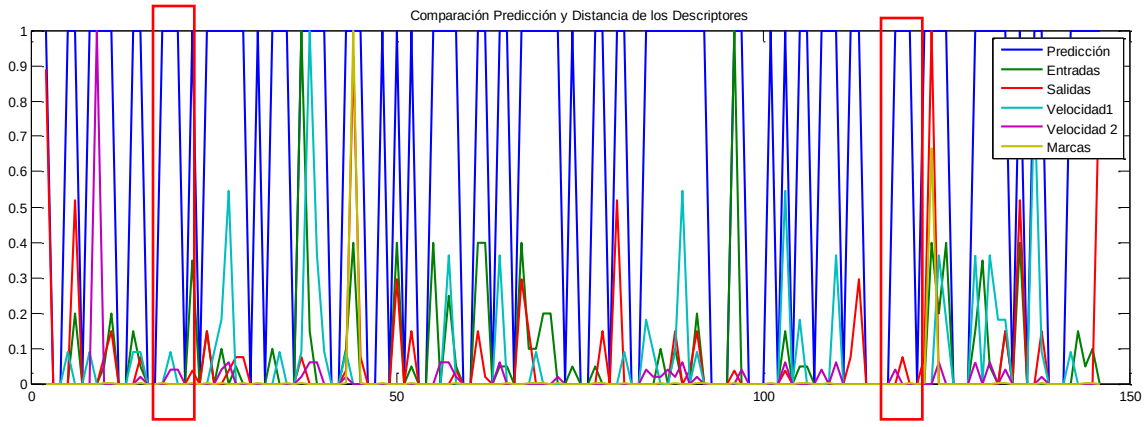


Figura 43. Predicción y Distancias Descriptores

Se puede concluir del gráfico, que no se requirió de distancias significativas con respecto a los datos de entrenamiento para que el dato fuese clasificado como intruso. En algunos casos, bastó con variaciones menores a 0.1 en el modelo normalizado, para que el modelo de predicción acertara en la clasificación.

En la tabla 10 se puede observar algunas de las muestras de las distancias entregadas por la aplicación no normalizadas. Se puede observar que las mayores distancias se encuentran en las variables tipo salida o marca, dado su comportamiento exponencial; mientras que en otros casos basto una diferencia de '1' (Filas resaltadas en amarillo) para que dato fuese clasificado como anómalo.

Predicción	Distancia Entradas	Distancia Salidas	Distancia Velocidad1	Distancia Velocidad2	Distancia Marcas
1	0	8	0	0	4
0	0	0	0	0	0
1	0	0	0	0	4
0	0	0	0	0	0
1	2	0	0	0	0
1	0	0	1	0	0
1	0	0	0	0	4
0	0	0	0	0	0
1	20	8	1	1	1
1	3	0	11	3	0
1	0	0	4	3	0
1	0	0	1	0	0
0	0	0	0	0	0
0	0	0	0	0	0
1	2	4	1	1	4
1	8	100	0	0	6144
1	0	8	0	0	1
0	0	0	0	0	0
0	0	0	0	0	0
1	0	0	0	0	4

Tabla 10.Comparación Predicción vs Distancias por Descriptor

Finalmente, las filas resaltadas en rojo representan las anomalías en las variables internas de control del programa. Modificaciones en este tipo de variables no son monitoreadas por los sistemas tradicionales, por lo que representa una ventaja del sistema IDS-SCADA al alertar sobre cambios que podrían no tener efectos mediáticos pero si a largo plazo.

6.Conclusiones y Trabajos Futuros

6.1. conclusiones

El análisis de la metodología implementada y de los resultados obtenidos permite concluir que el modelo propuesto es una buena alternativa para la identificación de intrusiones en los sistemas de control de las plataformas SCADA, especialmente cuando estas anomalías no son perceptibles mediante las metodologías tradicionales basadas en reglas o análisis de tráfico.

Comparado con la eficiencia de los trabajos reportados, se puede decir que el sistema IDS-SCADA propuesto tiene un desempeño comparable; sin embargo, es bueno resaltar que la metodología planteada no solo permite la identificación de una anomalía, sino que además indica en qué tipo de variable se ha presentado. Esto es sin lugar a duda notable, ya que en ninguno de los trabajos reportados se observó una aproximación semejante.

Por otra parte, el sistema IDS-SCADA propuesto no solo permitiría la detección de posibles infiltraciones, sino que podría configurarse como un gestor de alarmas para cualquier tipo de controlador o sistema embebido; indicando cuando estos presentan fallas internas o si las variables sensadas no se encuentran dentro de los rangos permitidos, sin la necesidad de utilizar metodologías basadas en umbrales. La utilización del estándar OPC, permite el acceso a miles de dispositivos de automatización a nivel mundial mientras que otras soluciones solo aplican para protocolos o controladores específicos.

Finalmente, la utilización de las máquinas de soporte vectorial del tipo “One-Class” facilitan enormemente el proceso de entrenamiento al requerir solo un tipo de clase. La velocidad de predicción en un entorno gráfico como Labview resultó ser menor a 1ms con lo que se esperaría que en otro tipo de entornos no gráficos resulte aún más eficiente.

6.2. Trabajos Futuros

Una de las posibles mejoras que podría realizarse al IDS-SCADA consiste en una aplicación de auto calibración que permita la obtención de la mejor combinación de hiperparámetros para la SVM de forma totalmente automática. Esto garantizaría un sistema totalmente autónomo en donde la intervención del ser humano sería mínima. Adicionalmente, la integración de un algoritmo de agrupamiento, permitiría no solo la

detección de intrusos, sino que además clasificaría las posibles alertas generadas de forma totalmente automática según sus características.

Otro de las posibles mejoras que podrían realizarse consiste en convertir el sistema en una plataforma para la prevención de intrusos que no solo detecte anomalías sino a que su vez las filtre, de manera que no puedan llegar hasta el controlador y alterar su funcionamiento. Este reto sería aún más exigente en términos de velocidad de respuesta considerando el volumen de información que viaja a través de las redes y las acciones de control rápidas que son ejecutadas por los PLC.

Finalmente, técnicas de programación paralela permitirían el análisis de sistemas SCADA a gran escala permitiendo la detección de anomalías en cientos o miles de controladores en forma simultánea.

A. Anexo: Programa en Ladder para PLC S7-1200

Totally Integrated Automation Portal

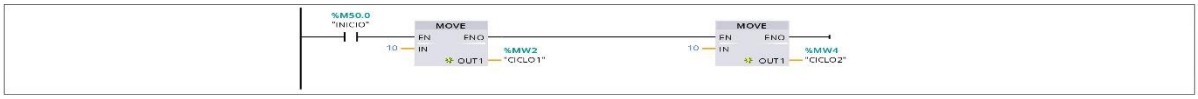
maestría_Andrés_V13 / PLC_1 [CPU 1214C DC/DC/DC] / Bloques de programa

Main [OB1]

Main Propiedades							
General							
Nombre	Main	**Número**	1	**Tipo**	OB	**Idioma**	KOP
Numeración	manual						
Información							
Título	"Main Program Sweep (Cycle)"	**Autor**		**Comentario**		**Familia**	
Versión	0.1	**ID personalizada**					

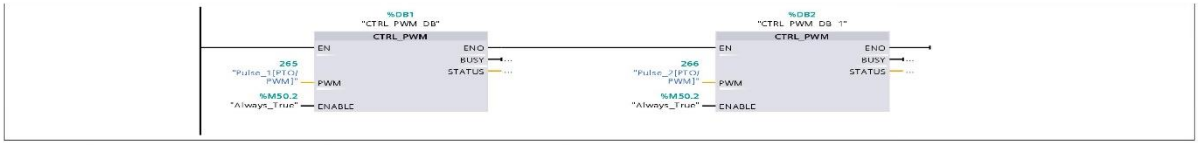
Main			
Nombre	**Tipo de datos**	**Valor predet.**	**Comentario**
Temp			
Constant			

Segmento 1:



Símbolo	Dirección	Tipo	Comentario
"CICLO1"	%MW2	Word	
"CICLO2"	%MW4	Word	
"INICIO"	%M50.0	Bool	

Segmento 2:

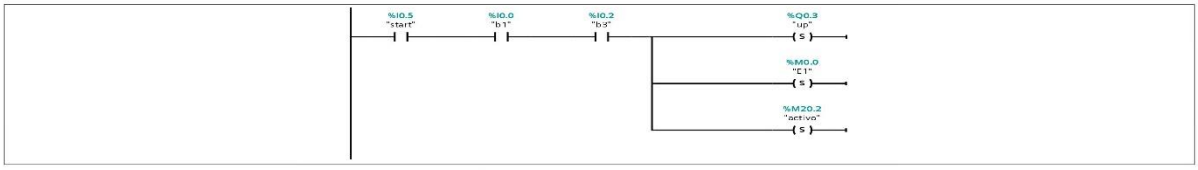


Símbolo	Dirección	Tipo	Comentario
"Always_True"	%M50.2	Bool	
"Pulse_1[PTO/PWM]"	265	HW_PWM	
"Pulse_2[PTO/PWM]"	266	HW_PWM	

Segmento 3:

Símbolo	Dirección	Tipo	Comentario
"conteo1"	%MW22	Word	
"conteo2"	%MW24	Word	
"Vel1"	%QW2	Int	
"Vel2"	%QW4	Int	

Segmento 4:

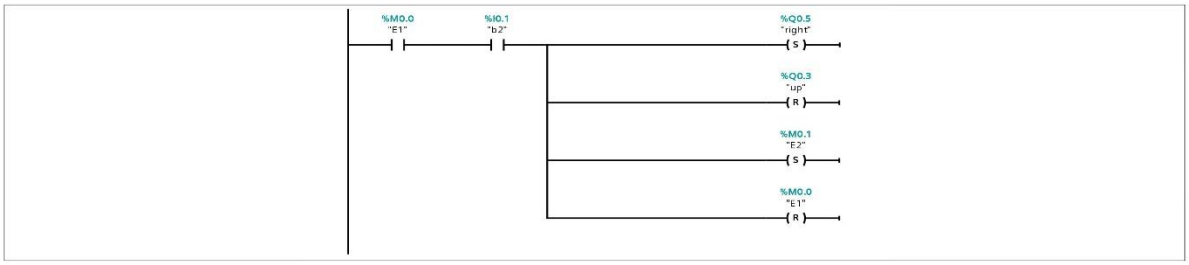


Símbolo	Dirección	Tipo	Comentario
"activo"	%M20.2	Bool	
"b1"	%I0.0	Bool	
"b3"	%I0.2	Bool	

Totally Integrated Automation Portal

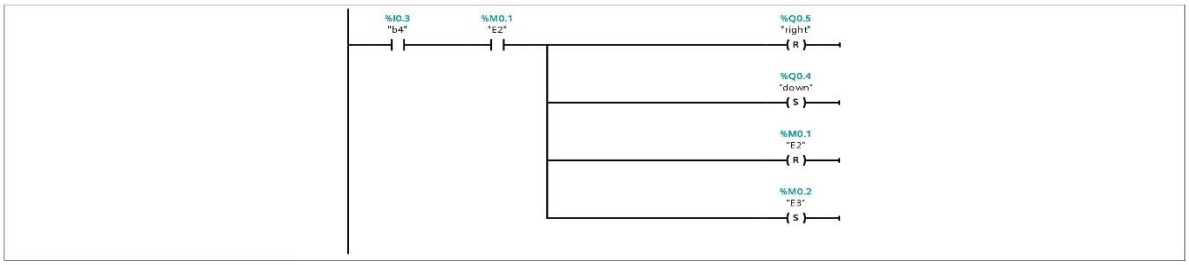
Símbolo	Dirección	Tipo	Comentario
"E1"	%M0.0	Bool	
"start"	%I0.5	Bool	
"up"	%Q0.3	Bool	

Segmento 5:



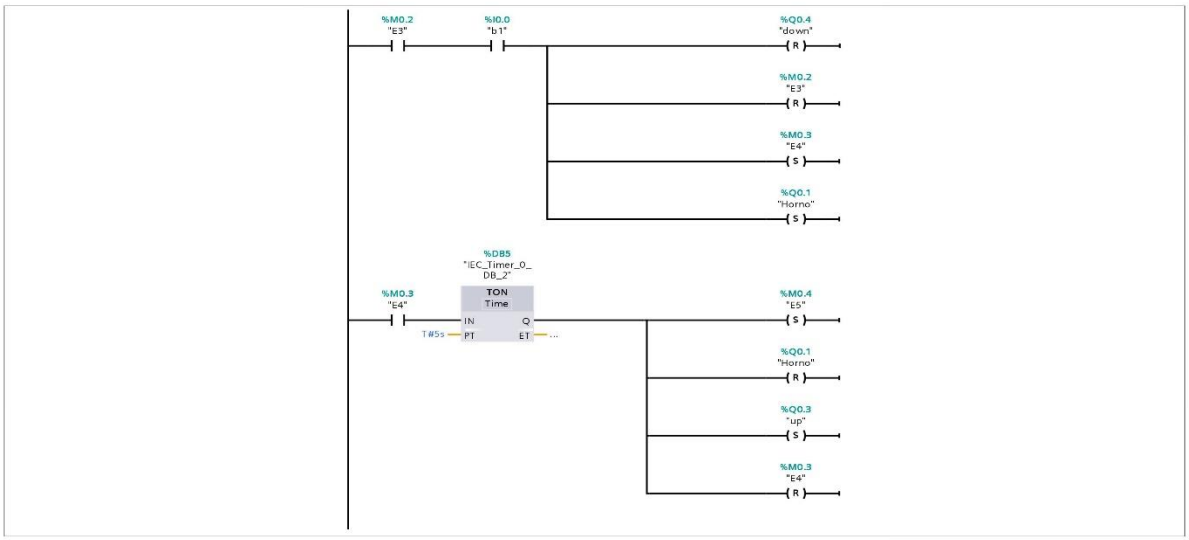
Símbolo	Dirección	Tipo	Comentario
"b2"	%I0.1	Bool	
"E1"	%M0.0	Bool	
"E2"	%M0.1	Bool	
"right"	%Q0.5	Bool	
"up"	%Q0.3	Bool	

Segmento 6:



Símbolo	Dirección	Tipo	Comentario
"b4"	%I0.3	Bool	
"down"	%Q0.4	Bool	
"E2"	%M0.1	Bool	
"E3"	%M0.2	Bool	
"right"	%Q0.5	Bool	

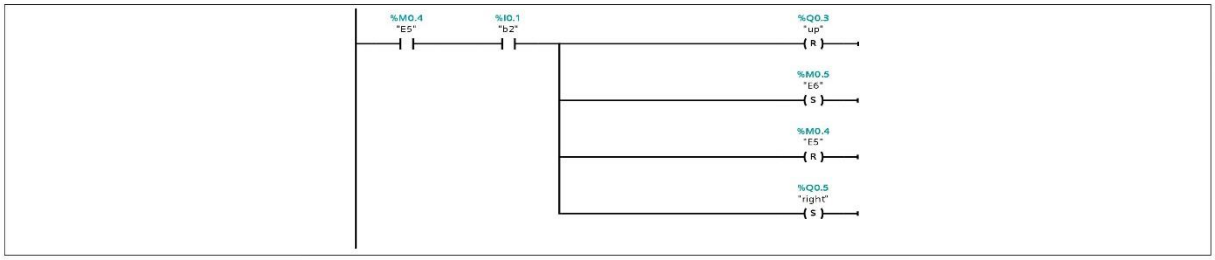
Segmento 7:



Totally Integrated Automation Portal

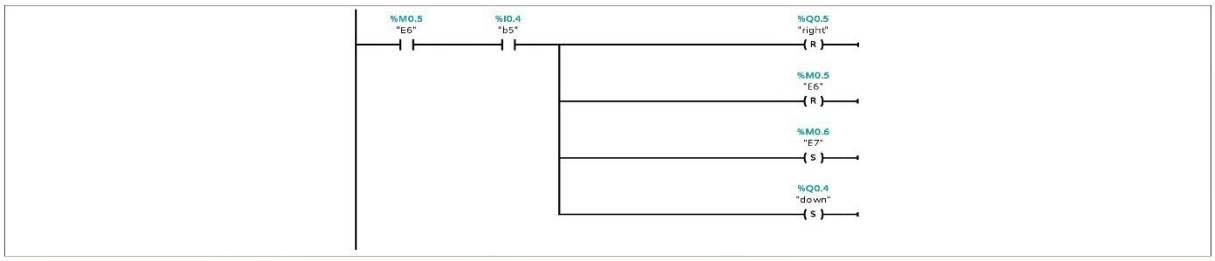
Símbolo	Dirección	Tipo	Comentario
"b1"	%I0.0	Bool	
"down"	%Q0.4	Bool	
"E3"	%M0.2	Bool	
"E4"	%M0.3	Bool	
"E5"	%M0.4	Bool	
"Horno"	%Q0.1	Bool	
"up"	%Q0.3	Bool	

Segmento 8:



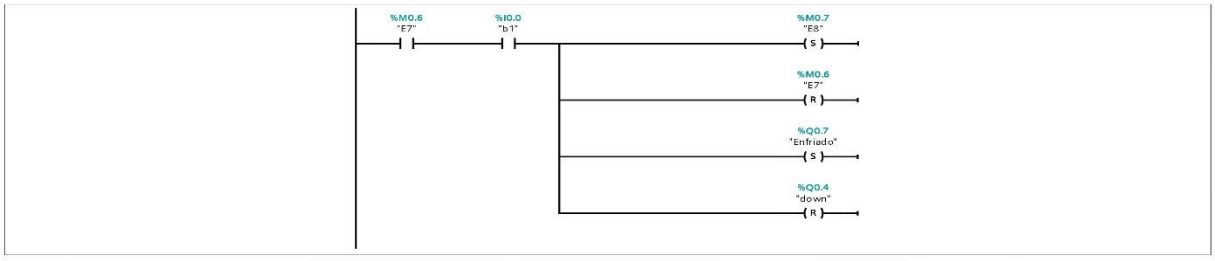
Símbolo	Dirección	Tipo	Comentario
"b2"	%I0.1	Bool	
"E5"	%M0.4	Bool	
"E6"	%M0.5	Bool	
"right"	%Q0.5	Bool	
"up"	%Q0.3	Bool	

Segmento 9:



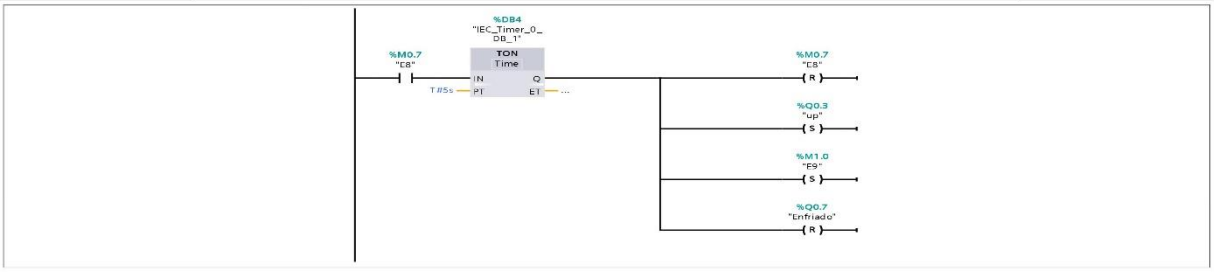
Símbolo	Dirección	Tipo	Comentario
"b5"	%I0.4	Bool	
"down"	%Q0.4	Bool	
"E6"	%M0.5	Bool	
"E7"	%M0.6	Bool	
"right"	%Q0.5	Bool	

Segmento 10:



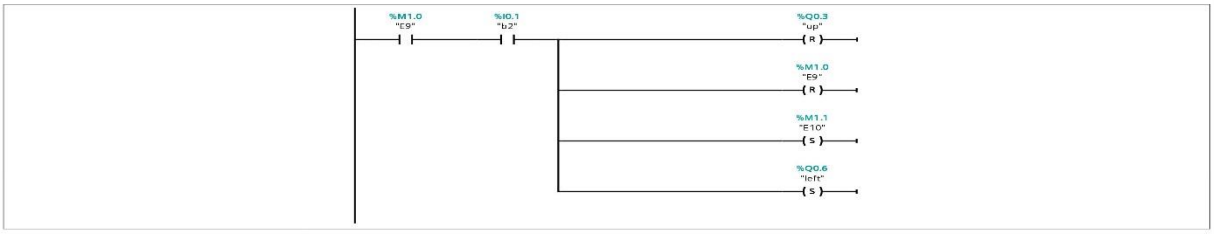
Símbolo	Dirección	Tipo	Comentario
"b1"	%I0.0	Bool	
"down"	%Q0.4	Bool	
"E7"	%M0.6	Bool	
"E8"	%M0.7	Bool	
"Enfriado"	%Q0.7	Bool	

Segmento 11:



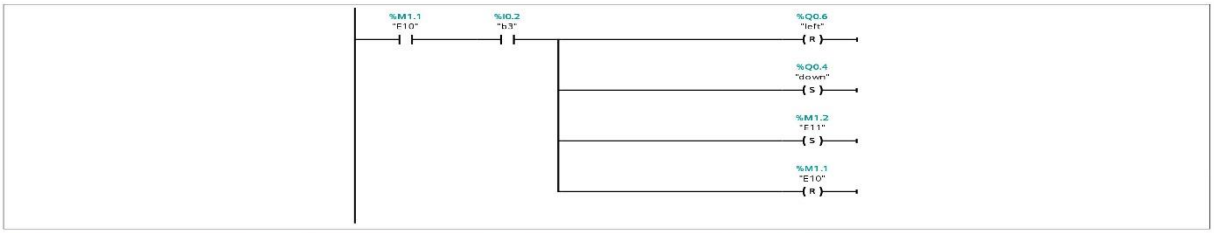
Símbolo	Dirección	Tipo	Comentario
"E8"	%M0.7	Bool	
"E9"	%M1.0	Bool	
"Enfriado"	%Q0.7	Bool	
"up"	%Q0.3	Bool	

Segmento 12:



Símbolo	Dirección	Tipo	Comentario
"b2"	%I0.1	Bool	
"E9"	%M1.0	Bool	
"E10"	%M1.1	Bool	
"left"	%Q0.6	Bool	
"up"	%Q0.3	Bool	

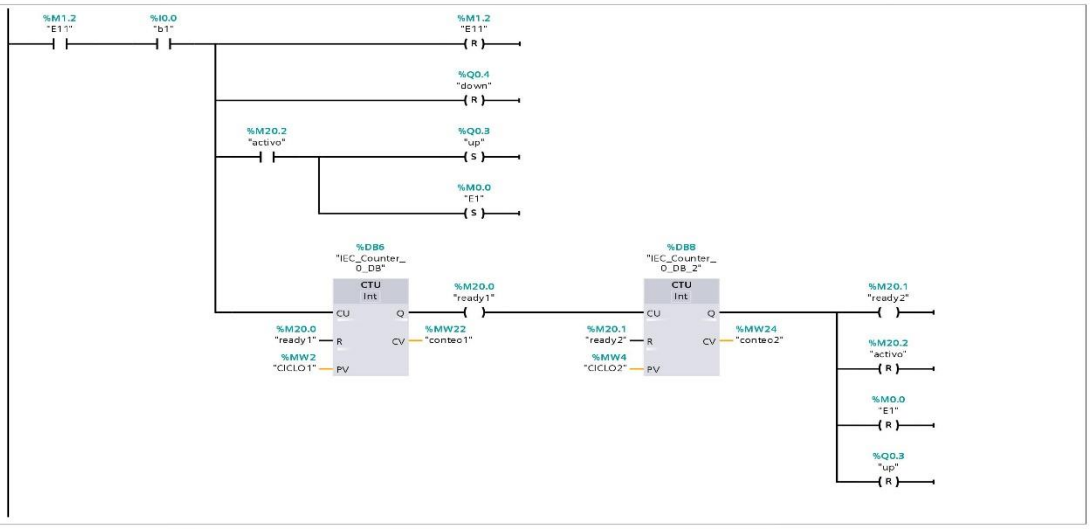
Segmento 13:



Símbolo	Dirección	Tipo	Comentario
"b3"	%I0.2	Bool	
"down"	%Q0.4	Bool	
"E10"	%M1.1	Bool	
"E11"	%M1.2	Bool	
"left"	%Q0.6	Bool	

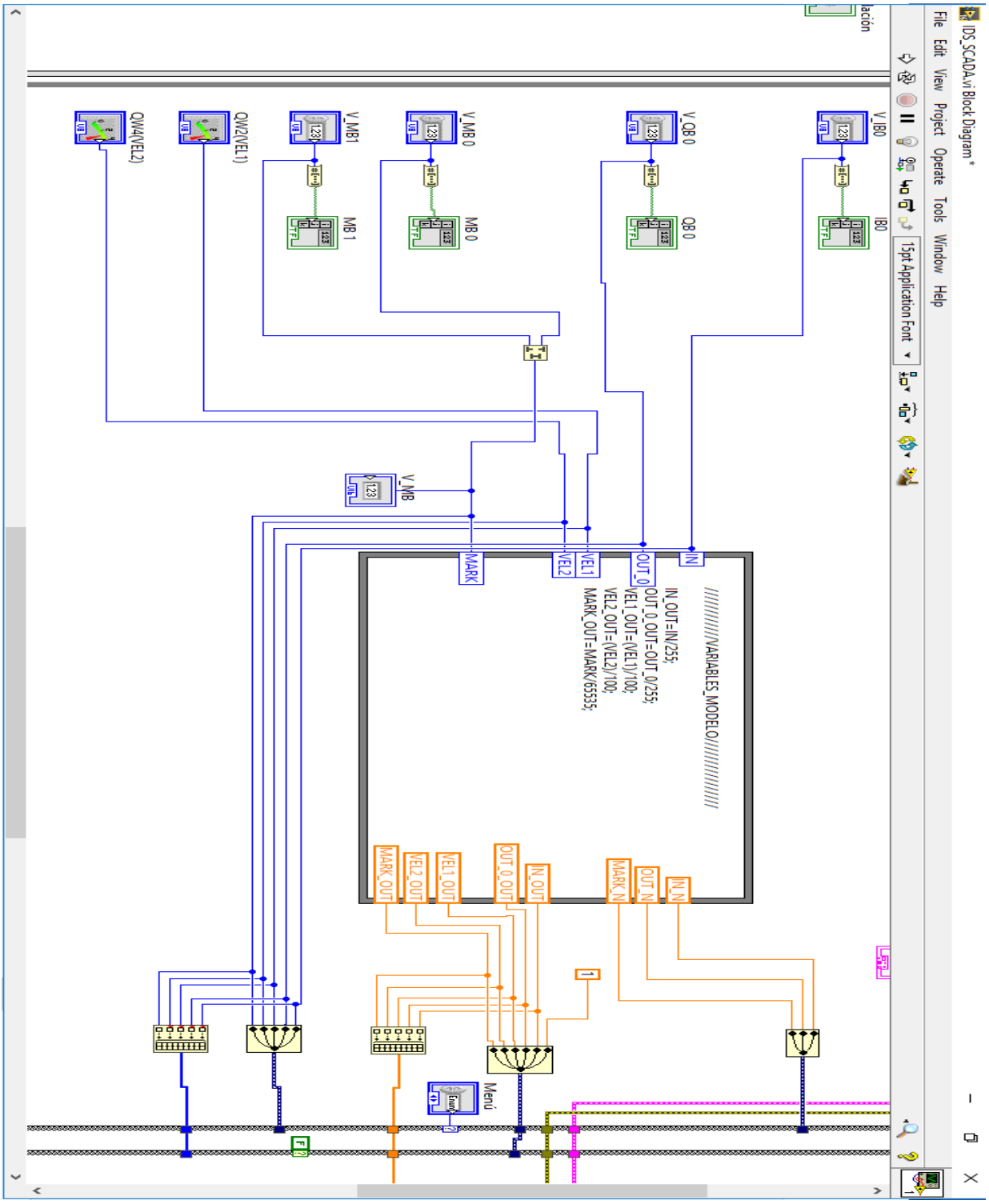
Segmento 14:

Totally Integrated Automation Portal

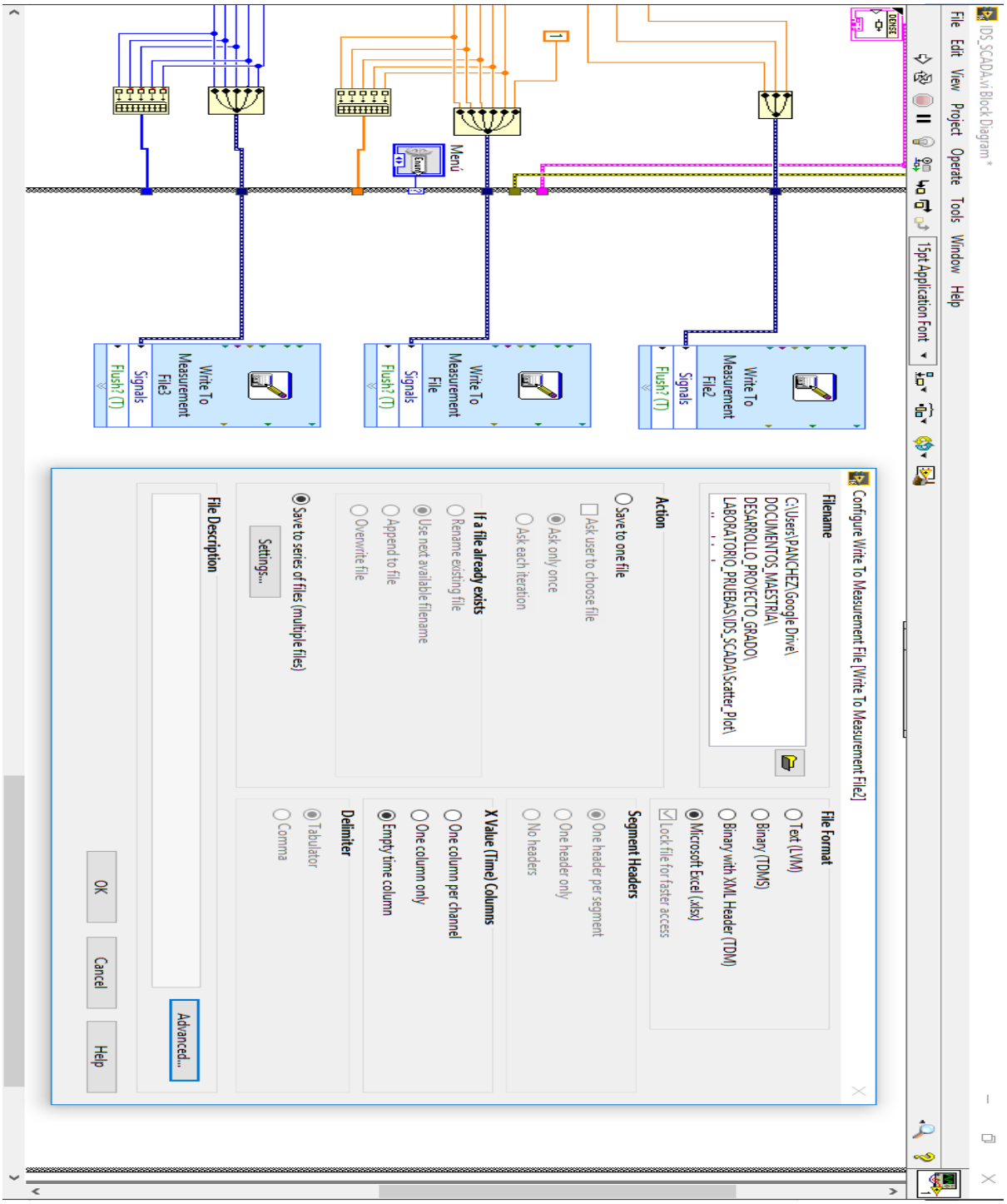


Símbolo	Dirección	Tipo	Comentario
"activo"	%M20.2	Bool	
"b1"	%I0.0	Bool	
"CICLO1"	%MW2	Word	
"CICLO2"	%MW4	Word	
"conteo1"	%MW22	Word	
"conteo2"	%MW24	Word	
"down"	%Q0.4	Bool	
"E1"	%M0.0	Bool	
"E11"	%M1.2	Bool	
"ready1"	%M20.0	Bool	
"ready2"	%M20.1	Bool	
"up"	%Q0.3	Bool	

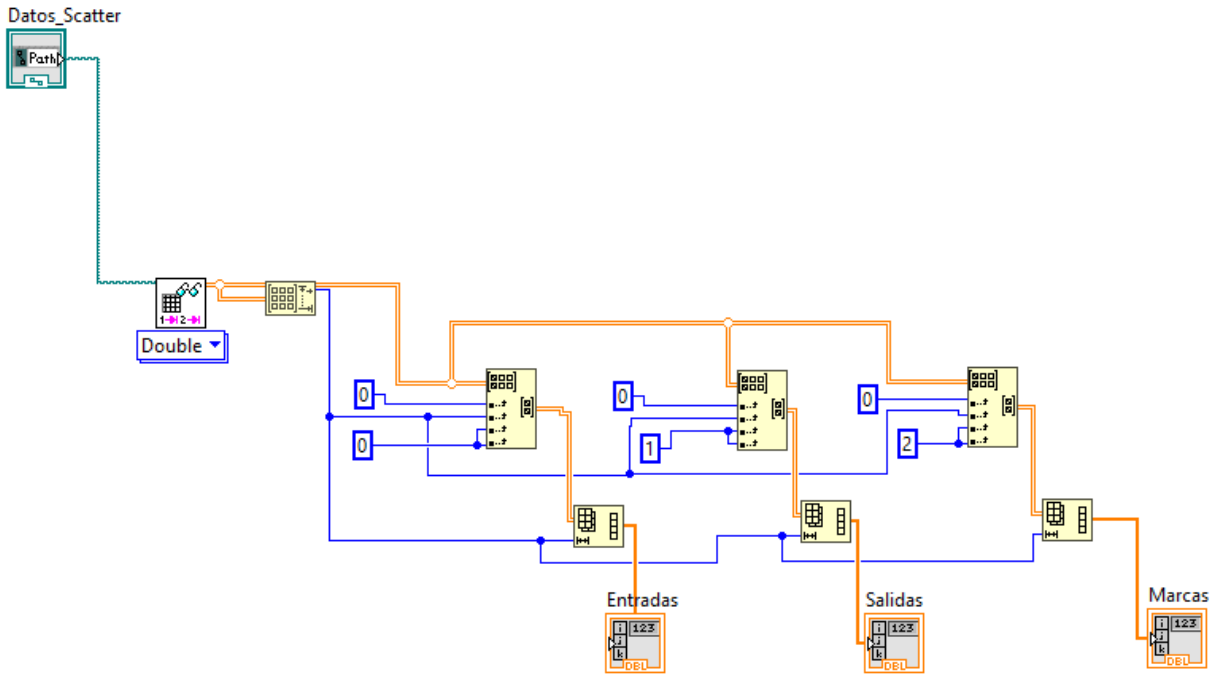
B. Anexo: Captura de Datos IDS-SCADA



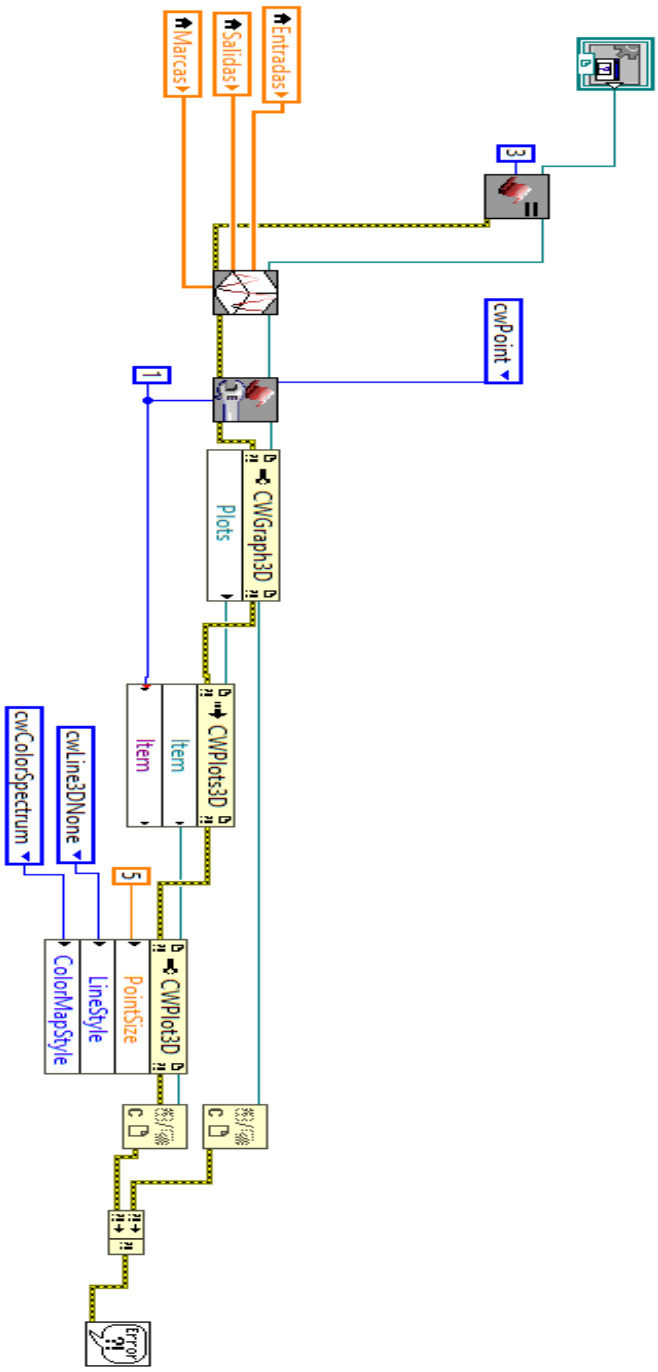
C. Anexo: Captura de Datos de Entrenamiento y Validación



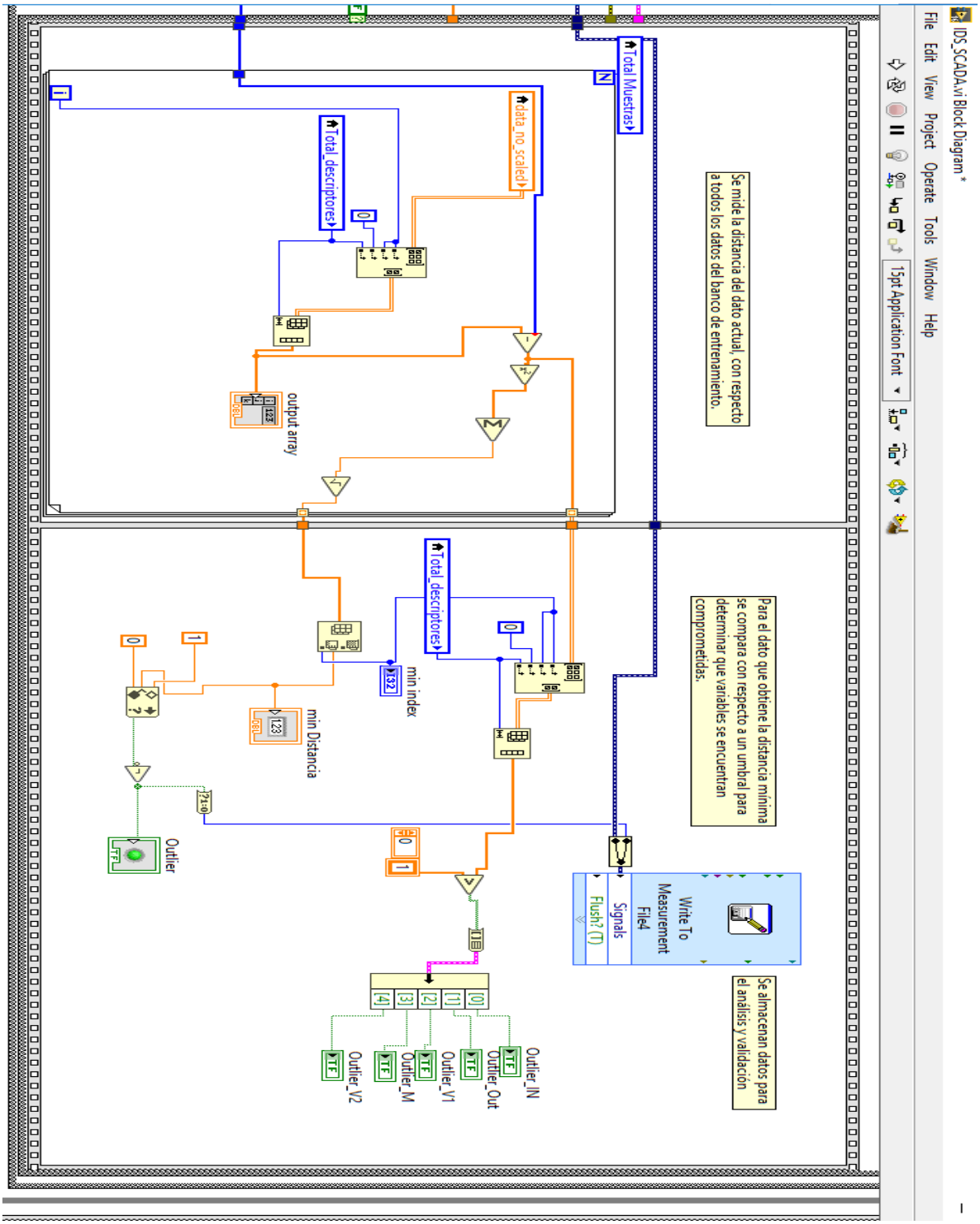
D. Anexo: Extracción de Datos para Visor 3D



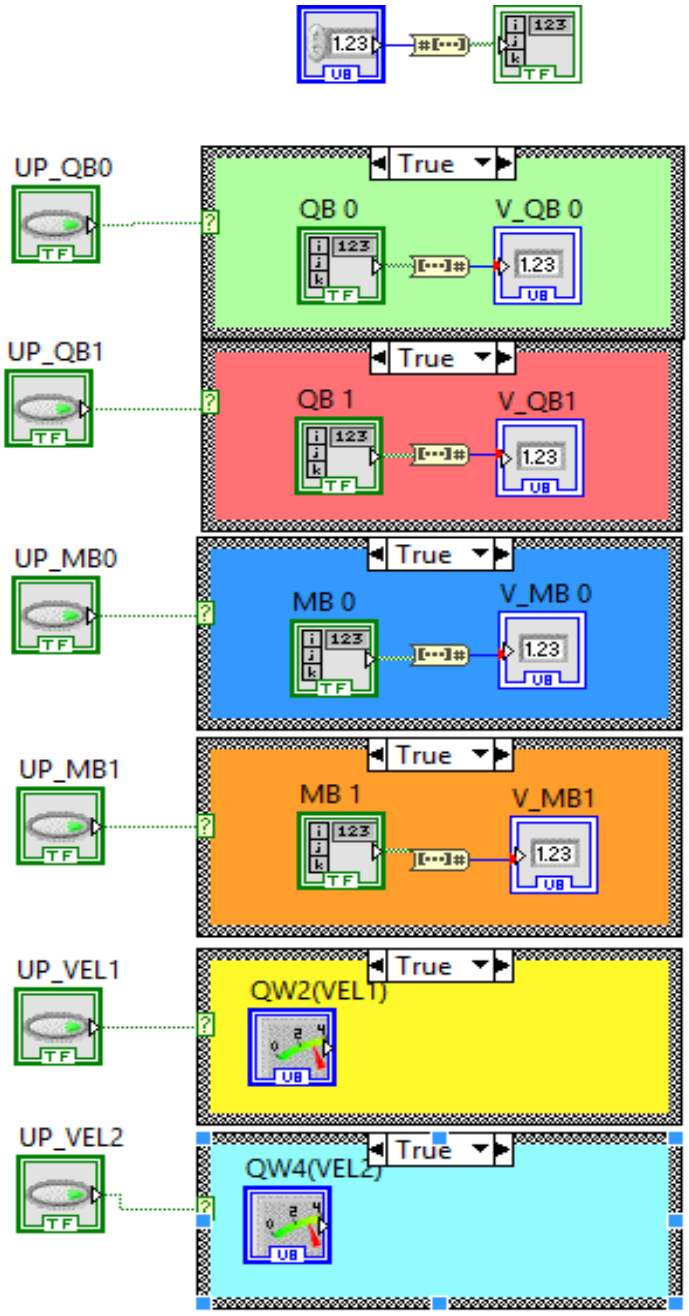
E. Anexo: Programación de Visor para vista de puntos en 3 dimensiones.



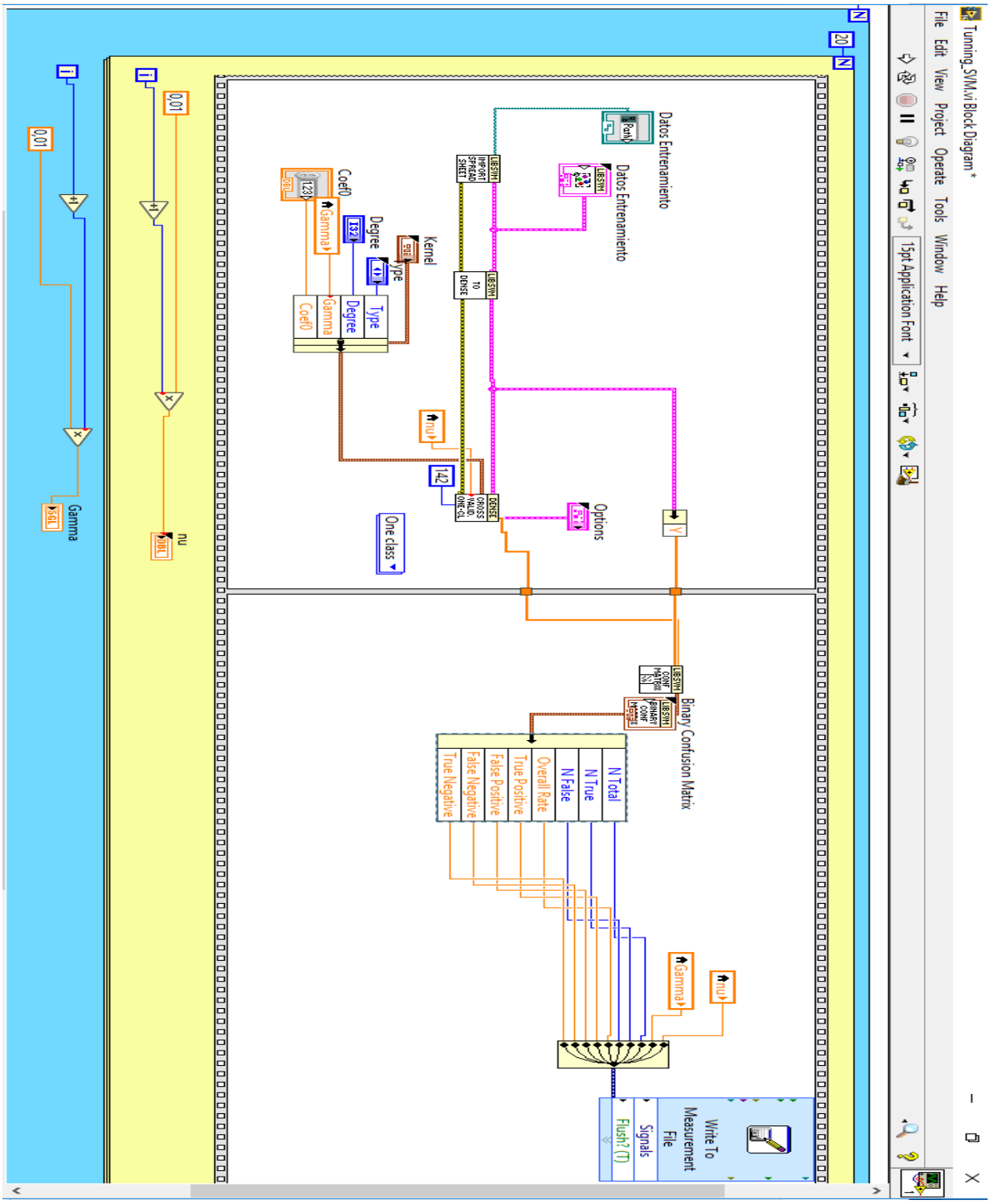
F. Anexo: Algoritmo de Distancia Crítica implementado en Labview



G. Anexo: Diagrama de Bloques Servidor Intruso



H. Anexo: Diagrama de Bloques Cálculo de Eficiencia de Entrenamiento



Bibliografía

[1] Al-Shaer, E., & Rahman, M. A. (2016). Security And Resiliency Analytics For Smart Grids: Static And Dynamic Approaches. Springer.

[2] Sommestad, T., Ericsson, G. N., & Nordlander, J. (2010). Scada System Cyber Security #X2014; A Comparison Of Standards. In 2010 Ieee Power And Energy Society General Meeting (Pp. 1–8). Http://Doi.Org/10.1109/Pes.2010.5590215

[3] Maglaras, L. A., & Jiang, J. (2014). Intrusion Detection In Scada Systems Using Machine Learning Techniques (Pp. 626–631). Ieee. Http://Doi.Org/10.1109/Sai.2014.6918252

[4] Melo, P. (2014, November 13). La Disponibilidad Como El Primer Eje De La Ciberseguridad Industrial. Retrieved May 8, 2017, From Http://www.Altadisponibilidadlogitek.Com/La-Disponibilidad-Como-El-Primero-De-Los-3-Ejes-Fundamentales-De-La-Ciberseguridad-Industrial/

[5]] Yang, Y., Pranggono, B., Littler, T., Yao, Z. Q., Eul Gyu Im, McLaughlin, K., … Sezer, S. (2012). Man-in-the-middle attack test-bed investigating cyber-security vulnerabilities in smart grid SCADA systems (pp. 138–138). Institution of Engineering and Technology. http://doi.org/10.1049/cp.2012.1831

[6] National Institute of Standards and Technology. Framework for Improving Critical Infrastructure Cybersecurity, 2014

[7] Lineamientos de política para ciberseguridad y ciberdefensa, Consejo Nacional de Política Económica y Social República de Colombia. Bogotá, 2011.

[8] Attacks Against SCADA Systems Doubled in 2014: Dell | SecurityWeek.Com. (n.d.). Retrieved May 8, 2017, from http://www.securityweek.com/attacks-against-scada-systems-doubled-2014-dell

[9] By. (2007, November 27). Detecting packet injection: a guide to observing packet spoofing by ISPs. Retrieved May 8, 2017, from https://www.eff.org/es/wp/detecting-packet-injection

[10] automatizacionindustrial. (2011, February 9). ¿ Que es la Automatización Industrial? Retrieved May 8, 2017, from https://automatizacionindustrial.wordpress.com /2011/02/09/queeslaautomatizacionindustrial/

[11] Conceptos Del Control Automático Industrial. (n.d.). Retrieved May 8, 2017, from http://www.sapiensman.com/control_automatico/

[12] Instrumentación Industrial - Antonio Creus.pdf. (n.d.). Retrieved May 8, 2017, from https://es.scribd.com/doc/205829081/Instrumentacion-Industrial-Antonio-Creus-pdf

[13] Software de Desarrollo de Sistemas NI LabVIEW - National Instruments. (n.d.). Retrieved May 8, 2017, from http://www.ni.com/labview/esa/

[14] SCI- Procesos de Control y Automatizacion. (n.d.). Retrieved May 8, 2017, from http://www.sistemasdecontrolindustrial.com/control%20y%20automatizacion.html

[15] Villajulca, J. C. (n.d.). Los Buses de Campo: directo al grano. Retrieved May 8, 2017, from http://www.instrumentacionycontrol.net/cursos-libres/automatizacion/curso-supervision-procesos-por-computadora/item/271-los-buses-de-campo-directo-al-grano.html

[16] History. (n.d.). Retrieved May 8, 2017, from https://opcfoundation.org/about/opc-foundation/history/

[17] Que es un Servidor OPC? (2005, January 1). [text]. Retrieved May 8, 2017, from http://matrikonopc.es/opc-servidor/index.aspx

[18] Unified Architecture. (n.d.). Retrieved May 9, 2017, from https://opcfoundation.org/about/opc-technologies/opc-ua/

[19] Sobre NI - National Instruments. (n.d.). Retrieved May 9, 2017, from http://www.ni.com/company/about-ni/esa/

[20] OPC. (n.d.). Retrieved May 9, 2017, from http://www.ni.com/opc/esa/

[21] Siemens España. LOGO! - El reconocido módulo lógico de Siemens, ahora también con Ethernet - El Futuro de la Industria - Siemens [WCMS3Article]. Retrieved May 15, 2017, fromhttp://w5.siemens.com/spain/web/es/industry/automatizacion/noticias/pages/logo!ahora2igualdegeniales.aspx

[22] tecnoplc. (2016, March 17). Comparación S7-200 S7-1200 Software y Hardware » tecnoplc. Retrieved May 15, 2017, from http://www.tecnoplc.com/comparacion-s7-200-s7-1200/

[23] Pierluigi Paganini. (2015, April 15). Dell report revealed attacks on SCADA system are doubled. Retrieved May 18, 2017, from http://securityaffairs.co/wordpress/35967/hacking/dell-attacks-on-scada-doubled.html

[24] Sayegh, N., Chehab, A., Elhajj, I. H., & Kayssi, A. (2013). Internal security attacks on SCADA systems (pp. 22–27). IEEE. https://doi.org/10.1109/ICCITechnology.2013.6579516

[25] Beresford, D. (2011). https://media.blackhat.com/bh-us-11/Beresford/BH_US11_Beresford_S7_PLCs_WP.pdf. Black Hat USA+2011 , (pág. 26).

[26] Larrieu, C.Sistemas de detección de instrusos (IDS).(2003). http://es.kioskea.net/contents/162-sistema-de-deteccion-de-intrusiones-ids.

[27] Carcano, A., Coletta, A., Guglielmi, M., Masera, M., Nai Fovino, I., & Trombetta, A. (2011). A Multidimensional Critical State Analysis for Detecting Intrusions in SCADA Systems. IEEE Transactions on Industrial Informatics, 7(2), 179–186. http://doi.org/10.1109/TII.2010.2099234

[28] Verba, J., & Milvich, M. (2008). Idaho National Laboratory Supervisory Control and Data Acquisition Intrusion Detection System (SCADA IDS) (pp. 469–473). IEEE. http://doi.org/10.1109/THS.2008.4534498

[29] Maglaras, L. A., & Jiang, J. (2014). OCSVM model combined with K-means recursive clustering for intrusion detection in SCADA systems (pp. 133–134). IEEE. http://doi.org/10.1109/QSHINE.2014.6928673

[30]Isaza Claudia V. Técnicas de agrupamiento en la supervisión de sistemas -conceptos y aplicaciones. Universidad de Antioquia. Medellín, Colombia. pag.33.

[31]Machine Learning. (s. f.). Recuperado 12 de mayo de 2016, a partir de http://www2.cs.uregina.ca/~dbd/cs831/notes/ml/1_ml.html.

[32] What is an intuitive explanation of overfitting? - Quora. (s. f.). Recuperado 12 de mayo de 2016, a partir de https://www.quora.com/What-is-an-intuitive-explanation-of-overfitting

[33] Chih-Chung Chang And Chih-Jen Lin, Libsvm : A Library For Support Vector Machines. Acm Transactions On Intelligent Systems And Technology, 2:27:1--27:27, 2011. Software Available At Http://Www.Csie.Ntu.Edu.Tw/~Cjlin/Libsvm

[34] Scikit-Learn: Machine Learning In Python, Pedregosa Et Al., Jmlr 12, Pp. 2825-2830, 2011.

[35] National Institute of Standards and Technology .Framework for Improving Critical Infrastructure Cybersecurity , 2014

[36] Paukatong, T. (2005). SCADA Security: A New Concerning Issue of an In-house EGAT-SCADA (pp. 1–5). IEEE. http://doi.org/10.1109/TDC.2005.1547116

[37] Common Cybersecurity Vulnerabilities in Industrial Control Systems. Control systems security program, National cybersecuriy division.Publication Year: 2011.Country: United States.

[38] Wang, T., Xiong, Q., Gao, H., Peng, Y., Dai, Z., & Yi, S. (2013). Design and Implementation of Fuzzing Technology for OPC Protocol (pp. 424–428). IEEE. http://doi.org/10.1109/IIH-MSP.2013.112

[39] Disso, J. P., Jones, K., & Bailey, S. (2013). A Plausible Solution to SCADA Security Honeypot Systems (pp. 443–448). IEEE. http://doi.org/10.1109/BWCCA.2013.77

[40] Sommestad, T., Ericsson, G. N., & Nordlander, J. (2010). SCADA system cyber security — A comparison of standards (pp. 1–8). IEEE. http://doi.org/10.1109/PES.2010.5590215

[41] Verba, J., & Milvich, M. (2008). Idaho National Laboratory Supervisory Control and Data Acquisition Intrusion Detection System (SCADA IDS) (pp. 469–473). IEEE. http://doi.org/10.1109/THS.2008.4534498

[42] Yang, Y., McLaughlin, K., Littler, T., Sezer, S., Pranggono, B., & Wang, H. F. (2013). Intrusion Detection System for IEC 60870-5-104 based SCADA networks (pp. 1–5). IEEE. http://doi.org/10.1109/PESMG.2013.6672100

[43] Carcano, A., Coletta, A., Guglielmi, M., Masera, M., Nai Fovino, I., & Trombetta, A. (2011). A Multidimensional Critical State Analysis for Detecting Intrusions in SCADA Systems. IEEE Transactions on Industrial Informatics, 7(2), 179–186. http://doi.org/10.1109/TII.2010.2099234

[44] Schuster, F., & Paul, A. (2012). A distributed intrusion detection system for industrial automation networks (pp. 1–4). IEEE. http://doi.org/10.1109/ETFA.2012.6489703

[45] Littler, T., Wang, H. F., Yang, Y., McLaughlin, K., & Sezer, S. (2013). Rule-based intrusion detection system for SCADA networks (pp. 1.05–1.05). Institution of Engineering and Technology. http://doi.org/10.1049/cp.2013.1729

[46] Kim, S.-J., Kim, B.-H., Yeo, S.-S., & Cho, D.-E. (2013). Network Anomaly Detection for M-Connected SCADA Networks (pp. 351–354). IEEE. http://doi.org/10.1109/BWCCA.2013.61

[47] Yang, Y., McLaughlin, K., Sezer, S., Littler, T., Im, E. G., Pranggono, B., & Wang, H. F. (2014). Multiattribute SCADA-Specific Intrusion Detection System for Power Networks. IEEE Transactions on Power Delivery, 29(3), 1092–1102. http://doi.org/10.1109/TPWRD.2014.2300099

[48] Maglaras, L. A., & Jiang, J. (2014). Intrusion detection in SCADA systems using machine learning techniques (pp. 626–631). IEEE. http://doi.org/10.1109/SAI.2014.6918252

[49] Maglaras, L. A., & Jiang, J. (2014). OCSVM model combined with K-means recursive clustering for intrusion detection in SCADA systems (pp. 133–134). IEEE. http://doi.org/10.1109/QSHINE.2014.6928673

[50] BBC, iWonder. (n.d.). El virus que tomó control de mil máquinas y les ordenó autodestruirse. Retrieved June 5, 2017, from http://www.bbc.com/mundo/noticias/2015/10/151007_iwonder_finde_tecnologia_virus_stuxnet

[51] Supported Device & Driver Plug-in List for NI OPC Servers - National Instruments. (n.d.). Retrieved June 5, 2017, from http://www.ni.com/white-paper/6417/en

[52] R, H. G., & Muñoz, Á. G. (2014). GENERACIÓN DE DIAGRAMAS LADDER MEDIANTE EL USO DE REDES DE PETRI DIFUSAS. Revista Vínculos, 10(2), 367–380.

[53] SIEMENS - SIMATIC S7-1200, CPU 1214C, CPU compacta, DC/DC/Relé. (n.d.). Retrieved June 7, 2017, from http://masvoltaje.com/simatic-s7-1200/1199-simatic-s7-1200-cpu-1214c-cpu-compacta-dc-dc-rele-6940408101319.html

[54] Siemens TIA Portal - El Futuro de la Industria - Siemens. (n.d.). [WCMS3Portfolio]. Retrieved June 7, 2017, from http://w5.siemens.com/spain/web/es/industry/automatizacion/simatic/tia-portal/pages/tiaportal.aspx

[55]One-class SVM with non-linear kernel (RBF) — scikit-learn 0.18.1 documentation. (n.d.). Retrieved June 10, 2017, from http://scikit-learn.org/stable/auto_examples/svm/plot_oneclass.html

USO EXCLUSIVO DEL PROGRAMA

Acta comité _____ **Fecha**_____

Favor tener en cuenta que los evaluadores propuestos no deben pertenecer a su mismo grupo de investigación, deben tener un título igual o superior al cual aspira el estudiante de posgrado y experiencia en investigación acreditada, vinculados a Universidades u organismos de enseñanza superior o investigación. No deben tener relación con la propuesta a evaluar ni publicaciones recientes con el estudiante, tutor o grupo de investigación acerca del tema de la propuesta, trabajo o tesis.

Para la selección de evaluadores se tendrá lo estipulado en el reglamento de posgrados de la Facultad de Ingeniería:

Para propuesta o trabajo de Maestría

- Serán 2 jurados con título de Magister o Doctor
- Por lo menos uno de los evaluadores debe ser externo a la Universidad de Antioquia
- Uno de ellos o ambos pueden ser internacionales

Para las propuestas de investigación de Doctorado

- Serán 2 jurados con título de Doctor
- Ambos jurados deben ser externos a la Universidad de Antioquia
- Por lo menos uno de los jurados debe ser internacional

Para las tesis Doctorales

- Serán 3 jurados con título de Doctor
- Por lo menos 2 de los jurados deben ser externos a la Universidad de Antioquia
- Por lo menos uno de los jurados debe ser internacional

Andrés Felipe Sánchez.
Nombre y Firma del estudiante
C.C.1017153.697

Nombre y Firma del Tutor
C.C. 71598507