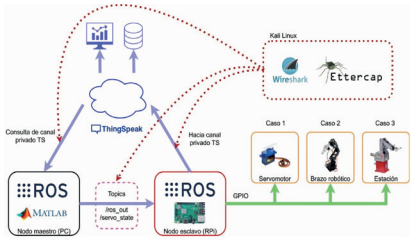
Ciberseguridad del internet de las cosas robóticas: plataforma experimental



Cybersecurity on internet of robotics things: experimental platform

Luis-Alberto Flores-Montaño[1], Juan-Carlos Herrera-Lozada[1], Jacobo Sandoval-Gutiérrez[2], Rodrigo Vázquez-López[1] y Daniel-Librado Martínez-Vázquez[2]

[1] Instituto Politécnico Nacional-CIDETEC. Av. Juan de Dios Bátiz, s/n. Nueva Industrial Vallejo - 07700 Ciudad de México (México).
[2] Universidad Autónoma Metropolitana Unidad Lerma. Av. de las Garzas #10, El panteón - 52005 Lerma de Villada (México).

DOI: https://doi.org/10.6036/10022 | Recibido: 01/dic/2020 • Inicio Evaluación: 09/dic/2020 • Aceptado: 04/may/2021

To cite this article: FLORES-MONTANO, Luis-Alberto; HERRERA-LOZADA, Juan-Carlos; SANDOVAL-GUTIÉRREZ, Jacobo; VÁZQUEZ-LÓPEZ, Rodrigo; MARTÍNEZ-VÁZQUEZ, Daniel-Librado. CYBERSECURITY ON INTERNET OF ROBOTICS THINGS: EXPERIMENTAL PLATFORM. *DYNA*. September 2021, vol. 96, no. 5, p. 540-545. DOI: https://doi.org/10.6036/10022

FINANCIACIÓN

El trabajo descrito en este artículo está financiado parcialmente por el Consejo Nacional de Ciencia y Tecnología (CONACYT) por medio de recursos recibidos gracias al Sistema Nacional de Investigadores (SNI) y el programa de Becas Nacionales. Adicionalmente, agradecemos al Instituto Politécnico Nacional por el financiamiento recibido por medio de los proyectos 20200332 y 20211103, a los proyectos de investigación Sistemas Ciberfísicos y Sistemas Autónomos de la Universidad Autónoma Metropolitana.

ABSTRACT

• The Internet of Robotic Things (IoRT) is a technology that looks for monitoring, operating, and maintaining the tasks of multiple robots through the cloud. However, using these robots in cyberspace has a risk and an inherent problem in cybersecurity. To analyze the implications of this technology, the objective was to design, operate and submit an IoRT system with the default configuration. The proposed methodology consisted of designing an IoRT architecture; implement three robotic platforms linked to the cloud, applying a sniffing and spoofing cyberattacks, assess the impacts, and propose solutions. The experiment used three prototypes: two servo motors, a 6-degree-of-freedom arm, and a workstation with a robot. Additionally, the tools of the experiment were a conventional computer, a Raspberry Pi microcomputer, the Robotic Operative System middleware, the Kali Linux distribution, and the ThingSpeak cloud service. The contributions of the work were three, first it was proven that four types of links are sufficient to homologate, and ensure the integrity, reliability, and availability in the operation of different types of robots. Also, it was possible the connection of these robots even though they are not designed to work on the internet through a slave-robot node link. Finally, a real list of the consequences was obtained, given the vulnerabilities and the attacks tested, as well as some recommendations.

Keywords: Cybersecurity, IoRT, Industry 4.0., Common Vulnerabilities and Exposures, Cloud, ROS.

RESUMEN

El Internet de las Cosas Robóticas (IoRT) es una tecnología que busca monitorear, operar y mantener las tareas de los múltiples robots a través de la nube. Sin embargo, al utilizar estos robots en el ciberespacio se tiene un riesgo y un problema inherente en la ciberseguridad. Para analizar dichas implicaciones de la tecnología, el objetivo fue diseñar, operar y someter un sistema IoRT con la configuración por defecto. La metodología propuesta consistió en diseñar una arquitectura IoRT; implementar tres plataformas robóticas vinculadas a la nube, realizar ataques de tipo esnifeo y suplantación, evaluar los impactos y proponer las soluciones. El experimento utilizó tres prototipos: dos servomotores, un brazo de 6 grados de libertad y una estación de trabajo con un robot. Adicionalmente, las herramientas del experimento fueron una computadora convencional, una microcomputadora Raspberry Pi, el middleware Robotic Operative System, la distribución Kali Linux y el servicio de nube ThingSpeak. Las aportaciones del trabajo fueron tres, primero se probó que cuatro tipos de enlace son suficientes para homologar, y procurar la integridad, confiabilidad y disponibilidad en la operación de distintos tipos de robots. También, se logró conectar robots aunque estos no estén diseñados para funcionar en internet mediante un enlace nodo esclavo-robot. Por último, se obtuvo un listado real de las consecuencias, dadas las vulnerabilidades y los ataques probados, así como algunas recomendaciones.

Palabras Clave: Ciberseguridad, Internet de las cosas robóticas, industria 4.0, Vulnerabilidades y exposiciones comunes, nube, ROS.

1. INTRODUCCIÓN

El ciberespacio comprende los sistemas electrónicos de hardware y software para realizar el almacenamiento, procesamiento y comunicación de la información. En estos sistemas se ha reportado un aumento de amenazas y vulnerabilidades cibernéticas a medida que los datos, los sistemas y los usuarios se conectan digitalmente [1]. Por ello, cuando se requiere conectar el ciberespacio a la Industria 4.0 [2] o al Internet de las cosas (IoT)[3] no hay confiabilidad plena.

1.1. CIBERSEGURIDAD

La ciberseguridad es la suma de las herramientas tecnológicas que evitan las vulnerabilidades y amenazas en el ciberespacio [4]. Cuando un atacante irrumpe la integridad, disponibilidad o confidencialidad del ciberespacio se considera una vulnerabilidad, mientras que una amenaza es la acción de aprovechar esa vulnerabilidad. Algunas de las clasificaciones de las vulnerabilidades son las interfaces y los servicios de red inseguros, configuración, autentificación y seguridad física insuficiente [5].

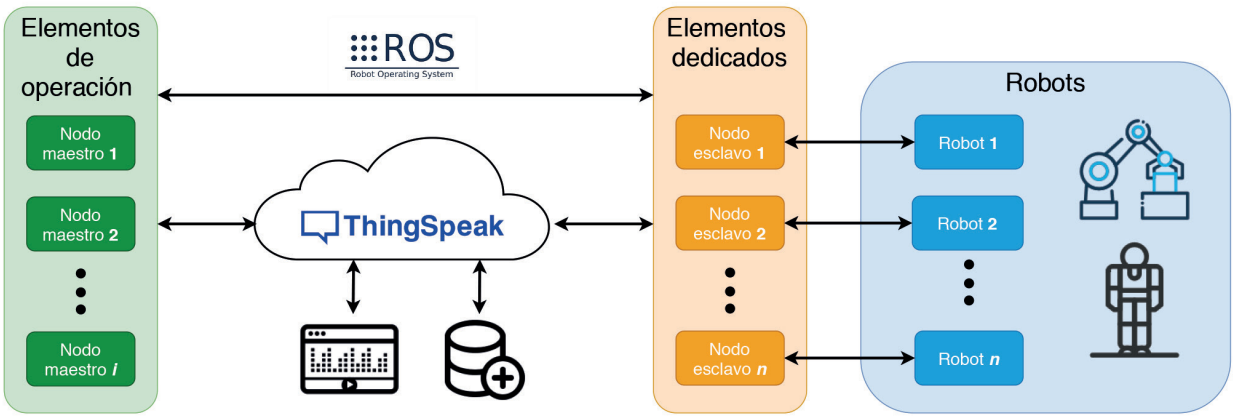


Figura 1. Propuesta de arquitectura del Internet de las Cosas Robóticas

En [6] se muestra que los ataques en el Internet Industrial de las Cosas (IIoT) no sólo tienen el propósito de irrumpir la información del sistema sino también a los dispositivos físicos que se encuentran en la arquitectura. Algunos ataques reportados para irrumpir la información son los ataques de canal lateral, suplantación de la identidad, ransomware, ataque de hombre en el medio, entre otros [6] [7] [8] [9]. Algunos de los ataques a los dispositivos físicos en la arquitectura son ataque de rastreo, denegación de servicio, barrido de puertos, ataques de tiempo lateral, de inyección a la medida, interferencia de Radio Frecuencia, entre otros [6] [7] [9] [10].

1.2. CIBERSEGURIDAD EN EL INTERNET DE LAS COSAS ROBÓTICAS IORT

La arquitectura del IoRT considera cinco capas: robot, red, internet, infraestructura, y de aplicación [11]. Las tres capas intermedias son compatibles con el IoT, por lo tanto, se infiere que parte de las implicaciones en la ciberseguridad (en red, de internet e infraestructura) del IoT aplican al IoRT. Sin embargo, considerado la capa inferior de los robots y la superior de las aplicaciones robóticas se crean nuevos retos en la investigación referente a la ciberseguridad.

Dos casos de estudio en la ciberseguridad de IIoT tienen similitud con la arquitectura del IoRT. En el primero se busca mostrar probables vulnerabilidades y amenazas comunes cuando se automatiza un servomotor inteligente [5]. En el segundo caso se refiere al análisis del impacto que tienen los ataques a una arquitectura compuesta de un sistema embebido con un servomotor [12].

Por lo tanto, consideramos que existen dos factores clave por experimentar en la ciberseguridad del IoRT. Primero definir una arquitectura que pueda pasar de los conceptos a un ámbito práctico y real. El segundo factor es proponer diferentes nodos y enlaces de comunicación, que consideren las probables vulnerabilidades y amenazas.

En consecuencia, el objetivo del trabajo fue proponer el diseño y operación de una arquitectura funcional del IoRT para identificar las potenciales vulnerabilidades y amenazas. Para verificar la propuesta se realizó una metodología que consta del diseño de un subsistema remoto de control maestro para la visualización y operación de los múltiples robots, un subsistema de nodos remotos esclavos los cuales operan un robot a la vez. Un sistema de mensajes como enlace para la operación directa o indirecta al robot mediante una tarjeta dedicada como interfaz de comunicación. Una plataforma en la nube para la visualización, almacenamiento y protección del tráfico de la información. Por último, la experimentación se realizó bajo la premisa de una configuración por defecto, en donde se utilizaron herramientas de software para realizar los ataques.

Las aportaciones del trabajo muestran un panorama de la ciberseguridad en el uso de múltiples robots conectados al internet. La plataforma IoRT diseñada y probada incluye los elementos básicos para realizar prototipos que pueden ser replicados utilizando dispositivos comerciales asequibles.

2. MATERIAL Y MÉTODOS

2.1. ARQUITECTURA IORT

La arquitectura del IoRT fue retomada con las ideas de [11] [13] [14] [15] [16] [17], mediante cuatro grupos de operación. El primer grupo se forma con los elementos de operación denominados nodos maestros (NM); estos son clientes remotos que funcionan desde una terminal de comandos Shell hasta un software de alto nivel como MATLAB. El segundo grupo se forma con un servicio de nube pública, privada o híbrida (TS); este servicio de nube permite el almacenamiento de datos, visualización u operación remota de los robots. El tercer grupo se forma con los elementos dedicados denominados nodos esclavos (NE); estos se encargan de

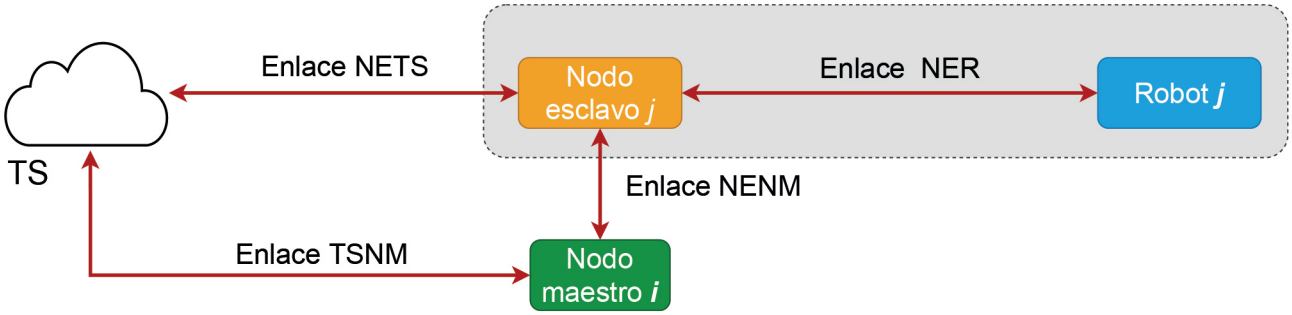


Figura 2. Diagrama de vulnerabilidades del envío de información física

la gestión e interpretación de las tareas robóticas. Por último, se encuentran las plataformas robóticas denominadas robots (R) que están vinculadas a cada NE. En la Figura 1 se muestra la propuesta de arquitectura.

2.2. VULNERABILIDAD Y AMENAZAS EN LA ARQUITECTURA IORT PROPUESTA

Para establecer la comunicación entre los nodos de los grupos (NM, TS, NE y R) de la Figura 1 se requieren cuatro tipos de enlaces TSNM, NETS, NMNE y NER. Estos enlaces permiten analizar las vulnerabilidades, amenazas e implicaciones posibles. En la Figura 2 se observan los enlaces.

En particular, el enlace NER tiene dos funciones no propuestas anteriormente en otras investigaciones, estas funciones son para disminuir las vulnerabilidades y habilitar la operación de cualquier robot en el IoRT. Los análisis correspondientes de estos enlaces se muestran en la Tabla 3.

2.3. FUNCIONAMIENTO DE LOS MENSAJES CON EL ENLACE NER

Para disminuir las vulnerabilidades del enlace NER se requiere implementar cierto tipo de mensajes especiales hacia el robot. Se considera utilizar dos tipos de mensajes en la arquitectura para la operación directa o indirecta. La carga útil que recibe el robot se extrae del mensaje completo que viaja por los demás enlaces y es definido en (1).

$$M = F + ID + PL + K \quad (1)$$

Donde:
M: Mensaje
F: Fecha (mes, día, año, hora, minuto y segundo)
ID: Identificador único de servicio en la nube
PL: Carga útil (Lo único que recibe el robot)
K: Llave digital

Para los valores de la carga útil en la operación directa se considera que los robots parten de un estado inicial. Independientemente del tipo de robot se debe caracterizar primero el espacio de trabajo antes de establecer la operación. Por ejemplo, para un brazo robótico de grados de libertad tendría los valores de torque, velocidad angular, aceleración o ángulos. Supongamos que se trabaja con los ángulos de cada grado de libertad como θ_i, por lo que estos deberán estar contenidos dentro de la carga útil como un arreglo $PL = [\theta_1, \theta_2, \ldots, \theta_n]$. En el caso de la operación indirecta, se tiene una carga útil con una tabla de verdad en formato de bits como arreglo $PL = [BIT_1, BIT_2, \ldots, BIT_n]$ tal que, el BIT_1 es el menos significativo y el BIT_n el más significativo.

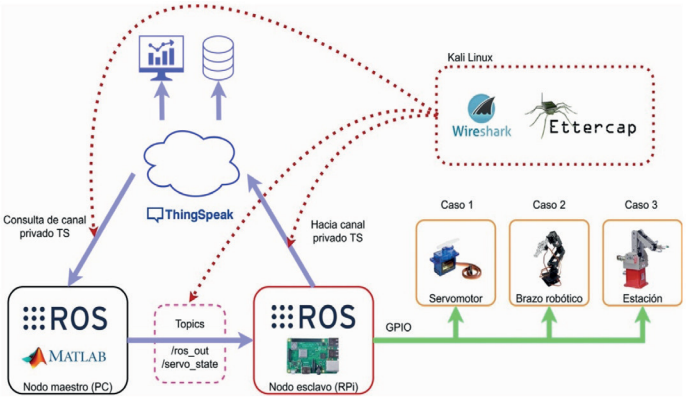


Figura 3. Dispositivos utilizados para la operación de las plataformas robóticas y para los respectivos ataques

2.4. EXPERIMENTOS

Para llevar a cabo los experimentos se utilizó la arquitectura propuesta con los dispositivos y el software siguiente:

- TS. Una plataforma de servicio en la nube IoT (ThingSpeak).
- NM. Un equipo de cómputo con MATLAB y el middleware Robotic Operating System (ROS)
- NE. Una microcomputadora Raspberry Pi con Raspberry Pi OS, ROS y Python.
- R. Tres plataformas robóticas.
  - La primera utilizando dos servomotores y la segunda un brazo robótico de 6 grados de libertad, ambos con operación directa.
  - La tercera es una celda de trabajo con banda transportadora y un robot industrial (Pegasus Control Software y Festo Pneumatic) en operación indirecta.

Para los enlaces y protocolos se empleo lo siguiente:

- TSNM y NETS. Un enlace de internet con un canal propio utilizando el API REST.
- NMNE. Una red local con el protocolo TCP/IP con tópicos privados de mensajes con el modelo publicar-suscribir.
- NER. Bus de comunicación de entradas y salidas de propósito general (GPIO).

Las consideraciones para los ataques por esnifeo y suplantación fueron las siguientes:

- Los protocolos de red ARP, DNS, TCP y UDP.
- La distribución de Kali Linux 2021 y las herramientas de prueba para la penetración fueron Ettercap y WireShark.

En la Figura 3 se puede observar el funcionamiento, las herramientas y la incidencia de los ataques en la arquitectura propuesta que lleva la configuración por defecto. Para los tres casos el proceso inicia en el NM enviando un mensaje al NE y queda a la espera de la confirmación del TS. El NE envía las instrucciones al R y este confirma que acato la instrucción. Por lo tanto, el NE envía la confirmación al TS. Este último, almacena el dato, grafica el valor y confirma al NM que ha terminado la instrucción y se encuentra listo para el siguiente mensaje.

2.4.1. Experimentos de funcionamiento.

2.4.1.1. Caso 1: Servomotores

Para este caso se utilizó el método de operación directa y sin configuración inicial, donde se conectaron dos servomotores al puerto GPIO de la Raspberry Pi como elemento dedicado (nodo esclavo).

El elemento de operación (nodo maestro), envía un mensaje cuya carga útil contiene el valor de la posición de los ángulos θ_1 y θ_2 equivalentes al servomotor 1 y 2. Una vez que el elemento dedicado recibe el mensaje, un código en Python genera las señales de modulación de ancho de pulso (PWM) de la posición deseada a la salida del puerto GPIO.

El experimento consistió en generar una secuencia compuesta de un conjunto de n valores θ_1, θ_2 y verificar el funcionamiento y correcto envío de las secuencias al servicio IoT (TS).

2.4.1.2. Caso 2: Brazo robótico

Para el segundo caso, se utilizó el método de operación directa y configuración inicial Denavit-Hartenberg para 6 grados de libertad, el cual se compone de servomotores. Se caracterizaron los rangos de operación de cada grado de libertad, los cuales

se observan en la Tabla 1. Es importante mencionar que dichos valores funcionan como indicadores de seguridad para evitar un daño en el dispositivo.

Para este caso se tiene una carga útil $PL = [\theta_1, \theta_2, \ldots, \theta_n]$ donde n = 6 que equivale al número de servomotores en el brazo robótico. El experimento consistió en generar una secuencia de movimientos que simulan alguna actividad.

2.4.1.3. Caso 3: Estación de trabajo

Para este caso se utilizó el método de operación indirecta y la configuración de diversas tareas. El robot realiza la secuencia de tomar la pieza 1 o 2 del alimentador de la banda transportadora en un ciclo infinito o variantes de estas. En la tabla 2 se muestran las posibles condiciones probadas para tres bits de carga útil.

Se utilizó el puerto GPIO de forma binaria para comunicar las entradas y salidas del robot. La salida solo consideró un bit de seguridad para comunicar la disponibilidad de realizar las tareas.

Servomotor	Rango de movimiento		Posición inicial °
	Mínimo °	Máximo °	
Base (θ_1)	0	180	90
Unión 1(θ_2)	0	180	0
Unión 2 (θ_3)	0	180	18
Unión 3 (θ_4)	9	135	90
Base gripper (tenaza) (θ_5)	0	180	90
Gripper (θ_6)	36	108	108

Tabla 1. Rangos de movimiento de los eslabones del brazo robótico. Fuente. Elaboración propia

2.4.2. Ataques realizados

Los ataques se realizaron en la infraestructura del IoRT, con el propósito de recabar información real sobre el funcionamiento de la plataforma ante estos eventos. Los pasos para la realización de las pruebas fueron:

- Obtener la tabla ARP de los dispositivos conectados a la red, con el propósito de identificarlos. Para ello, se utilizaron los comandos del sistema operativo.
- Una vez identificados los dispositivos, con las herramientas de Kali Linux se configuraron las herramientas para el esnifeo y la suplantación. Las pruebas particulares fueron el tráfico de análisis, suplantación de identidad y ataque de hombre en el medio.
- Posteriormente, se ejecutaron las pruebas de penetración con las herramientas Ettercap, WireShark en los nodos maestro y esclavo.

Finalmente se analizó e interpretó la información extraída del proceso de penetración.

Tarea del robot	Valor PL y de GPIO (4,6,7) de entrada
Paro total	[0, 0, 0]
Pieza 1 en ciclo infinito	[0, 0, 1]
Pieza 2 en ciclo infinito	[0, 1, 0]
Pieza 1 y después pieza 2 en ciclo infinito	[0, 1, 1]
Dos operaciones de pieza 1 por una pieza 2 en un ciclo infinito	[1, 0, 0]
Continúa ciclo	[1, 0, 1] hasta [1, 1, 1]

Tabla 2. Operaciones con base a la carga útil de una celda de trabajo. Fuente. Elaboración propia

3. RESULTADOS

3.1. RESULTADOS DE FUNCIONAMIENTO

La visualización de los movimientos de las plataformas robóticas a través de la nube de internet se muestra en la Figura 4. Los resultados de las gráficas fueron consistentes con las tareas ejecutadas físicamente, es decir, cada cambio se validó con la ejecución de las plataformas robóticas en sitio. En los ejes de las verticales se muestra el envío de la carga útil y el eje de las horizontales los ejes de tiempo.

Para la capa de robots los tres casos cumplieron con las tareas programadas. El período de actualización para comprobar los cambios fue desde los 18 segundos hasta 60 segundos, dependiendo la latencia de la red. Para el caso del servomotor, la respuesta es inmediata al utilizar una menor cantidad de información y al no tener una limitación en el espacio de trabajo. En el brazo con seis de grados de libertad, la respuesta fue mayor por el costo computacional debido a la dinámica de los mecanismos. En el caso de la estación de trabajo la respuesta depende de un tiempo asincrónico por emplear la operación indirecta.

3.2. RESULTADOS DE LOS ATAQUES

En la Figura 5a, se muestra la tabla de dispositivos con la asignación de direcciones físicas, IPs y los nombres de los fabricantes. En la Figura 5b se muestra el éxito de la suplantación realizada con la herramienta Ettercap para todos los nodos. Este resultado confirmó que la infraestructura utilizada no cuenta con un mecanismo de seguridad para evitar la suplantación. Sin embargo, de todos los datos interceptados no se encontró información confidencial de estos dispositivos. En la Figura 5c se muestran los diversos resultados obtenidos por la herramienta WireShark; dicha herramienta registró íntegramente la secuencia del tráfico de la información, los dispositivos, la interacción, el tiempo y el volcado de las tramas de información. En estos resultados, se pudo identificar campos de información legible y, por lo tanto, su privacidad fue vulnerada parcialmente.

Con los ataques de suplantación practicados a los enlaces TSNM y NETS con la herramienta Ettercap no se tuvo mayor riesgo, debido a que establecen una conexión segura. Sin embargo, con la herramienta WireShark, a pesar de la conexión segura se pudo descubrir la actividad del sistema IoRT. Es decir, obtener el nombre del servidor que es www.thingspeak.com y de esta manera, aplicando ingeniería inversa se intuyó que, el servidor necesita

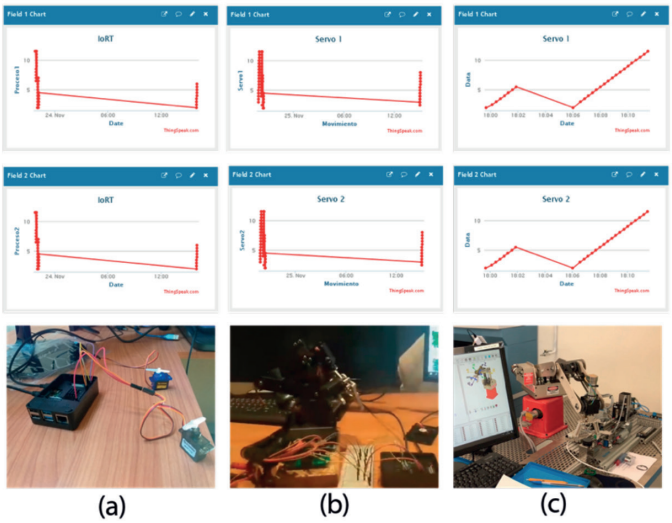


Figura 4. Gráficas y sus plataformas robóticas. (a) Servomotores, (b) Brazo robótico y (c) Estación de trabajo

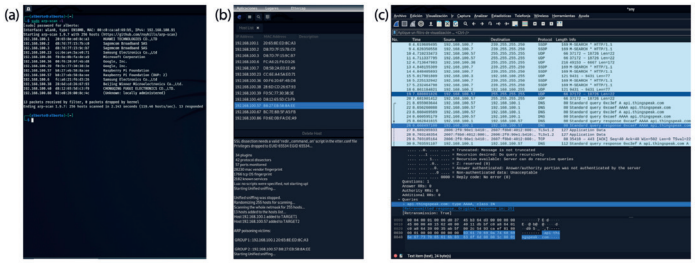


Figura 5. Resultados de las pruebas de ataque a la arquitectura

un número de identificación con 5 números y una llave privada de 16 caracteres. Pero al comparar la información para encontrar las claves, se observó que están cifradas con el algoritmo de encriptación SHA2.

En prueba inicial se experimentó una interrupción en el enlace NMNE debido al cortafuegos, es decir, el sistema catalogó esta conexión entre nodos como insegura. Adicionalmente, se intuyó una vulnerabilidad en el mensaje entre estos nodos, porque se maneja texto claro y por falta de un sistema de autentificación, sin embargo, las pruebas no permitieron corroborar esta hipótesis.

Para mayor detalle se anexa un enlace para visualizar los resultados de los experimentos. https://sway.office.com/KuRXXLXGnCkBO0iT?ref=Link

4. DISCUSIÓN Y CONCLUSIONES

4.1. DISCUSIÓN DE LA PLATAFORMA

Se alcanzó el objetivo propuesto al pasar de una arquitectura conceptual reportada en otras investigaciones a un diseño, construcción y funcionamiento real permitiendo que las plataformas robóticas se conecten al IoRT. Con el método propuesto de un nodo esclavo se habilita la función de comunicación con cualquier tipo de plataforma robótica sin importar que sea heterogénea o que esté inhabilitada para comunicarse a internet. En las tres plataformas robóticas utilizadas los sistemas consiguieron el valor agregado al pasar de una operación local a una operación remota y con monitoreo a través de la nube de internet.

Se validó el funcionamiento de un elemento dedicado capaz de operar dos tipos diferentes de sistemas robóticos. El NE para la primera operación establece una comunicación directa en el manejo de sus variables y en la segunda cuando el robot tiene un software y hardware de acceso limitado. El formato de mensajes diseñado y utilizado para la operación directa de robots es versátil para actuadores con diferentes configuraciones; sin embargo, la latencia de la red puede afectar el desempeño en la comunicación de mensajes por el tamaño de carga útil. En la operación indirecta el tamaño de la carga útil se reduce por utilizar bits para todas las operaciones que desempeña un robot. Para utilizar la operación

Enlace	Vulnerabilidad	Ataque potencial (Ataque probado) *	Comentarios
NMNE	Credenciales por defecto (Usuario y Contraseña) así como credenciales débiles o inseguras Ausencia de mecanismo de bloqueo Ausencia de opciones de encriptación o transmisión en texto claro No hay inicio de sesión o ausencia de factores de autenticación Puertos abiertos no deseados	**Suplantación de identidad*** **Ataque de tráfico de análisis*** Ataque físico Ataques de software Ataque de energía de agotamiento Ataque de lógica falsa Ataque de hombre en el medio	Se hicieron pruebas para el análisis de los datos que se transmiten a través de la red; sin embargo, no existió una evidencia de fuga de datos sensibles usando las herramientas Ettercap y Wireshark.
NETS	Credenciales por defecto Vulnerabilidades XSS Ausencia de mecanismo de bloqueo Vulnerabilidad de ataque DDoS Puertos abiertos no deseados Ausencia de opciones de encriptación o envío en texto claro Mecanismo de contraseña insegura o débil Ausencia de factores de autenticación Acceso al sistema operativo vía remota No es posible limitar las capacidades administrativas	**Ataque de tráfico de análisis*** **Suplantación de identidad*** **Ataque de hombre en el medio*** Ataque de lógica falsa Ataque de denegación de servicio (DDoS) Ataque de interferencia Ataque de sumidero Ataque de información de enrutamiento falsa	Se hicieron pruebas para el análisis de los datos que se transmiten a través de la red; sin embargo, no existió una evidencia de fuga de datos sensibles usando las herramientas Ettercap y Wireshark. Cabe mencionar que en la transmisión de información hacia el servicio (ThingSpeak) no se intentó ningún tipo de ataque, esto debido a las políticas de privacidad y legales del servicio.
TSNM	Vulnerabilidades XSS Vulnerabilidad de ataque DDoS Puertos abiertos no deseados Ausencia de factores de autentificación No es posible limitar las capacidades administrativas	**Ataque de información en tránsito*** **Ataque de tráfico de Análisis*** **Ataque de hombre en el medio*** Ataque de lógica falsa Ataque de interferencia Ataque de sumidero Ataque de información de enrutamiento falsa Ransomware Troyano/ Puerta trasera	Se hicieron pruebas para el análisis de los datos que se transmiten a través de la red; sin embargo, no existió una evidencia de fuga de datos sensibles usando las herramientas Ettercap y Wireshark. Cabe mencionar que en la transmisión de información hacia el servicio (ThingSpeak) no se intentó ningún tipo de ataque, esto debido a las políticas de privacidad y legales del servicio.
NER	Puertos externos no inhabilitados	**Ataque físico*** Inyección de código malicioso	Para este enlace se utilizó un ataque físico mediante desconexión de red y de energía eléctrica. En el caso de red se detuvo la operación y en el caso de energía se restableció el sistema.

Tabla 3. Listado de vulnerabilidad, ataques potenciales y probados por tipo de enlace.
Fuente. Elaboración propia con base en experimentos propios y [5] [6] [7] [8] [9] [10]

indirecta se debe conocer de manera especializada al robot para su intervención.

4.2. DISCUSIÓN DE LOS ATAQUES

Entre las diferentes herramientas utilizadas para los ataques, WireShark fue la más sofisticada y completa. Con el ataque de tráfico de análisis se obtuvo la mayor cantidad de información que fluye por la red interna. En la Tabla 3 se muestran las vulnerabilidades, así como los ataques potenciales y probados por tipo de enlace para mayores detalles.

4.3. CONCLUSIONES

Debido al éxito logrado con la plataforma del IoRT propuesta, se puede hacer las siguientes recomendaciones. Para lograr la compatibilidad de las plataformas robóticas sobre el IoRT, estas deberán considerar que los tiempos de ejecución, la infraestructura del internet, los niveles de seguridad y el tipo de permiso a los usuarios externos. Enfocándose a la Industria 4.0 el IoRT podría requerir de dispositivos exclusivos para este sector muy especializado y que posiblemente no sean compatibles con el IoT o el IIoT. Para reproducir la arquitectura IoRT propuesta se pueden sustituir los elementos de operación y dedicados con los diversos dispositivos similares en el mercado, así mismo, usar otro tipo de plataforma en la nube.

Derivado de los ataques probados en el ámbito de la ciberseguridad se pueden sugerir algunas recomendaciones para evitar vulnerabilidad y riesgos, tales como, la elaboración de normas, donde se deben incluir las definiciones particulares del IoRT y así, evitar vacíos normativos, por ejemplo, las que publica el Instituto Nacional de Estándares y Tecnología (NIST). Las amenazas y los ataques físicos implícitos en el IoRT, pueden ser los desastres naturales, incendios accidentales o provocados, tormentas, inundaciones, disturbios, sabotaje externo e interno deliberado.

Las vulnerabilidades pueden existir por los errores en los sistemas operativos, de configuración, en las aplicaciones de seguridad, y en las acciones u omisiones por parte de los usuarios no preparados para esta tecnología. De esta manera, también habría estrategias de prevención conocidas y aplicables como la gestión unificada de amenazas, prevención y detección de intrusiones IPS e IDS, cifrado de las comunicaciones, filtro de contenidos, herramientas de control P2P, gestión y control de ancho de banda, herramientas de monitorización y seguridad en la web.

Las VPN (Virtual Private Network) son algunas soluciones conocidas para resolver vulnerabilidades, sin embargo, su alto costo computacional no podría ser la mejor opción para el IoRT debido a la latencia en las comunicaciones. Otras posibles soluciones serían el encriptado de los datos o con la verificación de usuario a dos pasos. Para lo anterior, se tendría que invertir en recursos humanos, infraestructura y el incremento de la capacidad de recursos computacionales. La propuesta de los nodos esclavos y su enlace con el robot permite reducir vulnerabilidades y amenazas hacia el robot. Principalmente, se prevé que a pesar de recibir ataques exitosos en los otros enlaces diferentes al NER estos no impactarán en la operación del robot.

Para trabajos futuros se propone utilizar esquemas de prueba con ataques más severos, herramientas especializadas e ingeniería inversa para la evaluación de la seguridad.

REFERENCIAS

[1] PRIYADARSHINI, Ishaani. Cyber security risks in Robotics. In: Detecting and Mitigating Robotic Cyber Security Risks [online]. B.m.: IGI Global, 2017, p. 333–348. ISBN 9781522521556. Available at: doi: https://doi.org/ https://doi.org/10.4018/978-1-5225-2154-9.ch022

[2] CULOT, G, F FATTORI, M PODRECCA and M SARTOR. Addressing Industry 4.0 Cybersecurity Challenges. IEEE Engineering Management Review [online]. 2019, 47(3), 79–86. ISSN 1937-4178. Available at: doi: https://doi.org/https://doi.org/10.1109/EMR.2019.2927559

[3] CHOO, Kim-Kwang Raymond, Keke GAI, Luca CHIARAVIGLIO and Qing YANG. A multidisciplinary approach to Internet of Things (IoT) cybersecurity and risk management. Computers & Security [online]. 2021, 102, 102136. ISSN 0167-4048. Available at: doi: https://doi.org/https://doi.org/10.1016/j.cose.2020.102136

[4] PARK, S, S KIM, D D CHO, S JANG, J SUNG and S KIM. Multi-Robot Path Finding Testbed in Wireless Networks against Malicious Attacks. In: 2018 18th International Conference on Control, Automation and Systems (ICCAS). 2018, p. 120–123.

[5] ESWARAN, K., M.S.S. KUMAR, D. THANGAVELUSAMY and V. MURUGADOSS. Smart Servomotor for Robotics and its Cyber security. In: Proceedings - 2020 Advanced Computing and Communication Technologies for High Performance Applications, ACCTHPA 2020 [online]. 2020, p. 231–236. ISBN 9781728164533. Available at: doi: https://doi.org/10.1109/ACCTHPA49271.2020.9213226

[6] MOURTZIS, D., K. ANGELOPOULOS and V. ZOGOPOULOS. Mapping vulnerabilities in the industrial internet of things landscape. In: Procedia CIRP [online]. 2019, p. 265–270. Available at: doi: https://doi.org/10.1016/j.procir.2019.04.201

[7] ANDREA, I, C CHRYSOSTOMOU and G HADJICHRISTOFI. Internet of Things: Security vulnerabilities and challenges. In: 2015 IEEE Symposium on Computers and Communication (ISCC) [online]. 2015, p. 180–187. Available at: doi: https://doi.org/10.1109/ISCC.2015.7405513

[8] BASHEER, M.M. and A. VAROL. An Overview of Robot Operating System Forensics. In: 1st International Informatics and Software Engineering Conference: Innovative Technologies for Digital Transformation, IISEC 2019 - Proceedings [online]. 2019. ISBN 9781728139920. Available at: doi: https://doi.org/10.1109/UBMYK48245.2019.8965649

[9] VARGA, P., S. PLOSZ, G. SOOS and C. HEGEDUS. Security threats and issues in automation IoT. In: IEEE International Workshop on Factory Communication Systems - Proceedings, WFCS [online]. 2017. ISBN 9781509057887. Available at: doi: https://doi.org/10.1109/WFCS.2017.7991968

[10] JAMAI, I., L. BEN AZZOUZ and L.A. SAIDANE. Security issues in Industry 4.0. In: 2020 International Wireless Communications and Mobile Computing, IWCMC 2020 [online]. 2020, p. 481–488. ISBN 9781728131290. Available at: doi: https://doi.org/10.1109/IWCMC48107.2020.9148447

[11] RAY, P P. Internet of Robotic Things: Concept, Technologies, and Challenges. IEEE Access [online]. 2016, 4, 9489–9500. ISSN 2169-3536. Available at: doi: https://doi.org/10.1109/ACCESS.2017.2647747

[12] JIANG, X., M. LORA and S. CHATTOPADHYAY. An Experimental Analysis of Security Vulnerabilities in Industrial IoT Devices. ACM Transactions on Internet Technology [online]. 2020, 20(2). Available at: doi: https://doi.org/10.1145/3379542

[13] AFANASYEV, Ilya, Manuel MAZZARA, Subham CHAKRABORTY, Nikita ZHUCHKOV, Aizhan MAKSATBEK, Aydin YESILDIREK, Mohamad KASSAB and Salvatore DISTEFANO. Towards the Internet of Robotic Things: Analysis, Architecture, Components and Challenges. In: 2019 12th International Conference on Developments in eSystems Engineering (DeSE) [online]. B.m.: IEEE, 2019, p. 3–8. ISBN 978-1-7281-3021-7. Available at: doi: https://doi.org/10.1109/DeSE.2019.00011

[14] FOURNARIS, Apostolos P., Christos ALEXAKOS, Christos ANAGNOSTOPOULOS, Christos KOULAMAS and Athanasios KALOGERAS. Introducing Hardware-Based Intelligence and Reconfigurability on Industrial IoT Edge Nodes. IEEE Design & Test [online]. 2019, 36(4), 15–23. ISSN 2168-2356. Available at: doi: https://doi.org/10.1109/MDAT.2019.2908547

[15] SWATHI, K, T Uday SANDEEP and A Roja RAMANI. Performance Analysis of Microcontrollers Used In Iot Technology. International Journal of Scientific Research in Science, Engineering and Technology [online]. 2018, 4(4), 1268–1273. Available at: http://ijsrset.com/ IJSRSET1844270

[16] LI, Xing-Qian, Xin DING, Yue ZHANG, Zhi-Ping SUN and Hong-Wei ZHAO. IoT Family Robot Based on Raspberry Pi. In: 2016 International Conference on Information System and Artificial Intelligence (ISAI) [online]. B.m.: IEEE, 2016, p. 622–625. ISBN 978-1-5090-1585-6. Available at: doi: https://doi.org/10.1109/ISAI.2016.0137

[17] FU, Shuangquan and Pritesh Chandrashekhar BHAVSAR. Robotic Arm Control Based on Internet of Things. In: 2019 IEEE Long Island Systems, Applications and Technology Conference (LISAT) [online]. B.m.: IEEE, 2019, p. 1–6. ISBN 978-1-7281-2100-0. Available at: doi: https://doi.org/10.1109/LISAT.2019.8817333