*Cuadernos de Ingeniería. Nueva Serie.*Publicaciones Académicas
Fac. Ingeniería, Universidad Católica de Salta, vol. 9, 2016: 121-132

Seguridad de las Redes de Control Industrial – MODBUS/TCP con inspección profunda de paquetes[1]

Carlos Said[2]

Resumen

El artículo propuesto tiene como objetivo tratar la aplicación y la gestión de las prácticas de seguridad cibernética asociadas con la tecnología de la información y la tecnología operativa en relación con los entornos de redes de producción industrial (SCADA-ICS).
Se ahonda en los aspectos de las tecnologías de inspección profunda de paquetes de información y granularidad en el análisis de protocolos de datos, también llamados DPI.

Palabras clave: protocolos – SCADA - ICS, granularidad - DPI

Abstract

The proposed article aims to address the implementation and management of cyber security practices associated with information technology and operational technology regarding network environments of industrial production (SCADA-ICS).
The guidance in this article is to complement comprehensive cybersecurity specific areas related to deep packet inspection or granularity analysis of transmission protocols.

Keywords: protocols, SCADA, ICS, granularity, DPI

Introducción

Puede observarse que el aumento del riesgo de la seguridad cibernética en redes SCADA (Supervisory Control and Data Acquisition, en español Redes de Supervisión, Control y Adquisición de Datos) / ICS (Industrial Control Systems, en español Sistemas de Control Industrial) lleva a los proveedores de tecnologías de seguridad, a desarrollar mecanismos avanzados que se ocupen de los desafíos específicos de seguridad de los sistemas de control.

Una manifestación tangible de ello, se observa en el gráfico 1. Este identifica vulnerabilidades de seguridad explotadas en redes de control industrial.

---

[1] Este artículo fue presentado en: 2nd PAN-American Interdisciplinary Conference, PIC 2016, 24-26 February 2016, Buenos Aires, Argentina

[2] Facultad de Ingeniería - Universidad Católica de Salta.

ID	Disc Date	Title
117992	2015-01-30	SCADA SpecView Unspecified Remote Information Disclosure
		SCADA SpecView contains an unspecified flaw that may allow a remote attacker to gain access to potentially sensitive information. No further details have been provided.
119930	2015-03-13	Events SCADA HMI Unspecified Information Disclosure
		Events SCADA HMI contains an unspecified flaw that may allow an attacker to gain access to potentially sensitive information. No further details have been provided by the researcher.
119432	2015-03-10	SCADA Engine BACnet OPC Server BACnOPCSever.exe Format String
		SCADA Engine BACnet OPC Server contains a format string flaw in BACnOPCSever.exe. The issue is triggered as format string specifiers (e.g. %s and %x) are not properly sanitized in user-supplied input when handling a specially crafted request. This may allow a remote attacker to cause a denial of service or potentially execute arbitrary code.
119406	2015-03-10	SCADA Engine BACnet OPC Server Remote Item Manipulation
		SCADA Engine BACnet OPC Server contains a flaw related to authentication that may allow a remote attacker to insert, read, or delete any items in the database. No further details were provided.
120994	2015-04-17	Modbus SCADA (WLC Systems) Unspecified Traversal Remote Issue
		Modbus SCADA (WLC Systems) contains a flaw that allows traversing outside of a restricted path. The issue is due to the program not properly sanitizing user input, specifically path traversal style attacks (e.g. '../'). With a specially crafted request, a remote attacker can have an unspecified impact.
117990	2015-01-30	ScadaBR Unspecified File Upload Remote Command Execution
		ScadaBR contains an unspecified flaw that allows a remote attacker to execute arbitrary commands. This flaw exists because the program does not properly verify or sanitize user-uploaded files. By uploading a crafted file, the remote system will place the file in a user-accessible path. Making a direct request to the uploaded file will allow the attacker to execute commands with the privileges of the web server.
117102	2015-01-13	Clorius Controls ISC SCADA Java Web Client Insecure Credential Encryption MitM Credential Disclosure
		Clorius Controls ISC SCADA contains a flaw in the Java web client. The issue is triggered when credentials are not sufficiently encrypted, which may disclose credentials to a man-in-the-middle attacker.
119431	2015-03-10	SCADA Engine BACnet OPC Server Packet Handling Remote Heap Buffer Overflow
		SCADA Engine BACnet OPC Server contains an overflow condition that is triggered as user-supplied input is not properly validated when handling a specially crafted packet. This may allow a remote attacker to cause a heap-based buffer overflow, resulting in a denial of service or potentially allowing the execution of arbitrary code.

Gráfico 1. Incidentes de seguridad en redes SCADA-ICS

Fuente: http://blog.osvdb.org/category/vulnerability-statistics/

Es prudente iniciar el trabajo haciendo mención a que SCADA hace referencia a un sistema que opera sobre canales de comunicación para brindar control de equipos remotos.

El sistema de control puede combinarse con un sistema de captura/adquisición de datos para que a través del canal de comunicación se conozca el estado del equipo remoto, con fines de monitoreo, registro y/o control. En este caso suele llamarse sistema de control industrial (ICS). Los procesos industriales que se desean controlar son procesos que existen en el mundo físico. Los sistemas SCADA suelen distinguirse respecto de otros sistemas ICS por el hecho de controlar procesos de gran escala, que incluyen dispersión geográfica (número de sitos) y distancias relevantes.

Un aspecto de estos desafíos es el uso generalizado de protocolos de comunicación para sistemas ICS que no fueron diseñados pensando en la seguridad. Asegurar estos protocolos sin afectar su funcionalidad de control requiere de tecnología avanzada.

¿Cuáles son estos protocolos? MODBUS/TCP, BACnet IP, DNP3, solo para citar algunos de ellos.

Sin ahondar en este artículo, en todos estos protocolos (nuestro foco estará en MODBUS), pero debemos al menos mencionar en qué consisten los dos restantes que hemos mencionado).

BACnet es un protocolo de comunicación de datos para la construcción de redes de automatización y control.Fue desarrollado por motivación de de la Sociedad Americana de Ingenieros de Calefacción, Refrigeración y Aire Acondicionado (ASHRAE).

BACnet es un estándar nacional norteamericano, una norma europea, una norma nacional en más de 30 países, y un estándar mundial ISO. El protocolo es mantenido por el Comité Permanente nr^{o}. 135 del Proyecto para el Standard ASHRAE Standard.

DNP3 (Distributed Network Protocol, en español Protocolo de Red Distribuido) es un conjunto de protocolos de comunicación utilizados entre los componentes de los sistemas de automatización de procesos. Su uso principal es en empresas de electricidad y agua. Fue desarrollado para las comunicaciones entre los distintos tipos de equipos de adquisición y control de datos. En los sistemas SCADA, se utiliza en las estaciones maestras (master o centros de control), unidades terminales remotas (en inglés RTU, Remote Terminal Units), y dispositivos electrónicos inteligentes (en inglés,IED, Intelligent Electronica Devices). Se utiliza esencialmente para las comunicaciones entre una estación maestra y las RTUs o IEDs.

ICCP, (en inglés inter-control center communications protocol), es una parte de la norma IEC 60870-6, el cual se utiliza para las comunicaciones entre las estaciones maestras.

Retornemos a los aspectos de seguridad. Un ejemplo de estos mecanismos avanzados de seguridad de la información que 'fluye' usando estos protocolos como transporte es la inspección profunda de paquetes (en inglés, DPI o Deep Packet Inspection).

Por un lado, los sistemas de detección de intrusiones (IDS, Intrusion Detection Systems) brindan prestaciones de monitoreo para categorías genéricas de ataques básicos. Por otro lado, los firewalls utilizan básicamente listas de control de acceso, que permiten o bloquean todos los mensajes de un protocolo industrial como Modbus TCP.

Protocolo MODBUS

MODBUS es un protocolo de comunicación serial publicado originalmente por Modicon (ahora Schneider Electric) para su uso con sus controladores lógicos programables (en inglés PLC, programable logical controllers). Es simple y por ello se convirtió desde entonces en un protocolo de comunicación estándar de facto, disponible para la conexión de dispositivos electrónicos industriales. Fue desarrollado con aplicaciones industriales en mente, es público y gratuito, y esencialmente 'transporta bits' sin imponer mayores restricciones.

Modbus permite la comunicación entre múltiples dispositivos conectados a la misma red: por ejemplo un sistema que mide la temperatura y la humedad y comunica los resultados a un computador. Modbus se utiliza a menudo para conectar un computador supervisor con una terminal remota (RTU) en redes SCADA. Los tipos de datos suelen nombrarse con base en el uso en los relés de conducción: una salida física de un solo bit se llama bobina (o en inglés, Coil), y una entrada física de un solo bit se llama 'entrada discreta' o 'contacto'.

La actualización de los protocolos Modbus está a cargo de la Organización Modbus. Es una asociación de usuarios y proveedores de dispositivos compatibles con Modbus que busca impulsar la evolución del protocolo.

Solemos distinguir entre Modbus RTU sobre TCP (en inglés transmission control protocol) y de Modbus TCP.

Modbus RTU sobre TCP es en síntesis, un mensaje Modbus RTU transmitido con una envoltura (o empaquetado) TCP / IP y se envía a través de una red en lugar de líneas seriales. El servidor no tiene 'SlaveID' (identificación segura) ya que utiliza en lugar de ello una dirección IP.

Respecto a Modbus TCP, la guía de implementación de mensajes Modbus proporcionada por Schneider Automation define un protocolo modificado específicamente para su uso a través de TCP/IP. La especificación oficial Modbus se puede encontrar en www.modbus.org/specs.php. Las principales diferencias entre Modbus RTU y Modbus TCP escapan al alcance de este artículo, siendo eventualmente el objeto de una próxima redacción.

La estructura de un mensaje Modbus IP está dividido en dos partes. La primera de ellas es llamada Encabezamiento del Protocolo de Aplicación Modbus (o en Inglés, Modbus Application Protocol (MBAP) Header), que consiste de:

1. Identificación de la transacción (2 bytes) – Usado para sincronización entre mensajes cliente-servidor.
2. Identificación del protocolo (2 bytes) – Se encuentra en 0 para Modbus, y se reserva para usos futuros.
3. Longitud (2 bytes) – Usado para definir la cantidad de bytes restantes en la estructura del mensaje.
4. Identificador de la unidad (1 byte) – Dirección de la 'unidad esclava' a la cual le está siendo enviado el mensaje. Para Modbus, la dirección de del dispositivo / unidad 'esclavo' es la dirección IP, y por ello el identificador de unidad es puesto al valor hexadecimal 0xFF.

La segunda parte del mensaje es típicamente un mensaje de datos del protocolo Modbus (en Inglés, Modbus Protocol Data Unit (PDU)). Consiste de:

1. Código de Función (1 Byte) – Funciones soportadas por Modbus.
2. Datos de la función (n bytes) – Datos que acompañan el código de función, tales como respuestas o comandos.

Es esencialmente en esta segunda parte del mensaje donde las herramientas y técnicas de inspección profunda de paquetes, realiza su tarea de análisis.

Encabezamiento del Protocolo de Aplicación Modbus				Mensaje de datos del protocolo Modbus	
Identificación de la transacción	Identificación del protocolo	Longitud	Identificador de la unidad	Código de Función	Datos de la función

Gráfico 2. Estructura de un mensaje Modbus IP.

La granularidad en el análisis de los mensajes

La inspección profunda de paquetes, ahonda en el análisis de los datos, para intentar entender para qué se utiliza el protocolo, y como consecuencia de ello proporcionar protección, no sólo detección. Para hacer esto debe tener la capacidad de 'mirar' y analizar los campos del mensaje de datos del protocolo Modbus (ver gráfico 2).

Para realizar lo mencionado en el párrafo previo, DPI puede por ejemplo, mediante la determinación de si un mensaje Modbus es de lectura o de escritura, descartar todos los mensajes de escritura; o sólo permitir la escritura de determinados registros. Esto da lugar a que la protección se adapte a las necesidades de la aplicación, permitiendo que los mensajes de control esenciales fluyan según sea necesario, mientras que se bloquean los mensajes potencialmente peligrosos o inapropiados.

Para entender DPI, primero es importante entender cómo funciona un firewall tradicional. Un firewall es un dispositivo que monitorea y controla el tráfico que fluye en una red o entre redes. Captura el tráfico que pasa a través de él y lo compara con un conjunto predefinido de reglas (llamado Access Control Lists o ACL, Lista de Control de Accesos). Cualquier mensaje que no coincide con las ACL se descarta.

El firewall clásico permite que las ACL puedan comprobar tres campos principales en un mensaje, a saber:

1. La dirección IP de la computadora que envía el mensaje (IP de origen),
2. La dirección IP del equipo que recibe el mensaje (IP de destino),
3. El protocolo de la capa superior que figura en el paquete IP, como se define en el campo «Número de puerto TCP de destino».

El tercer punto, número de puerto de destino TCP puede necesitar un poco más de explicación. Estos puertos no son puertos físicos, como un puerto de Ethernet; son números especiales que forman parte de cada mensaje TCP o UDP para identificar el protocolo de la capa de aplicación que se transporta en el mensaje.

Por ejemplo, Modbus/TCP utiliza el puerto 502. Estos números están registrados bajo la autoridad regulatoria Internet Assigned Numbers Authority (IANA).

Para poner todo lo dicho en conjunto, imagine que sólo desea permitir que el tráfico de Internet (el tráfico HTTP, en inglés Hyper Text Transfer Protocol) de un cliente en la dirección IP 192.168.1.20, hacia un servidor web con una dirección 192.168.1.51. Entonces escribiría una regla ACL similar a la siguiente

Permitir Origen=192.168.1.20 Destino=192.168.1.51 Puerto=HTTP

Se podría cargar esta ACL en el firewall y siempre y cuando se cumplan los tres criterios, el mensaje será permitido.

O supongamos que desea bloquear todo el tráfico Modbus que circule a través del firewall. Se podría definir una regla que bloquea todos los paquetes que contienen 502 en el campo de puerto de destino.

El problema con este esquema es su dicotomía. Permite un determinado protocolo o lo bloquea. No es posible el control de 'grano fino', la granularidad en el análisis, del protocolo.

Esto no es lo mejor, porque los mismos protocolos SCADA/ICS no tienen granularidad. Desde el punto de vista del número de puerto, un mensaje de lectura de datos se ve similar a un mensaje de actualización del firmware (o de escritura).

Por esto si permitimos que mensajes de lectura de datos, desde un HMI (en inglés, Human Machine Interface, o Interfaz Hombre-Máquina) a un PLC, pasen a través de un firewall tradicional también se está permitiendo que los mensajes de programación (o escritura) pasen a través de él. Se trata de un problema de seguridad relevante.

El firewall tiene que ahondar en los protocolos para entender exactamente para que está siendo utilizando. Una vez se aplican las reglas tradicionales, el firewall inspecciona el contenido de los mensajes y aplica reglas más detalladas.

Por ejemplo, un firewall Modbus con DPI determina si el mensaje Modbus es un mensaje de lectura o un mensaje de escritura y luego descarta todo mensaje de escritura.

Los firewalls con DPI pueden también «sanitizar» (expresión usada para manifestar: 'transformar en seguro') el tráfico de mensajes con formato o comportamientos inusuales.

Ejemplo

Un caso real de una empresa responsable de la gestión de puertos. Un control detallado/granular (también llamado 'control de grano fino') del tráfico SCADA/ICS mejora la seguridad y confiabilidad de uno de sus sistemas. La misma utiliza PLCs en sus controles y puentes para garantizar la seguridad de las naves marítimas y el tráfico de vehículos. Asegurarse de que estos PLC no sean manipulados es crítico para la seguridad tanto de las embarcaciones como del tráfico vehicular en los puentes.

El problema que esta empresa enfrenta es que un número de computadoras necesita ejecutar continuas operaciones de acceso a los datos de los PLCs. Sin embargo sólo a los equipos de control especiales se debe permitir enviar comandos y afectar el funcionamiento de los mismos. El mecanismo de contraseña tradicional o soluciones de firewall estándares no se consideran seguros, porque no ofrecen el control de 'grano fino' o granularidad en el análisis del mensaje, necesario.

La solución fue utilizar firewalls Modbus DPI para controlar todo el tráfico a los PLCs. Sólo a los mensajes Modbus de lectura se les permite llegar a los PLCs (a excepción de algunos equipos de alta seguridad). Todos los comandos de programación Modbus remotos (es decir de escritura, no de lectura) son bloqueados por lo que la programación se limita a los ingenieros ubicados en los sitios.

Proporcionar seguridad tan solo bloqueando o permitiendo familias completas de protocolos entre las redes no es suficiente para las operaciones SCADA/ICS actuales. Los protocolos en los cuales estos sistemas se sustentan son de gran capacidad funcional y potencialmente expuestos a ataques cibernéticos. Es tiempo de considerar cómo podemos utilizar las tecnologías DPI para que nuestros sistemas de control industrial sean más seguros y confiables.

Conclusiones

Los analistas de la industria mencionan la existencia de más de siete millones de nodos MODBUS, tan solo en Estados Unidos y Europa. Por esto se torna relevante el análisis de los esquemas de seguridad implementados. Sirva tan solo a modo de referencia la evaluación de los siguientes aspectos al considerar un firewall como esquema de seguridad. No es la única evaluación factible y cada industria/empresa deberá realizar su propia evaluación, inclusión y exclusión de factores a la hora de escoger.

Funcionalidad	Packet Filter Firewalls	Stateful Firewalls	Aplicación	Deep Packet Inspection Firewall industrial
1. Definir reglas básicas sin considerar relaciones entre paquetes	SI	SI	SI	SI
2. Definir reglas básicas considerando relaciones entre paquetes	NO	SI	SI	SI
3. Definir reglas básicas a nivel aplicación (DPI).	NO	NO	Dep	SI
4. Bloquear tráfico con malware al poder segmentar por **protocolos industriales**	NO	Dep.	NO	SI
5. Definir reglas a nivel protocolo (function codes) si el protocolo es **Modbus, Ethernet/IP** o asingar un único puerto **sobre OPC**	NO	NO	NO	SI
6. Eventos y logs	SI	SI	SI	SI
7. Preparados para entornos industriales	NO	NO	NO	SI

Fuente: http://www.ciberseguridadlogitek.com/

Referencias

Good Practice Guide on Firewall Deployment for SCADA and Process Control Networks, Centre for the Protection of National Infrastructure (CPNI), London.

Morris, T. H., Jones, B. A., Vaughn, R. B. & Dandass, Y. S. (2013), Deterministic Intrusion Detection Rules for MODBUS Protocols, in «Proceedings of the 46th Hawaii International Conference on System Sciences (HICSS), 2013», IEEE, pp. 1773-1781.

NIST SP: 800-12. An Introduction to Computer Security: The NIST Handbook.

NIST SP 800-82 Rev 2, Guide to Industrial Control Systems (ICS) Security, May 2015.

Programmable Logic Controllers. A Practical Approach TO IEC 61131-3 Using CoDeSys, Autor: Dag H. Hanssen, Institute of Engineering and Safety, University of Tromso, Norway. Editorial: John Wiley & Sons, Ltd, 2015.

www.bacnet.org

www.modbus.org/specs.php

www.simplymodbus.ca/TCP.htm