Implementación del algoritmo criptográfico AES para un controlador de tráfico vehicular

Implementation of a cryptography algorithm AES on a vehicular traffic controller

Brayan Steven Higuera Neira
Ingeniero de Telecomunicaciones, ingeniero de soporte de sonda de Colombia, S.A. Bogotá, Colombia. Contacto: *brayan.higuera@co.sonda.com*

Luis F. Pedraza
Ingeniero Electrónico, magíster en Ciencias de la Información y las Comunicaciones, estudiante de Doctorado en Ingeniería de Sistemas y Computación de la Universidad Nacional de Colombia. Docente e investigador de la Universidad Distrital Francisco José de Caldas. Bogotá, Colombia.
Contacto: *lfpedrazam@udistrital.edu.co*

Fecha de recepción: 13 de noviembre de 2011 **Clasificación del artículo: investigación**
Fecha de aceptación: 16 de octubre de 2012 **Financiamiento: ninguno**

Palabras clave: AES, ethernet, Labview, microcontrolador, semáforo, tráfico vehicular.

Keywords: AES, ethernet, Labview, microcontroller, traffic light, vehicular traffic.

RESUMEN

Este artículo presenta la implementación del algoritmo criptográfico AES-128 (Advanced Encryption Standard) en un microcontrolador de 8-bits ATMega128 con base en la necesidad del envío de información confidencial de un controlador de tráfico vehicular por una red IP (Internet Protocol) a un servidor operando con el software LabVIEW (Laboratory Virtual Instrumentation Engineering Workbench). Usando las funcionalidades de la tarjeta BigAVR de Atmel, se diseña el código en lenguaje BASIC mediante el compilador MikroBasic y como interfaz de comunicación el controlador ENC28j60. Asimismo, para que sea posible establecer la comunicación encriptada entre el servidor y el controlador de tráfico, se realiza en LabVIEW el cifrado y descifrado AES, mostrando los tiempos y resultados finales de los procesos implicados en un ciclo de cifrado. Igualmente, se argumenta la importancia del uso criptográfico en la información administrada por un controlador de tráfico vehicular monitoreado a través de una central remota.

ABSTRACT

This paper presents the implementation of a cryptography algorithm, namely AES-128, running on an 8-bit micro-controller (ATMega128). This application is motivated by the need to send confidential information from a vehicular traffic controller through an IP network to reach a server that uses LabVIEW (Laboratory Virtual Instrumentation Engineering Workbench). Taking advantage of the functionalities offered by the BigAVR card, manufactured by Atmel, a code is designed and programmed using BASIC language together with the MikroBasic compiler; controller ENC28j60 was used as communications interface. In order to make encrypted communication possible between the server and the traffic controller, AES encryption and decryption was implemented in LabVIEW. The paper presents final results and implementation periods for all processes involved in an encrypting cycle. Likewise, the importance of using cryptography is highlighted when having information managed by a vehicular traffic controller monitored from a remote central unit.

* * *

1. INTRODUCCIÓN

Con el pasar de los años, la ingeniería ha observado el aumento continuo en los problemas de movilidad en el ámbito mundial, en tanto el crecimiento desmedido en la venta de vehículos frente al insuficiente desarrollo de la infraestructura vial, ha fomentado la búsqueda de soluciones alternativas a los problemas de desplazamiento diarios. Un proyecto reciente, diseñado para la ciudad de Bogotá, es un prototipo de comunicación TCP/IP [1] que permite el monitoreo y notificación de fallas en una red de semaforización para su corrección desde una central remota a través de protocolos universales.

Los controladores robustos de tráfico vehicular, ubicados en puntos estratégicos en las calles, deben comunicarse en línea con la central de monitoreo y realizar el envío de datos confidenciales referentes a distintas situaciones ocurridas en tiempo real en cada punto de la ciudad; por ejemplo, imágenes captadas por cámaras acerca de la cantidad de autos en una vía, aproximación de un vehículo de emergencia, detección de fallas en las bombillas de un semáforo, entre otros escenarios igual de importantes. Asimismo, la central recibe, procesa y retransmite información confidencial al controlador para solucionar un problema de movilidad, lo cual incluye variar los tiempos de estado de los semáforos y permitir el paso continuo de un vehículo con prioridad, entre otros aspectos [2], [3].

De allí se desprende la necesidad de que la información esté protegida de algún ataque delincuencial, pues cualquier fallo de seguridad arriesgaría la eficiencia, credibilidad y la competencia del controlador de tráfico y, peor aún, pondría en peligro el bienestar de la ciudadanía. La criptografía se presenta como un elemento esencial que debe implementarse en cualquier tipo de comunicación en línea (como internet) para garantizar un grado superior de protección a esta información confidencial, evitando fraudes y brechas de seguridad en el sistema.

2. ANTECEDENTES

Pocos estudios se han realizado para el problema de la vulnerabilidad de la información manejada en un controlador de tráfico vehicular. La documentación encontrada acerca de las implementaciones de seguridad en estos sistemas de comunicación es bastante reducida; sin embargo, aparecen dos investigaciones que, a su manera, dan confiabilidad a dicha información. En [4], se

patenta un esquema de seguridad por comunicaciones ópticas en el envío y recepción de información entre una intersección semaforizada y los vehículos de emergencia próximos a la intersección. Este método permite que, de manera segura, se transmita un código de identificación desde un emisor óptico a cada intersección semaforizada. Dicho emisor transmite pulsos de luz que representan un código encriptado que es simplemente un cifrado con una clave de encriptación variable en el tiempo de al menos un código de identificación. Un detector óptico situado en la intersección recibe los pulsos de luz transmitidos, los descifra e identificando el correcto código de identificación habilita el semáforo correspondiente para priorizar el paso del vehículo de emergencia. En [5], se implementa un algoritmo de cifrado de clave pública Diffie-Hellman con firmas digitales para la comunicación de una red VANET (Vehicular Adhoc Network) y una intersección de tráfico vehicular, de igual forma se busca que los vehículos de emergencia identificados únicamente a través de la encriptación puedan priorizar su paso por las intersecciones y estas, a su vez, puedan advertir de este evento a los demás vehículos dentro de la VANET, reduciendo accidentes y riesgos de choque. También el envío cifrado incluye información tan importante como la identificación del semáforo (coordenadas geográficas) o el tiempo restante para el cambio de luz en la intersección.

Siendo estos controladores de tráfico elementos netamente electrónicos, la implementación criptográfica debe realizarse sobre procesadores, microcontroladores u otro tipo de microchips. Acerca de este tipo de desarrollos, en [6] se da una solución de un sensor de red confiable para analizar la eficiencia de las comunicaciones a través del desarrollo de un cifrado AES en un microcontrolador ATmega644p. En [7], se demuestra que un algoritmo criptográfico puede implementarse hasta en microcontroladores de 4 bits que tienen características de muy baja potencia (5 - 60μA), su desarrollo se centra en la tecnología RFID (Radio Frequency Identification). Por otra parte, en [8] se realiza un eficiente diseño del cifrador criptográfico de bloques AES en una tarjeta inteligente construida sobre un dispositivo de desarrollo ATMega163 de arquitectura RISC (Reduced Instruction Set Computer) internamente. En [9] se presenta una autenticación criptográfica en la comunicación de una tarjeta de desarrollo 8051 a través del protocolo UDP (User Datagram Protocol), sus autores corroboran que la implementación criptográfica en dispositivos con bajo poder computacional es una necesidad potencial en las comunicaciones de las redes públicas.

3. DESCRIPCIÓN DEL ALGORITMO AES (ADVANCED ENCRYPTION STANDARD)

Anunciado por el NIST (National Institute of Standards and Technology) en 2001 [10], el algoritmo AES (Advanced Encryption Standard), también conocido como Rijndael, es un cifrador simétrico de bloque que se puede procesar en los modos de 128, 192 y 256 bits. En el modo 128 de la aplicación, la longitud del bloque de entrada, del bloque de salida y de la matriz de estado es de 128 bits, representado por Nb = 4 (número de columnas de la matriz de estado) que es el número de palabras de 32 bits en el estado. La longitud de la clave es representada por Nk = 4, 6 o 8 (número de columnas en la matriz de la clave) según los bits que la contienen, 128, 192 y 256 respectivamente. Para este caso se tendría Nk = 4.

Para el cifrado y descifrado, AES usa una función de ronda compuesta de cuatro transformaciones diferentes orientadas a los bytes [11]:

1) *SubBytes*: una sustitución de bytes usando una tabla de sustitución (S-Box).

2) *ShiftRows*: cambio de filas de la matriz de estado por distintas configuraciones.

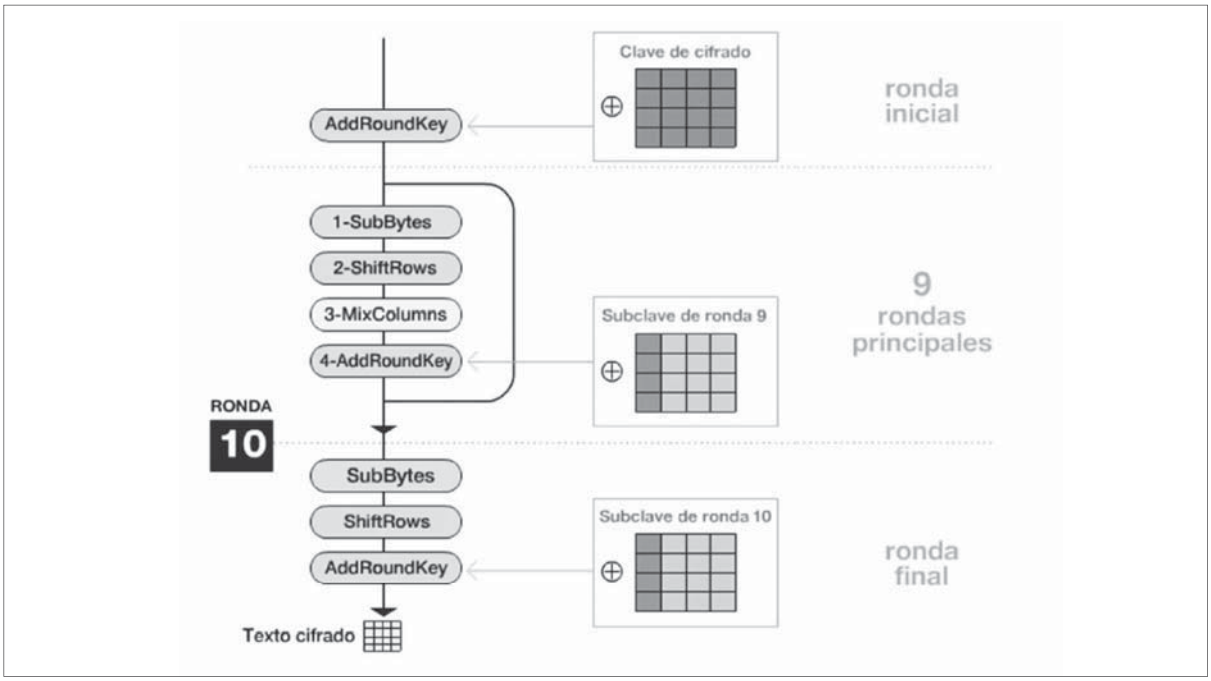

Figura 1. Proceso de cifrado AES
Fuente: elaboración propia.

3) *MixColumns*: mezcla de datos entre las columnas del vector de estado.

4) *AddRoundKey*: operación XOR entre la matriz de estado y la clave de ronda.

El proceso de cifrado consiste en aplicar de forma reiterativa estas cuatro operaciones invertibles sobre la matriz de estado de 128 bits. Estas funciones deben aplicarse dentro del proceso de acuerdo con la secuencia descrita en la figura 1.

Como se puede observar, una operación inicial *AddRoundKey* precede la primera ronda repetitiva de aplicación de las cuatro funciones. En la última ronda se aplican de nuevo todas las operaciones pero *MixColumns* es omitida.

La función de sustitución *SubBytes* realiza el remplazo lineal de los valores de la matriz de estado con los valores de la tabla S-Box. Esta operación se realiza byte a byte, tal como se muestra en la figura 2.

La permutacion ShiftRows tiene el efecto de mover los bytes a las posiciones más bajas en la fila, mientras los bytes más "bajos" saltan a la parte superior de la fila. El desplazamiento cíclico de las últimas tres filas de la matriz de estado puede observarse gráficamente en la figura 3.

La función *MixColumns* actúa sobre los bytes de una misma columna de la matriz de estado, esta transformación considera las columnas de bytes como polinomios cuyos coeficientes pertenecen a un campo de Galois. *MixColumns* toma, en su entrada, 4 bytes y genera también, a su salida, 4 bytes invertibles, tal como se observa en la figura 4. En conjunto, las funciones *ShiftRows* y *MixColumns* proveen difusión al cifrado.

En la operación *AddRoundKey,* la clave de cifrado se adiciona a la matriz de estado mediante opera-

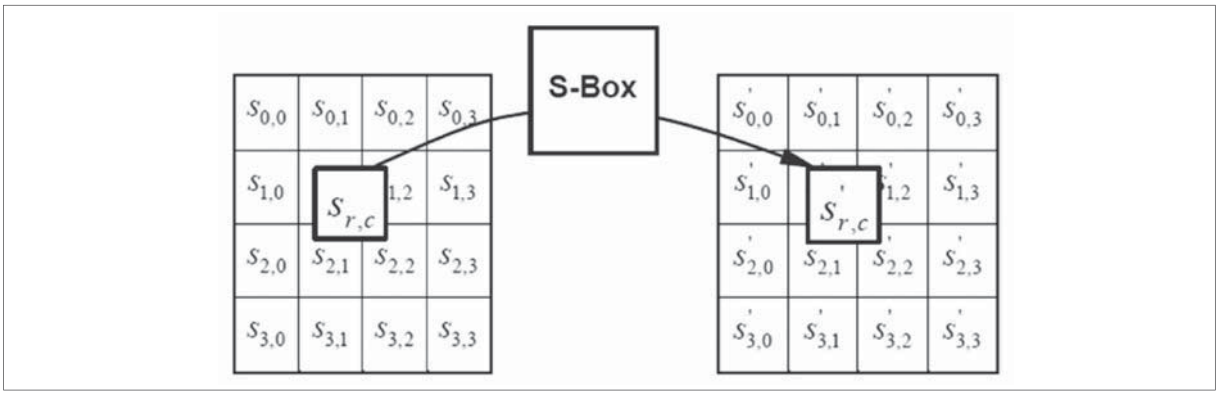

Figura 2. Sustitución *SubBytes*
Fuente: tomada de [10].

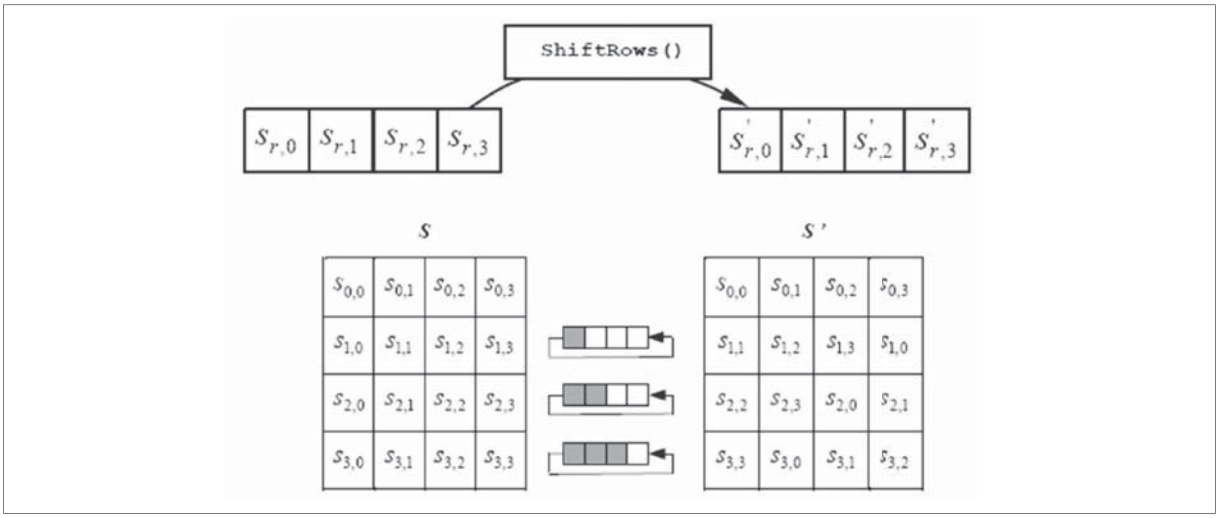

Figura 3. Permutación *ShiftRows*
Fuente: tomada de [10].

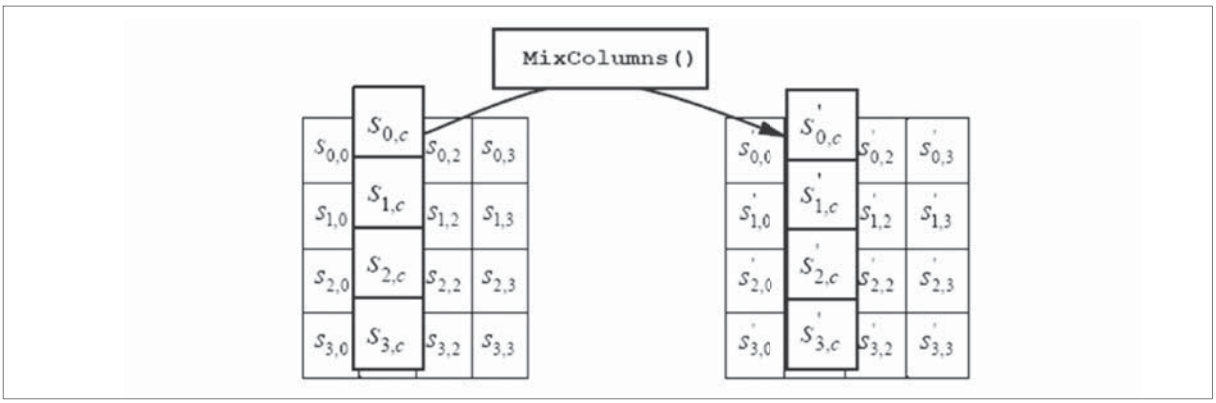

Figura 4. Transformación *MixColumns*
Fuente: tomada de [10].

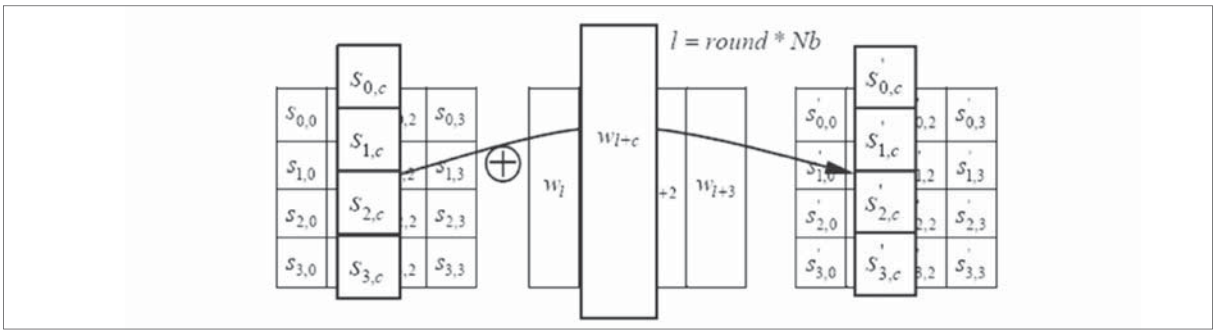

Figura 5. Operación *AddRoundKey*
Fuente: tomada de [10].

ciones XOR byte a byte. Para cada una de las rondas una subclave es derivada de la clave principal, usando el proceso de expansión de clave Rijndael, donde cada subclave es del mismo tamaño que la matriz de estado. La función *AddRoundKey* se ejecuta como se muestra en la figura 5.

Las transformaciones de cifrado pueden ser invertidas y luego implementadas en orden reverso para producir un descifrador del algoritmo AES. En efecto, las funciones que allí se utilizan son: *InvSubBytes*, *InvShiftRows*, *InvMixColumns* e *InvRoundKey*, las cuales realizan operaciones muy similares a las del proceso de cifrado, pero con elementos diferentes y mediante rondas en sentido contrario.

4. METODOLOGÍA

El esquema general de comunicación que se consigue aplicar con este desarrollo presenta una transmisión de información cifrada a través de las redes de comunicación, en algunos casos redes públicas como internet, en donde se debe intensificar la protección de esta información. Este

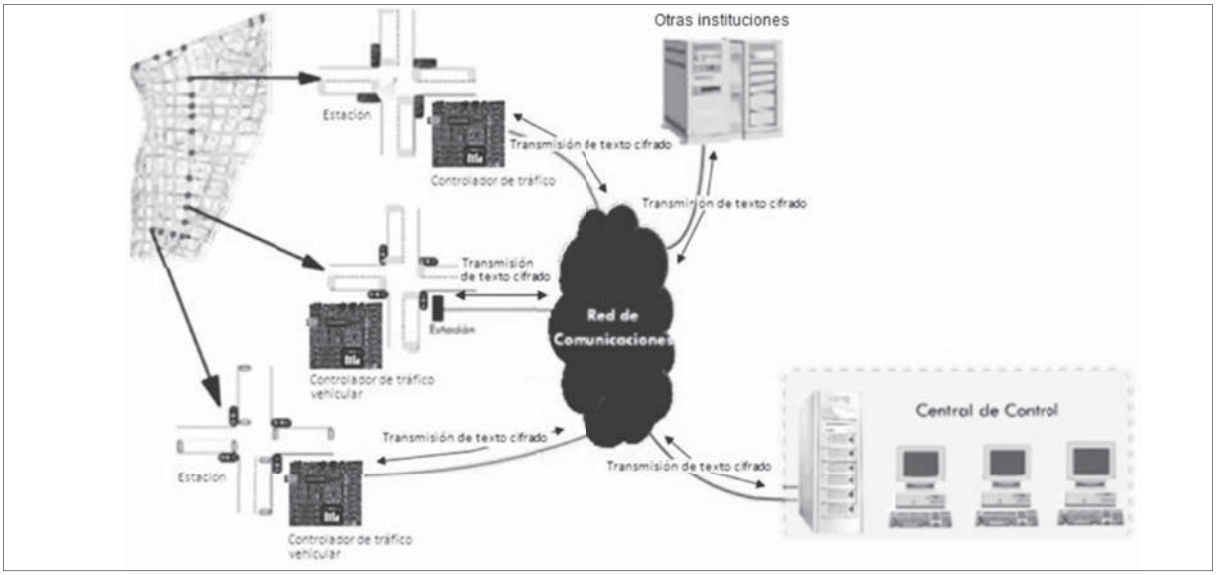

Figura 6. Esquema general de comunicación cifrada
Fuente: tomado de [1].

Tabla 1. Información transmitida

	Central	Controlador
Tipo de información y número de bytes	Fecha y hora 28 bytes	Fecha y hora 28 bytes
	Carácter "-" de separación 1 byte	Estado de la bombillas 16 bytes
	Tiempo de verde N/S y S/N 2 bytes	Reporte de vehículos de emergencia 1 byte
	Tiempo de amarillo intermitente 1 byte	
Total bytes enviados	32 bytes	45 bytes

Fuente: elaboración propia.

esquema se representa en la figura 6, en la cual se observan las partes y las vías de comunicación [1].

La información se envía en tramas que contienen la información de estado en tiempo real de la intersección semaforizada o, en el caso de la central, el envío de bytes de control de tiempos en los semáforos. En la tabla 1 se observa el tipo de información y la cantidad de bytes de la trama por cada transmisión del controlador o de la central.

Para establecer una comunicación segura entre el controlador de tráfico vehicular y la central de monitoreo, la aplicación del algoritmo debe hacerse de forma integral en cada una de las partes, es decir, los procesos de cifrado y descifrado AES deben implementarse tanto en el microcontrolador como en el software de la estación de monitoreo, que para este caso es LabVIEW. El proceso de cifrado AES fue programado bajo el mismo patrón en los 2 entes de la comunicación (microcontrolador y central de monitoreo), de manera que se consiga una simetría de encriptación en todo el sistema; dicho patrón o modelo se puede observar en el diagrama de la figura 7. De igual manera, el proceso de descifrado tiene una estructura similar, donde solo cambian las rondas, que

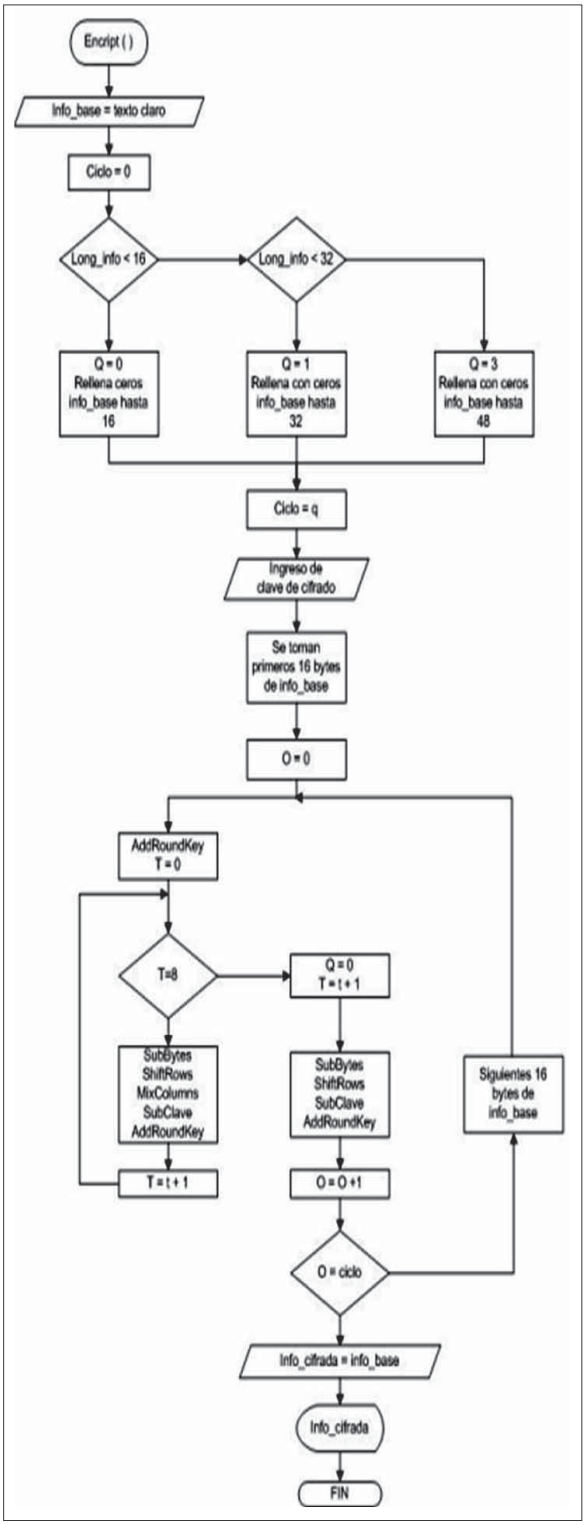

Figura 7. Diagrama de flujo del cifrado
Fuente: elaboración propia.

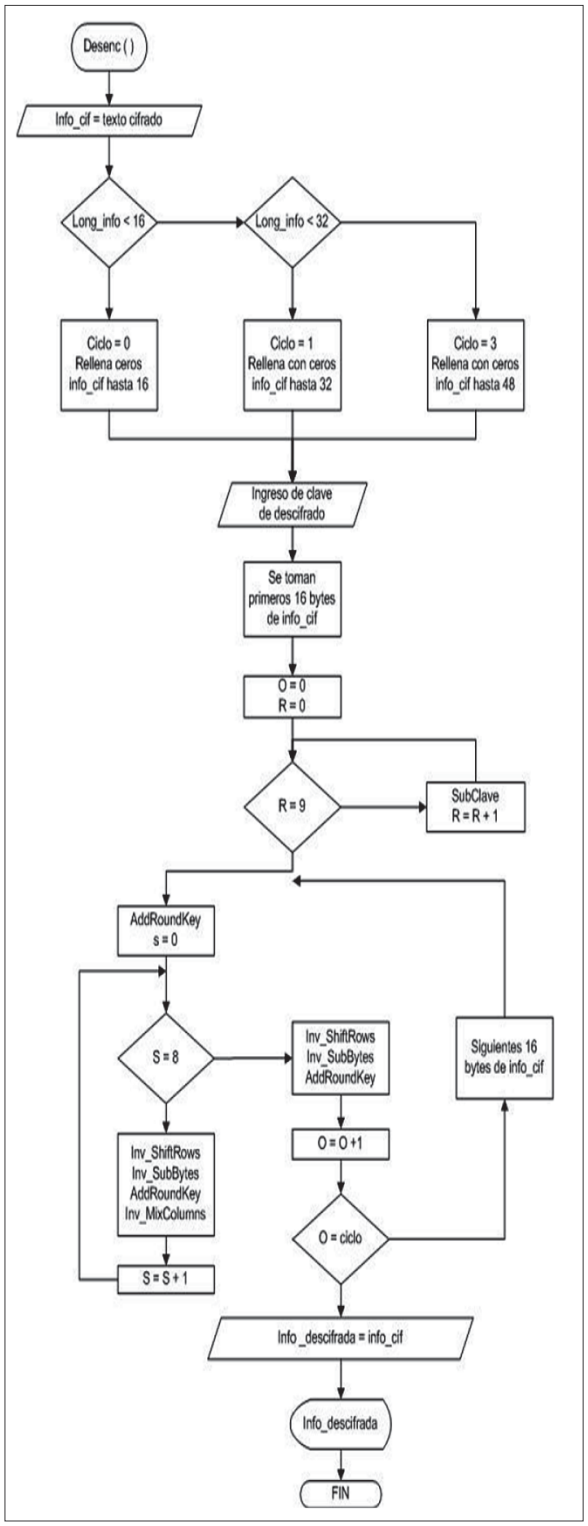

Figura 8. Diagrama de flujo del descifrado
Fuente: elaboración propia.

se realizan con funciones invertidas y en distinto orden en comparación con el proceso de cifrado. El diagrama del descifrado se puede observar en la figura 8.

4.1 Aplicación en el microcontrolador

La presente aplicación del algoritmo de encriptación AES se desarrolla sobre un microcontrolador ATmega128 CMOS (Complementary Metal-Oxide Semiconductor) de 8 bits de baja potencia basado en arquitectura RISC. El ATmega128 de Atmel logra un rendimiento de aproximadamente 1 MIPS (Millions Instructions per Seconds) por MHz. De igual manera, con este microcontrolador se tienen 128 Kbytes de memoria flash programables, programa de cerradura para seguridad de software, interfaz SPI (Serial Peripheral Interface) para programación In-system, entre otras ventajas.

Como herramienta de desarrollo se utiliza el compilador MikroBasic Pro para AVR, también del fabricante Atmel, con el fin de obtener un avanzado lenguaje BASIC. La programación del microcontrolador se realiza mediante el módulo SPI, así como la comunicación con la tarjeta Ethernet encargada de la conexión en red del ATmega128. Mediante MikroBasic se logran obtener todos los datos relacionados con los resultados del proceso de cifrado, como: tiempos de respuesta, cambios en variables, entre otros.

Dos librerías, una para el cifrado llamada "Lib_AES_Encript.mbas" y una para el descifrado nombrada como "Lib_AES_Desenc.mbas", inicialmente configuradas para su programación en microcontroladores ATmega128, tienen la disponibilidad de ser llamadas en cualquier programa diseñado para la comunicación con microcontroladores. Desde el programa principal, ejecutado por el controlador de tráfico vehicular, se inicia el cifrador AES mediante la función *encript( )*,

la cual ya tiene predefinida la clave del sistema. Asimismo, se indica en la variable *long_info* la cantidad de bytes a encriptar (máximo 48 bytes). A continuación se presenta un ejemplo del proceso de cifrado de una matriz de 16 bytes con todos sus valores en 0xFF:

for*i=0* ***to*** *37*

info_base[i]=0xff

next*i*

long_info= 38

encript()

Al finalizar la función *encript( ),* los datos cifrados (38 bytes) se almacenan en la misma matriz *info_base,* de tal manera que el programa disponga de la información encriptada desde esta misma matriz y no consuma recursos de memoria en una nueva variable. La estructura del llamado a un ciclo de descifrado AES es bastante similar al anterior, solo cambian las variables y en este caso el nombre de la función a la cual se hace el llamado que es *desenc()*.

4.2 Estación de monitoreo

La central de monitoreo como base principal de operaciones de la red de semaforización tiene la capacidad de supervisar cada controlador de tráfico mediante una comunicación segura. Para esto, el panel de control está diseñado sobre la plataforma LabVIEW, es por ello que la implementación AES se realiza sobre este mismo software adaptando el cifrado a la aplicación de monitoreo previamente desarrollada.

Así como en el microcontrolador, se implementa AES en la central en la que se organiza el cifrado y descifrado en 2 módulos de fácil aplicación para cualquier proyecto de comunicación desarrollado con LabVIEW. De igual manera, estos 2 módulos principales realizan eventuales llamados a otros Sub-VIS encargados de realizar tareas específicas como la generación de subclaves o la multiplicación de números dentro del campo de Galois.

Cada notificación generada en el controlador de tráfico vehicular es cifrada y enviada a través de una red IP a la central de monitoreo, allí la aplicación en LabVIEW se encarga de descifrar y organizar la información para, finalmente, almacenarla y desplegarla al operador del punto de monitoreo en un archivo .txt para una mejor interpretación de esta, tal como se muestra en la figura 9.

Consecuentemente, el operador del centro de monitoreo verifica esta información y procede

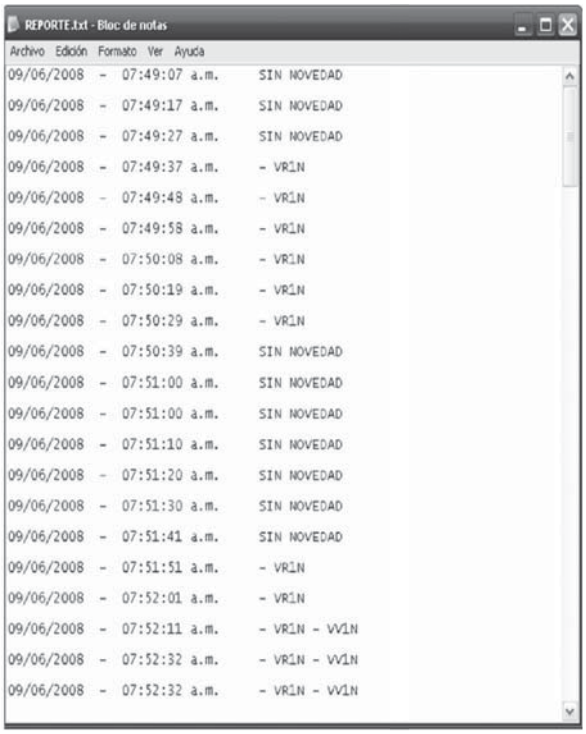

Figura 9. Archivo .txt de reporte de novedades
Fuente: elaboración propia.

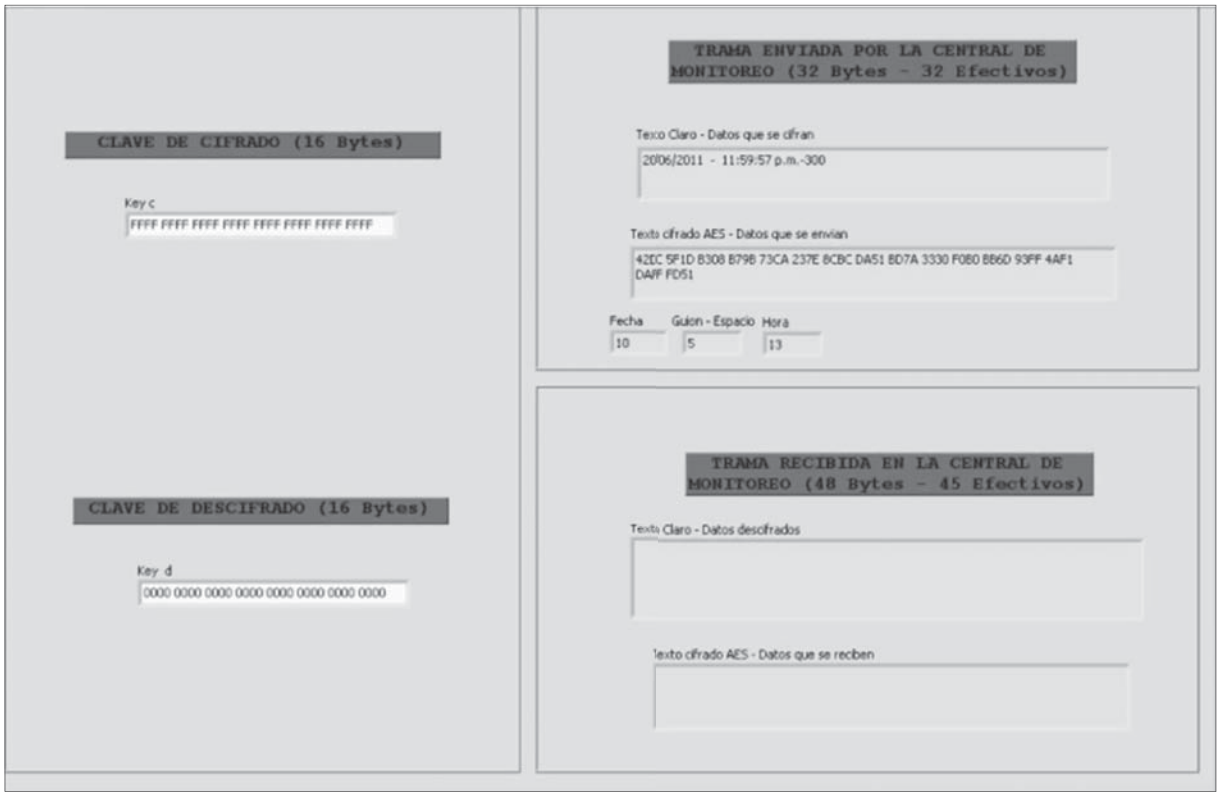

Figura 10. Pestaña "Cifrado AES" en central de monitoreo
Fuente: elaboración propia.

a cifrar y remitir ciertas órdenes al controlador para que, a su vez, se determine qué tipo de acción debe tomarse en cada caso. En el panel de control de la central de monitoreo se adiciona la pestaña "cifrado AES" mediante la cual se logran controlar los datos cifrados salientes y entrantes, los datos descifrados y su longitud, y lo que es aún más importante, son habilitados 2 espacios numéricos hexadecimales para la configuración de la clave de cifrado y descifrado del procesador AES de LabVIEW.

Finalmente, la central de monitoreo puede tener una visión específica de la información confidencial que transmite continuamente al controlador de tráfico. El estudio de los datos presentados en el panel de control, como se observa en la figura 10, puede proporcionar certeza en la veracidad de la información y facilitar cualquier plan de acción ante un posible fallo de seguridad.

5. ANÁLISIS DE RESULTADOS

Mediante las pruebas realizadas se observa una efectividad del 100 % en cuanto a la encriptación de datos, tal como se aprecia en la figura 11, siempre y cuando no se presenten fallas continuas en el canal de comunicación que modifiquen, agreguen o supriman bits ocasionando errores en la interpretación final de los datos.

Es altamente improbable que existan claves débiles o semidébiles en AES, debido a la estructura de su diseño, que busca eliminar la simetría en las subclaves. También se ha comprobado que es

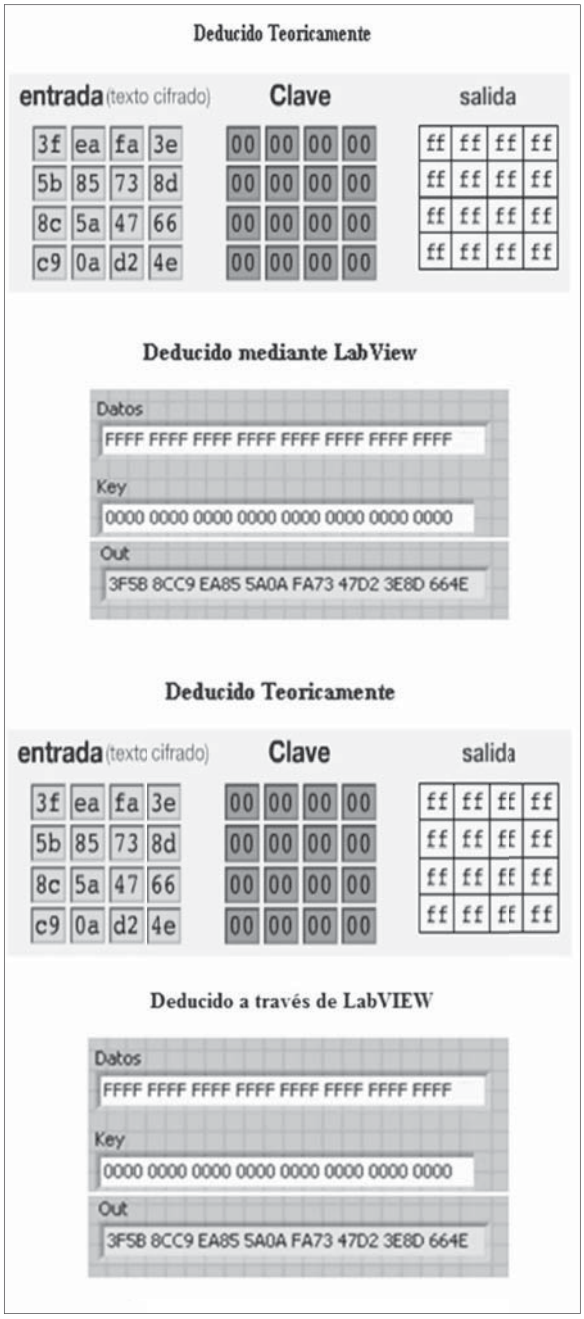

Figura 11. Resultado de encriptación
Fuente: elaboración propia.

resistente a criptoanálisis, tanto lineal como diferencial [12]. En cuanto a los ataques por fuerza bruta, se entiende que la resistencia del algoritmo es proporcional a la longitud de la clave usada.

Tabla 2. Tiempo medio de criptoanálisis para quebrantar una clave simétrica

Longitud de la clave (bits)	Tiempo necesario para quebrantar la clave
40	2 segundos
48	9 minutos
56	40 horas
64	14 meses
72	305 años
80	78 250(2^{16}) años
96	5 127 160 311(2^{32}) años
112	336 013 578 167 538(2^{48}) años
128	22 020 985 858 787 784 059(2^{64}) años

Fuente: elaboración propia.

La tabla 2 muestra la cantidad de tiempo requerido para romper por fuerza bruta la clave de un algoritmo simétrico en razón de la longitud de su clave de cifrado.

Adicionalmente, se tiene la ventaja de que el módulo responsable de la comunicación Ethernet del controlador posee un segmento CRC (código de redundancia cíclica) que permite encontrar, si existe, la alteración de datos durante la transmisión o almacenamiento. Teniendo en cuenta la eficiencia de este procesador de cifrado AES en cuanto a los tiempos de ejecución, la tabla 3 expone detalladamente dichos tiempos para las funciones ejecutadas por el microcontrolador, trabajando a una frecuencia de 8 MHz y con una longitud de información de 48 bytes.

El proceso más lento se detecta en las funciones *Inv_MixColumns* y *MixColumns*, en las cuales se realizan repetidas operaciones XOR dentro del campo de GaloisGF(). Para la central de monitoreo es bastante complicado realizar las mismas mediciones, debido a que su duración normalmente es menor a 1 segundo trabajando con una longitud de datos de 48 bytes o menos y LabVIEW no tiene una herramienta *stopwatch* para determinar dichos valores.

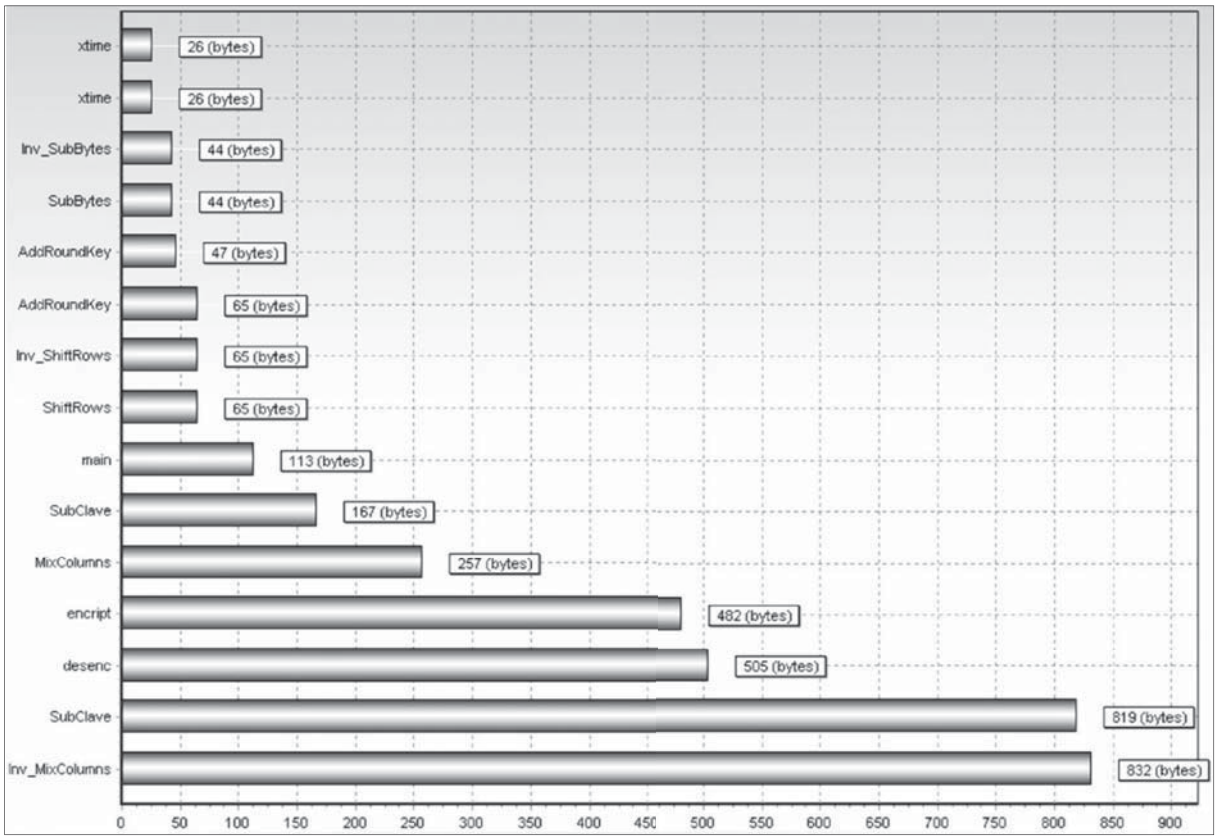

Figura 12. Numero de bytes empleados por cada función
Fuente: elaboración propia.

Consecuentemente, la cantidad de instrucciones y variables programadas en cada una de las funciones resulta responsable de la cantidad de tiempo de ejecución de las mismas, esto quiere decir que, normalmente, a mayor número de instrucciones es superior la cantidad de procesos a realizar y por ende mayor la duración total de la función. La figura 12 muestra el número de bytes ocupados por las variables e instrucciones de cada función.

6. CONCLUSIONES Y RECOMENDACIONES

La aplicación del algoritmo AES, en el controlador de tráfico vehicular y en la central de monitoreo, garantiza la transmisión de información con un alto grado de confiabilidad a través de cualquier red IP, ya que estos datos, al ser vulnerables, pueden engañar al sistema y generar caos en las redes de semaforización, lo que provocaría un colapso en la movilidad de cualquier ciudad o municipio.

Es indispensable el estudio de criptografía en dispositivos de bajo poder computacional, como los microcontroladores, en vista de su implementación en redes inteligentes de semaforización, debido a la necesidad de asegurar la información que se transmite hacia la central de monitoreo a través de redes públicas, para reducir riesgos de pérdida y suplantación de información por su captura ilegal.

Por lo general, un microcontrolador que es utilizado para transmitir información a través de una red, carece de algún método que proporcione seguridad y confiabilidad en los mensajes enviados, este puede ser uno de los grandes enigmas del uso de dispositivos electrónicos de bajo poder computacional en sistemas de comunicación. Con la aplicación eficaz de un algoritmo criptográfico, como AES, se obtiene una solución a bajo costo y confiable para proteger la información.

Debido a la complejidad en el proceso del módulo de descifrado de la central de monitoreo, se recomienda realizar la instalación del software LabVIEW sobre un servidor con las características de robustez adecuadas, para que el desarrollo del algoritmo AES se realice en cortas fracciones de tiempo, de tal manera que se reduzca la duración total de ejecución del módulo *decrypt* y se agilice el proceso de obtención de la información.

En proyectos que desean implementar el cifrado de información, puede ocurrir que, entre el transmisor y el receptor, sea complejo realizar el intercambio o la programación inicial de las claves; por tal razón, las librerías de cifrado AES podrían complementarse con el diseño de un algoritmo criptográfico de clave pública que permita resolver fácil y eficazmente el problema del intercambio inicial de claves.

Al adoptar el diseño de un algoritmo criptográfico para dotar de confiabilidad al envío de información de la central de monitoreo hacia el controlador y viceversa, es necesario que el host usado en la central posea medidas de seguridad, con el fin de evitar otro tipo de ataques que afecten la comunicación o permitan el espionaje, dejando así al descubierto las claves del sistema. Asimismo, es recomendable implementar métodos para contrarrestar esto, como un antivirus actualizado, un antispyware, entre otros.

REFERENCIAS

[1] L. Ramirez y R. Molina, *Prototipo de un Sistema de Comunicacion TCP/IP para la deteccion de fallas en un controlador de tráfico vehicular*, [Tesis Ingeniería Electrónica], Universidad Cooperativa de Colombia, 2008.

[2] O. Salcedo, L.F. Pedraza y C.A. Hernández, "Modelo de Semaforización Inteligente para la Ciudad de Bogotá", *Revista Ingenieria Universidad Distrital,* vol. 11, no 2, pp. 61-69, 2006.

[3] L.F. Pedraza, C.A. Hernandez y D.A. Lopez, "Control de tráfico vehicular usando ANFIS", *Ingeniare. Rev. chil. ing*, vol. 20, no 1, pp. 79-88, 2012. [en línea]. Disponible: http://www.scielo.cl/scielo.php?pid=S0718-33052012000100008&script=sci_arttext

[4] M. Schwartz and C. Meyer, *LED emitter for optical traffic control systems*, Patente US 8072346 B2, 2011.

[5] F. Dotzer, F. Kohlmayer, T. Kosch and M. Strassberger, "Secure communication for intersection assistance", in *Proceedings of the 2nd International Workshop on Intelligent Transportation*, Hamburg, Germany, 2005.

[6] H. Lee, K. Lee, and Y. Shin, "AES Implementation and Performance Evaluation on 8-bit Microcontrollers", *International Journal of Computer Science and Information Security*, 2009.

[7] M. Vogt, A. Poschmann and C. Paar, "Cryptography is feasible on 4-Bit microcontrollers - A proof of concept", *IEEE International Conference on RFID*, pp. 241-248, April, 2009.

[8] K. Schramm and C. Paar, "IT security project: implementation of the Advanced Encryption Standard (AES) on a smart card", *Information Technology: Coding and Computing*, 2004. "Proceedings. ITCC 2004". *International Conference,* vol. 1, pp. 176- 180, April, 2004.

[9] B. Groza, P. Murvay, I. Silea and T. Ionica, "Cryptographic Authentication on the Communication from an 8051 Based Development Board over UDP", *Internet Monitoring and Protection,* 2008. "ICIMP '08". *The Third International Conference*, pp. 92-97, 2008.

[10] National Institute of Standards and Technology (NIST), "ADVANCED ENCRYPTION STANDARD (AES)", *Federal Information Processing Standards* (FIPS) Publication 197. November 2001. [En línea] Available: http://csrc.nist.gov

[11] J. Aguirre, *Libro Electrónico de Seguridad Informática y Criptografía*, Madrid: Universidad Politécnica de Madrid, 2006.

[12] M. Lucena, *Criptografía y Seguridad en Computadores*, Jaén: Escuela Politécnica Superior de España, 4ª ed, Creative Commons, 2010.