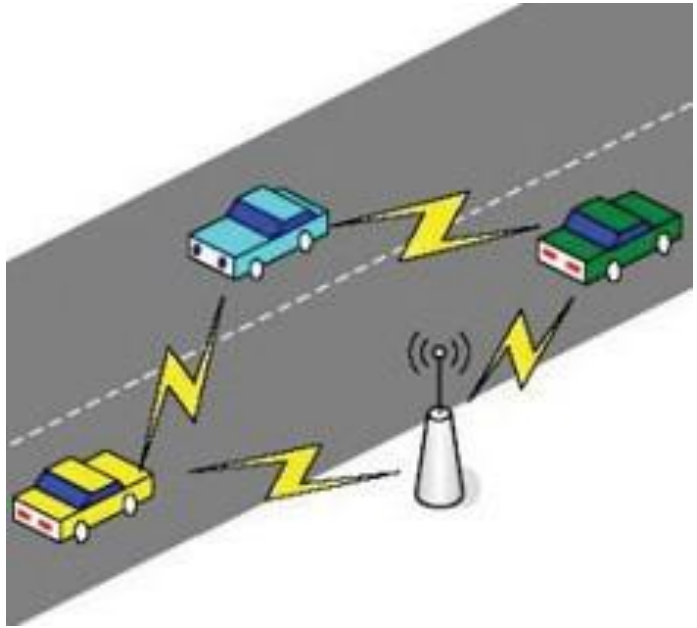
Comunicação segura em VANET

João L. M. Freitas, Leonardo da R. Souza, Patrick R. Sardou, Nilson M. Lazarin
Bacharelado em Sistemas de Informação
Centro Federal de Educação Tecnológica Celso Suckow da Fonseca (Cefet/RJ) – Nova Friburgo, RJ
{joao.frhb,leorsouza15,ptrcksardou,nilsonmori}@gmail.com

Introdução

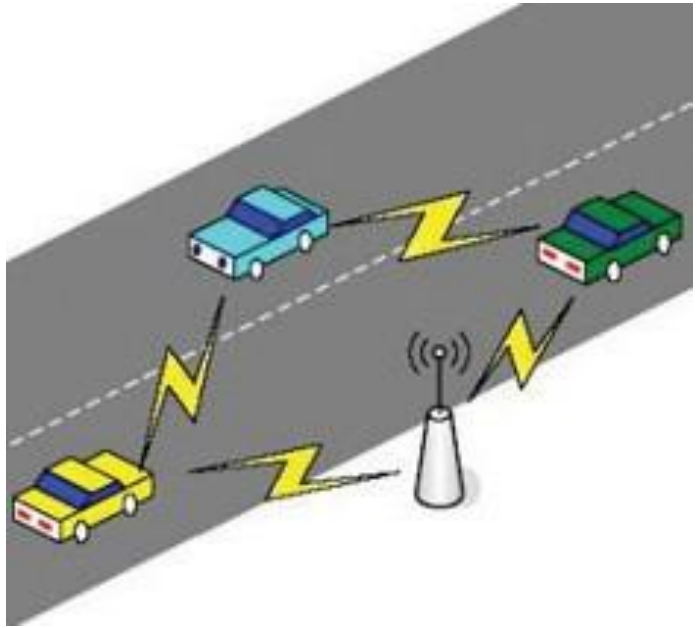
As redes ad-hoc veiculares (VANETs) integram protocolos de conectividade móvel para agilizar a transferência de dados entre veículos bem como equipamentos em estradas. Na VANET, o dispositivo sem fio envia informações para veículos próximos e as mensagens podem ser transmitidas de um veículo para outro [Sabahi 2011].



João L. M. Freitas, Leonardo da R. Souza, Patrick R. Sardou, Nilson M. Lazarin
{joao.frhb,leorsouza15,ptrcksardou,nilsonmori}@gmail.com

Introdução

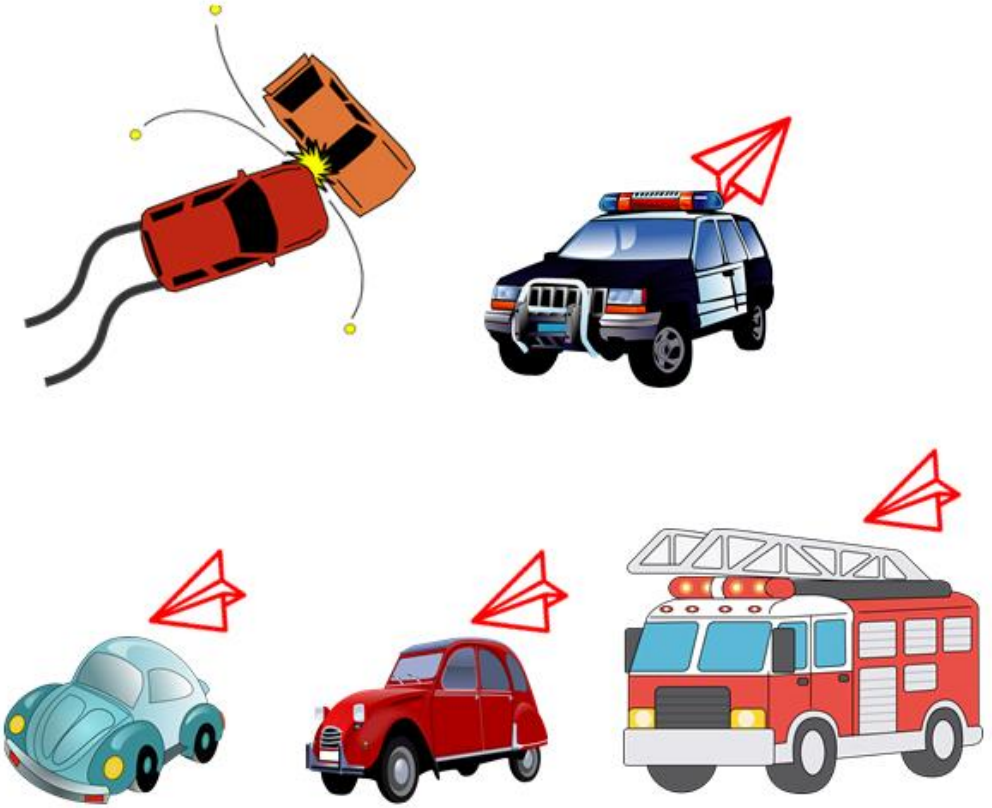
Semelhante a outras tecnologias, na VANET existem alguns problemas importantes e perceptíveis. Um dos mais importantes deles é a segurança. Uma vez que a rede é aberta e acessível de qualquer lugar no alcance do rádio VANET, a interferência nessa comunicação torna-se um alvo fácil para usuários mal-intencionados [Sabahi 2011].



João L. M. Freitas, Leonardo da R. Souza, Patrick R. Sardou, Nilson M. Lazarin
{joao.frhb,leorsouza15,ptrcksardou,nilsonmori}@gmail.com
CEFET/RJ

Objetivo

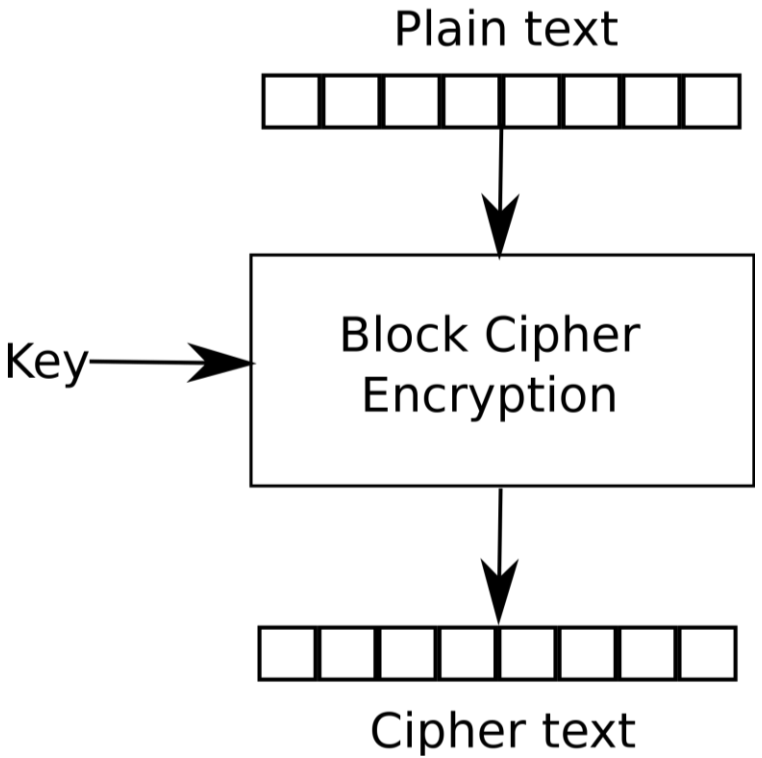
Possibilitar a troca segura de mensagens em uma rede VANET, realizando assim a criptografia na transmissão de mensagens entre módulos RF (Rádio Frequência).

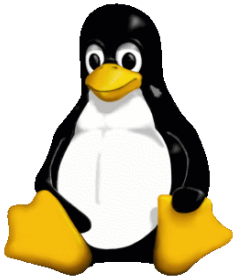


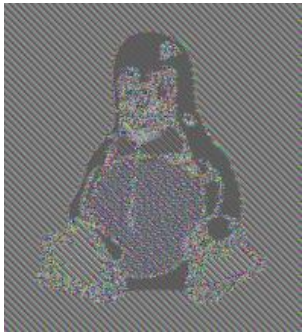
João L. M. Freitas, Leonardo da R. Souza, Patrick R. Sardou, Nilson M. Lazarin
{joao.frhb,leorsouza15,ptrcksardou,nilsonmori}@gmail.com

Fundamentação Teórica

No modo de operação ECB (*Electronic Codebook*), sob uma determinada chave, qualquer bloco de texto simples sempre é criptografado para o mesmo bloco de texto cifrado [Dworkin 2001].





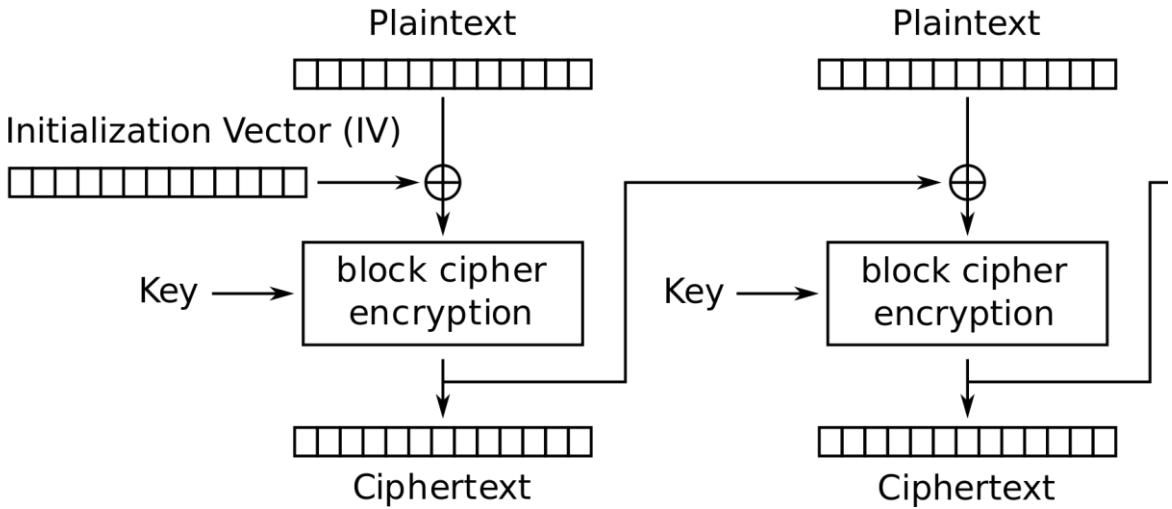


João L. M. Freitas, Leonardo da R. Souza, Patrick R. Sardou, Nilson M. Lazarin
{joao.frhb,leorsouza15,ptrcksardou,nilsonmori}@gmail.com
CEFET/RJ

Fundamentação Teórica

O modo de operação CBC (Cypher Block Chaining) realiza a combinação dos blocos de texto simples com os blocos de texto cifrado anteriores.

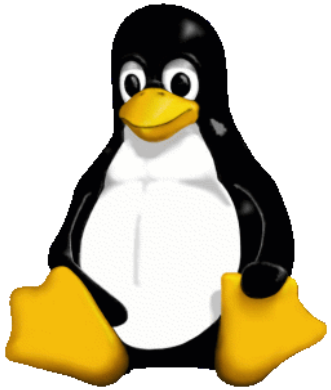
O modo CBC requer um vetor de inicialização (IV) para combinar com o primeiro bloco de texto simples [Dworkin 2001].

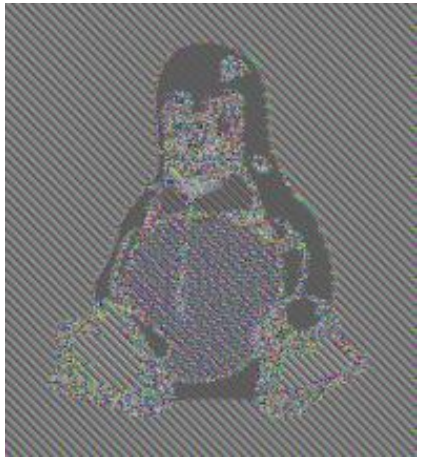


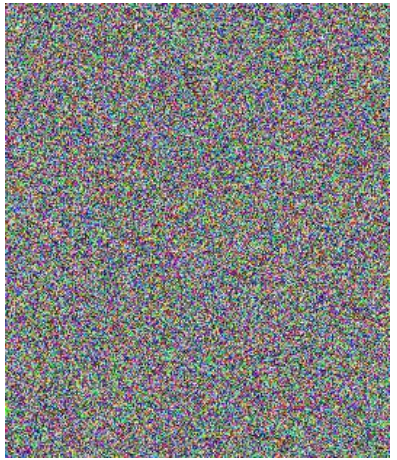
João L. M. Freitas, Leonardo da R. Souza, Patrick R. Sardou, Nilson M. Lazarin
{joao.frhb,leorsouza15,ptrcksardou,nilsonmori}@gmail.com
CEFET/RJ

Fundamentação Teórica

Em seu processo de criptografia, o CBC pode operar com o vetor de inicialização de duas formas, mantendo-o fixo para a cifragem de cada bloco da mensagem, ou obtendo esse vetor de inicialização de modo aleatório [Dworkin 2001].



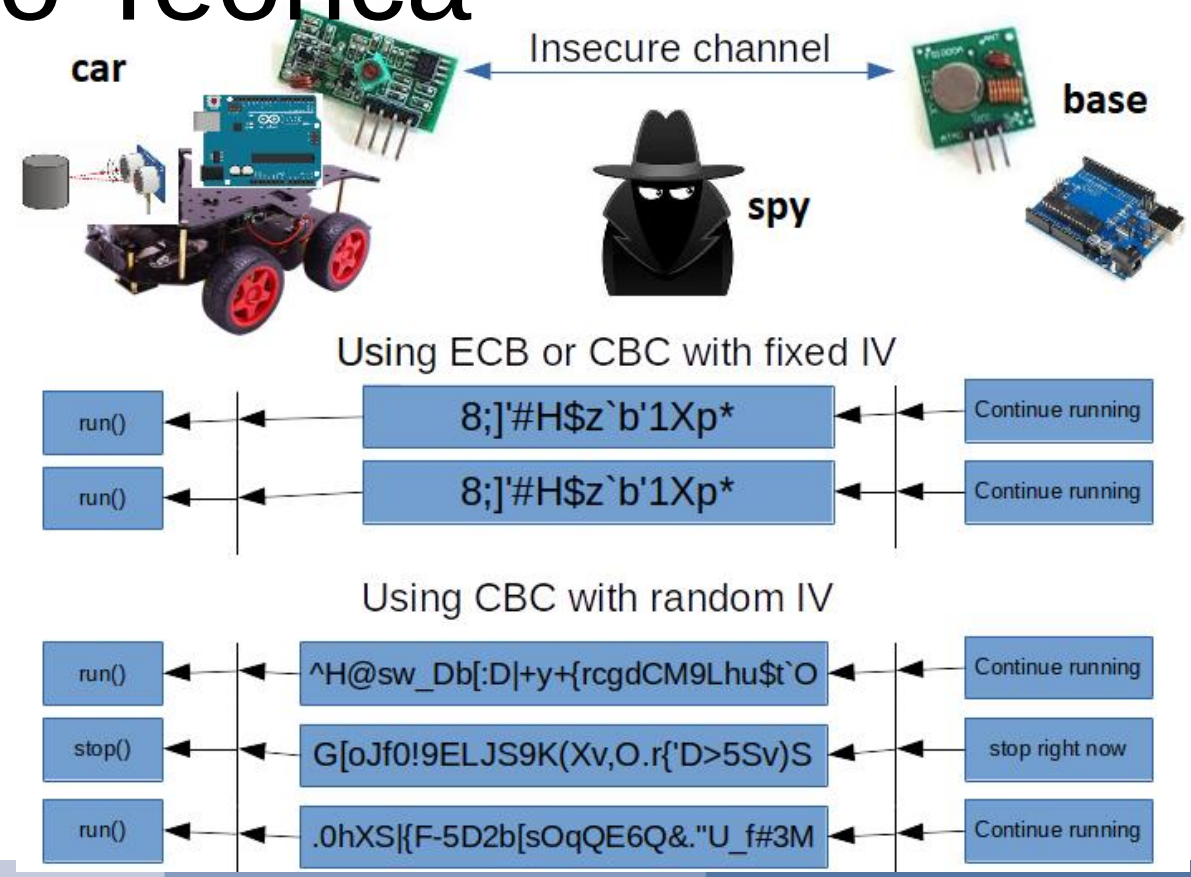




João L. M. Freitas, Leonardo da R. Souza, Patrick R. Sardou, Nilson M. Lazarin
{joao.frhb,leorsouza15,ptrcksardou,nilsonmori}@gmail.com
CEFET/RJ

Fundamentação Teórica

No cenário onde um veículo recebe comandos remotos de uma base. Neste ambiente os dados podem ser capturados por um intruso e uma vez que utilizando ECB ou CBC com IV fixo a mensagem cifrada será também fixa.



João L. M. Freitas, Leonardo da R. Souza, Patrick R. Sardou, Nilson M. Lazarin
{joao.frhb,leorsouza15,ptrcksardou,nilsonmori}@gmail.com
CEFET/RJ

Trabalhos Relacionados

O estudo realizado por [Rocha et al. 2020] sobre o desempenho e conformidade das bibliotecas de criptografia para aplicação na internet das coisas, apresenta um comparativo entre bibliotecas criptográficas para o Arduino.

Os resultados obtidos demonstram que a biblioteca Securino se mostra superior às outras analisadas, por atender à todas as especificações do AES (*NIST FIPS PUB 197 e SP 800-38-A*), garantindo um alto nível de segurança.

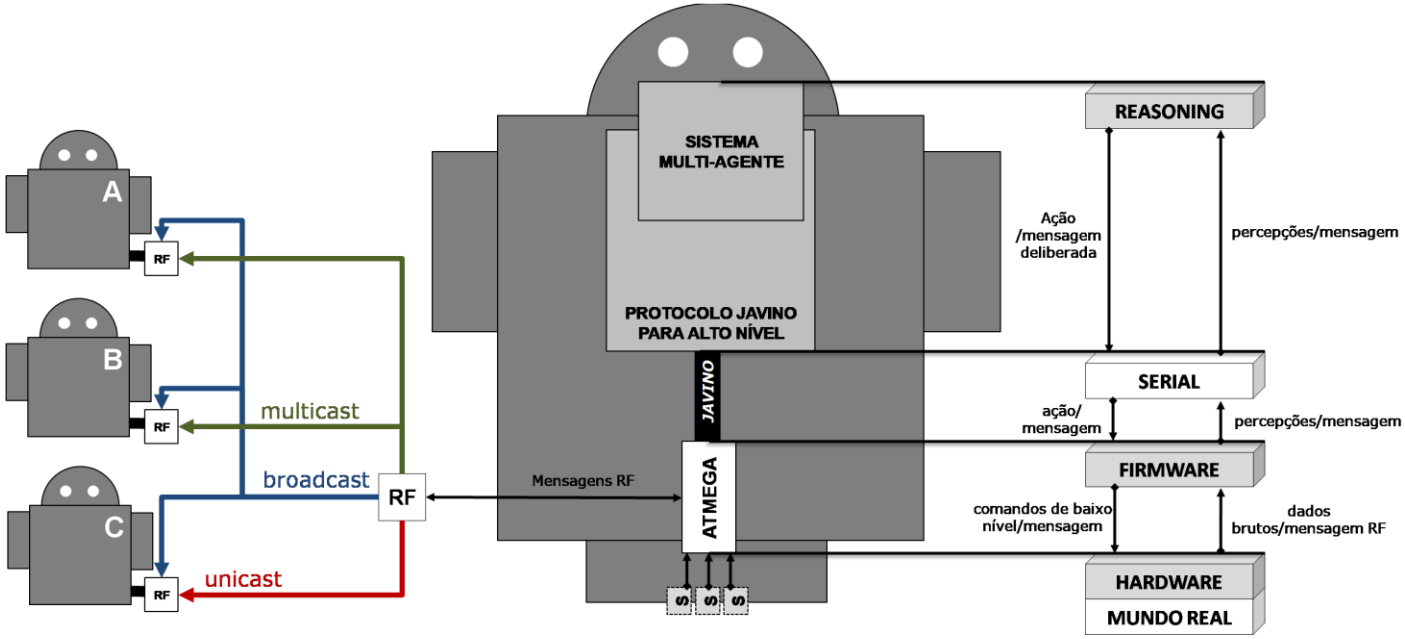
Entretanto, a biblioteca apresentada não possibilita a comunicação RF.

Biblioteca	Chave	Modo	Licença	Teste de Embarcação	NIST FIPS 197	NIST SP 800-38A
AES for microcontrollers (Arduino & Raspberry pi) https://github.com/spaniakos/AES	128 192 256	CBC	Copyright	OK	X	X
AES_128bit_Arduino https://github.com/beranm14/AES_128bit_Arduino	128	ECB	-	OK	OK	X
AESLib https://github.com/DavyLandman/AESLib	128 192 256	ECB CBC	GPL	OK	OK	X
AES library for Arduino Board https://github.com/indrabagus/arduino-aes	128	CBC	Apache	OK	OK	X
ArduinoAES256 https://github.com/qistoph/ArduinoAES256	256	ECB	Copyright	OK	X	X
Arduino-AES https://github.com/DanielVukelich/Arduino-AES	128 92 256	ECB	GPL	X	-	-
Arduino-AES https://github.com/edogaldo/Arduino-AES	128 192 256	ECB CBC	Copyright	OK	OK	X
Arduino Cryptography Library https://github.com/rweather/arduinolibs	128 192 256	ECB CBC	MIT	OK	OK	X
Micro-aes https://github.com/DarkCaster/Micro-AES-Arduino	128 192 256	ECB CBC	MIT	OK	OK	X
Ptolemy-XV https://github.com/octaviovieira/Ptolemy-XV	128 192 256	ECB	-	X	-	-
Securino https://github.com/nilsonmori/securino	128	ECB CBC	GPL	OK	OK	OK

João L. M. Freitas, Leonardo da R. Souza, Patrick R. Sardou, Nilson M. Lazarin
{joao.frhb,leorsouza15,ptrcksardou,nilsonmori}@gmail.com
CEFET/RJ

Trabalhos Relacionados

O protocolo de comunicação apresentado por [Lazarin et al. 2021], estendeu a biblioteca Javino, possibilitando que agentes BDI embarcados possam se comunicar como outros agentes embarcados de um SMA distinto de forma efetiva, através de RF, com mensagens ***unicast***, ***multicast*** e ***broadcast***.

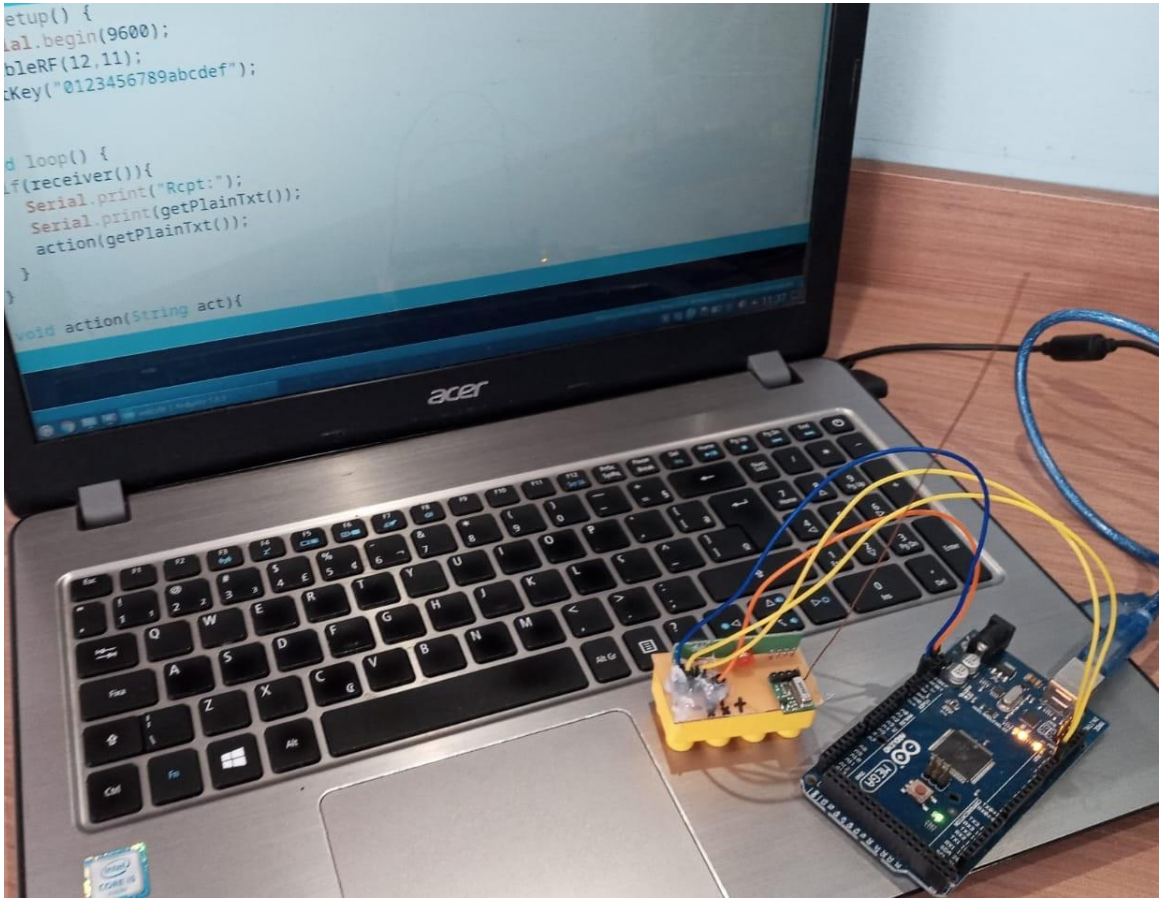


Entretanto o protocolo não implementa segurança na comunicação, trafegando as informações em claro.

João L. M. Freitas, Leonardo da R. Souza, Patrick R. Sardou, Nilson M. Lazarin
{joao.frhb,leorsouza15,ptrcksardou,nilsonmori}@gmail.com
CEFET/RJ

Implementação

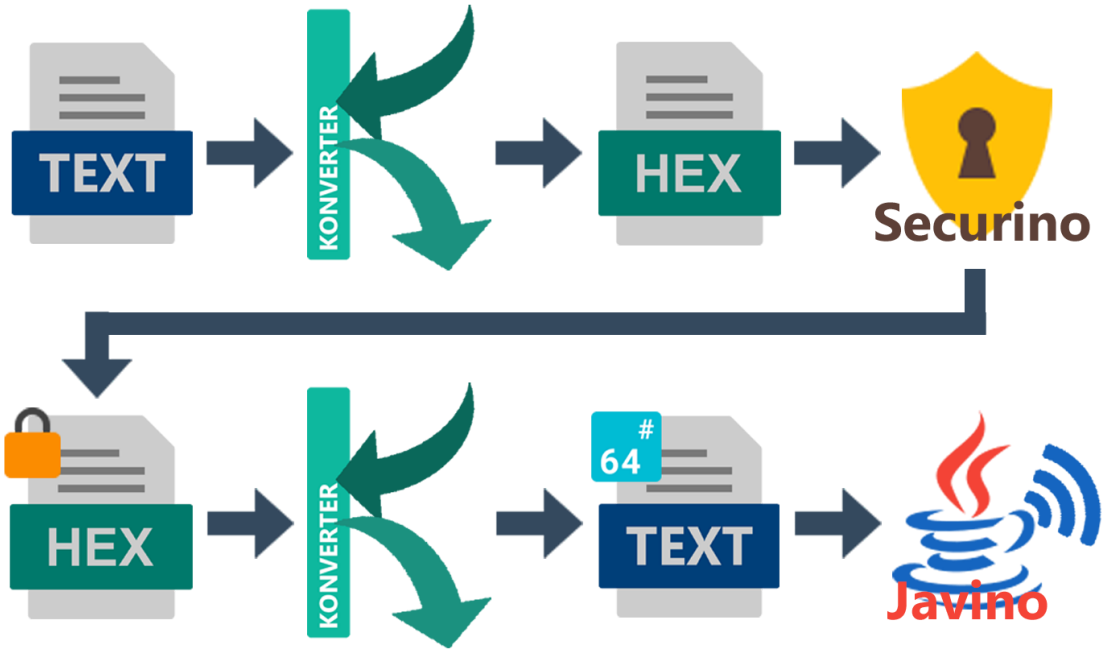
Para a implementação do projeto foram utilizadas duas placas Arduino Mega, com módulos TX/RX RF 433MHz.



João L. M. Freitas, Leonardo da R. Souza, Patrick R. Sardou, Nilson M. Lazarin
{joao.frhb,leorsouza15,ptrcksardou,nilsonmori}@gmail.com
CEFET/RJ

Implementação

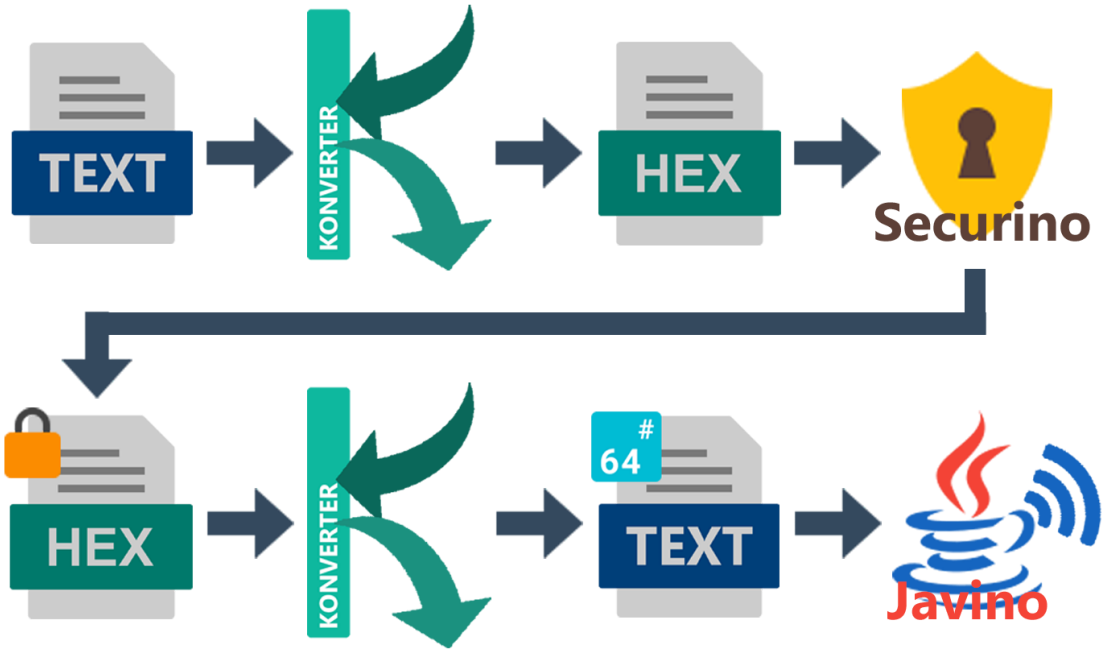
A biblioteca Securino é a responsável por realizar a criptografia da mensagem que será enviada. Esta biblioteca, trabalha com um vetor de bytes hexadecimal. Foi necessária a criação de uma biblioteca adicional para a conversão de texto puro para hexadecimal e vice-versa, nomeada Konverter.



João L. M. Freitas, Leonardo da R. Souza, Patrick R. Sardou, Nilson M. Lazarin
{joao.frhb,leorsouza15,ptrcksardou,nilsonmori}@gmail.com
CEFET/RJ

Implementação

Após cifrar a mensagem, a biblioteca Javino realiza o envio da mensagem através do módulo RF. Porém, é necessário uma nova conversão pois o Javino trabalha apenas com codificação de texto base64.



João L. M. Freitas, Leonardo da R. Souza, Patrick R. Sardou, Nilson M. Lazarin
{joao.frhb,leorsouza15,ptrcksardou,nilsonmori}@gmail.com
CEFET/RJ

Experimentos

base | securityvanet

```
void setup(){
  Serial.begin(9600);
  enableRF(12,11);
  setKey("0123456789abcdef");
}

void loop() {
  String cmd="command=go ahead";
  comm(cmd);
  delay(1000);
  cmd="command=stop now";
  comm(cmd);
  delay(1000);
}

void comm(String str){
  Serial.print("Send:");
  Serial.print(str);
  trasmiter(str);
  Serial.println("    [OK]");
}
```

spy

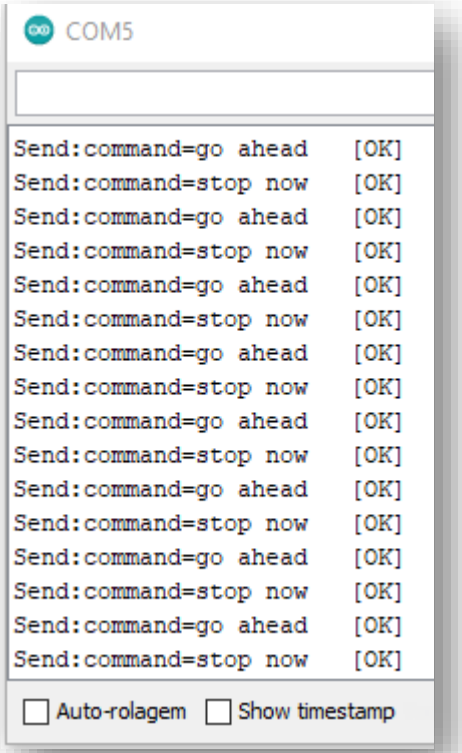
```
#include <VirtualWire.h>
void setup(){
    Serial.begin(9600);
    vw_set_rx_pin(11);
    vw_setup(2048);
    vw_rx_start();
}

void loop(){
    String m;
    uint8_t buf[VW_MAX_MESSAGE_LEN];
    uint8_t buflen = VW_MAX_MESSAGE_LEN;
    if (vw_get_message(buf, &buflen)){
      for (int i=0; i < buflen; i++){
        m=m+String((char)buf[i]);
    }
    Serial.println(m);
    }
}
```

veiculo | securityvanet

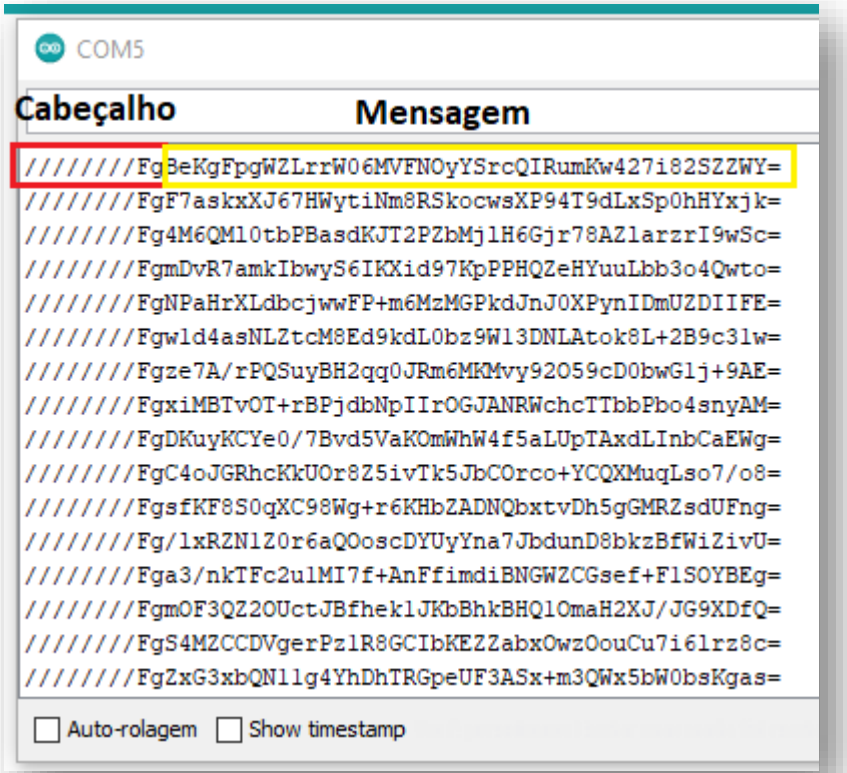
```
void setup() {
  Serial.begin(9600);
  enableRF(12,11);
  setKey("0123456789abcdef");
}

void loop() {
  if(receiver()){
    Serial.print("Rcpt:");
    Serial.print(getPlainTxt());
    action(getPlainTxt());
  }
}

void action(String act){
  if(act=="command=go ahead"){
    Serial.println("  (Start [OK])");
  }else if(act=="command=stop now"){
    Serial.println("  (Stop  [OK])");
  }else{
    Serial.println(" (CMD [ERROR])");
  }
}
```


João L. M. Freitas, Leonardo da R. Souza, Patrick R. Sardou, Nilson M. Lazarin
{joao.frhb,leorsouza15,ptrcksardou,nilsonmori}@gmail.com
CEFET/RJ

Experimentos



```
Send:command=go ahead    [OK]
Send:command=stop now    [OK]
Send:command=go ahead    [OK]
Send:command=stop now    [OK]
Send:command=go ahead    [OK]
Send:command=stop now    [OK]
Send:command=go ahead    [OK]
Send:command=stop now    [OK]
Send:command=go ahead    [OK]
Send:command=stop now    [OK]
Send:command=go ahead    [OK]
Send:command=stop now    [OK]
Send:command=go ahead    [OK]
Send:command=stop now    [OK]
Send:command=go ahead    [OK]
Send:command=stop now    [OK]
```



Cabeçalho **Mensagem**

```
////////FgBeKgFpgWZLrrW06MVFNOyYSrcQIRumKw427i82SZZWY=
////////FgF7askxXJ67HWytiNm8RSkocwsXP94T9dLxSp0hHYxjk=
////////Fg4M6QMl0tbPBasdKJT2PZbMjlH6Gjr78AZlarzrI9wSc=
////////FgmDvR7amkIbwyS6IKXid97KpPPHQZeHYuuLbb3o4Qwto=
////////FgNPaHrXLdbcjwwFP+m6MzMGPkdJnJ0XPynIDmUZDIIFE=
////////Fgwld4asNLZtcM8Ed9kdL0bz9Wl3DNLAtok8L+2B9c3lw=
////////Fgze7A/rPQSuyBH2qq0JRm6MKMvy92O59cD0bwGlj+9AE=
////////FgxiMBTvOT+rBPjdbNpIIrOGJANRWchcTTbbPbo4snyAM=
////////FgDKuyKCYe0/7Bvd5VaKOmWhW4f5aLUpTAxdLInbCaEWg=
////////FgC4oJGRhcKkUOr8Z5ivTk5JbCOrco+YCQXMuqLso7/o8=
////////FgsfKF8S0qXC98Wg+r6KHbZADNQbxtvDh5gGMRZsdUFng=
////////Fg/lxRZNlZ0r6aQOoscDYUyYna7JbdunD8bkzBfWiZivU=
////////Fga3/nkTFc2ulMI7f+AnFfimdiBNGWZCGsef+FlSOYBEg=
////////FgmOF3QZ2OUctJBfheklJKbBhkBHQlOmaH2XJ/JG9XDfQ=
////////FgS4MZCCDVgerPzlR8GCIbKEZZabxOwzOouCu7i6lrz8c=
////////FgZxG3xbQNllg4YhDhTRGpeUF3ASx+m3QWx5bW0bsKgas=
```

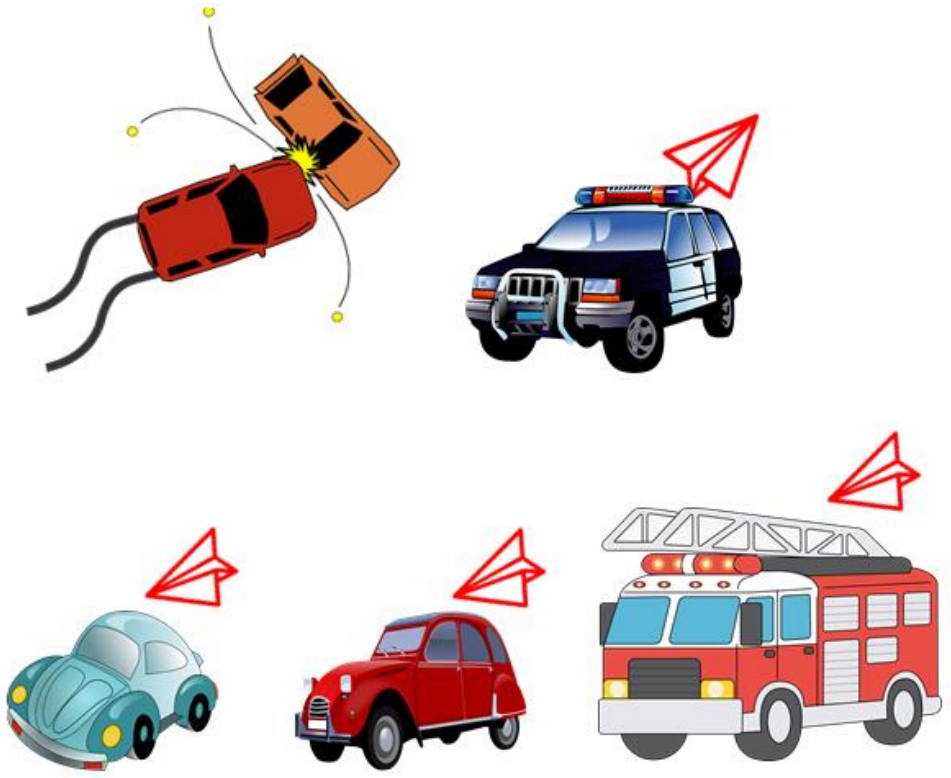
```
Rcpt:command=stop now  (Stop  [OK])
Rcpt:command=go ahead  (Start [OK])
Rcpt:command=stop now  (Stop  [OK])
Rcpt:command=go ahead  (Start [OK])
Rcpt:command=stop now  (Stop  [OK])
Rcpt:command=go ahead  (Start [OK])
Rcpt:command=stop now  (Stop  [OK])
Rcpt:command=go ahead  (Start [OK])
Rcpt:command=stop now  (Stop  [OK])
Rcpt:command=go ahead  (Start [OK])
Rcpt:command=stop now  (Stop  [OK])
Rcpt:command=go ahead  (Start [OK])
Rcpt:command=stop now  (Stop  [OK])
Rcpt:command=go ahead  (Start [OK])
Rcpt:command=stop now  (Stop  [OK])
Rcpt:command=go ahead  (Start [OK])
```

Realizado por SBC Sociedade Brasileira de Computação
ERRC 2021 online
19ª Escola Regional de Redes de Computadores
27 a 29 de outubro de 2021
INSTITUTO FEDERAL Sul-rio-grandense Câmpus Charqueadas
Comunicação segura em VANET
João L. M. Freitas, Leonardo da R. Souza, Patrick R. Sardou, Nilson M. Lazarin
{joao.frhb,leorsouza15,ptrcksardou,nilsonmori}@gmail.com
CEFET/RJ

Conclusão

Este trabalho apresentou uma implementação de biblioteca de comunicação para VANET, através da evolução da biblioteca apresentada por [Lazarin and Pantoja 2015], permitindo comunicação ***Broadcast*** via Rádio Frequência, através da integração com a biblioteca apresentada por [McCauley 2013].

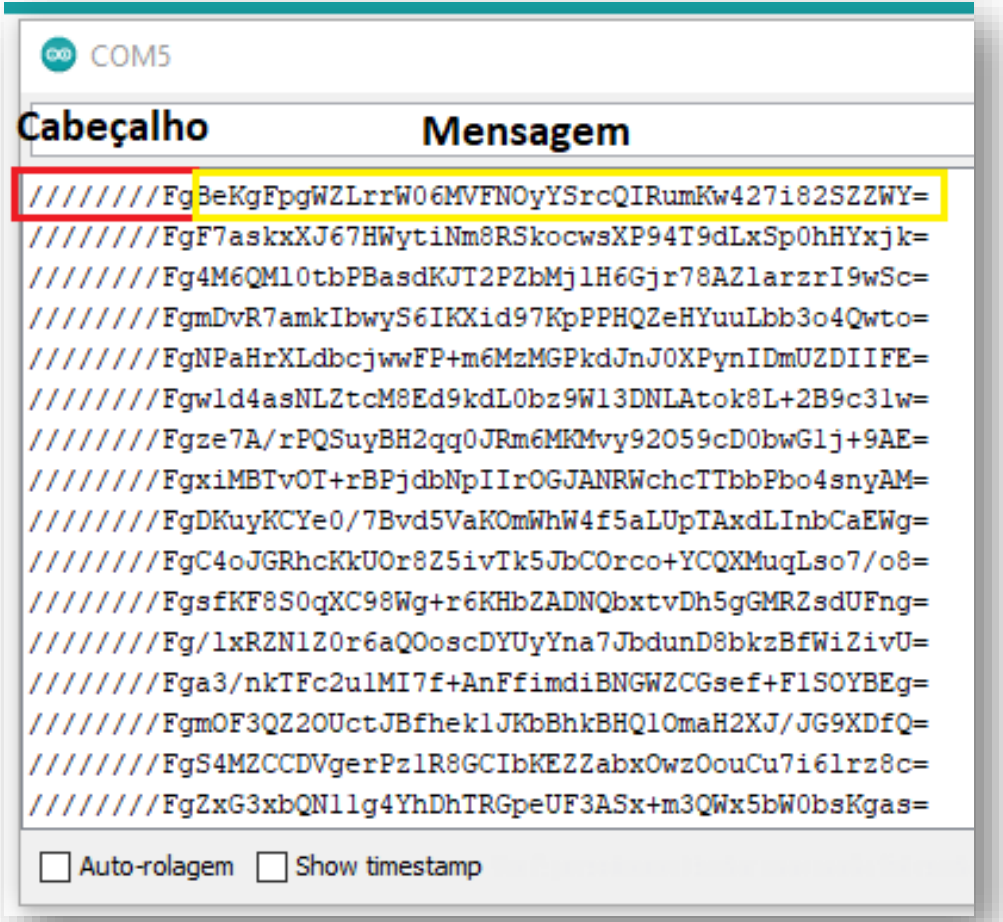
Além disso, foi possível garantir o sigilo da informação que transita no meio de difusão, através da integração com a biblioteca apresentada por [Rocha et al. 2020].



João L. M. Freitas, Leonardo da R. Souza, Patrick R. Sardou, Nilson M. Lazarin
{joao.frhb,leorsouza15,ptrcksardou,nilsonmori}@gmail.com

Conclusão

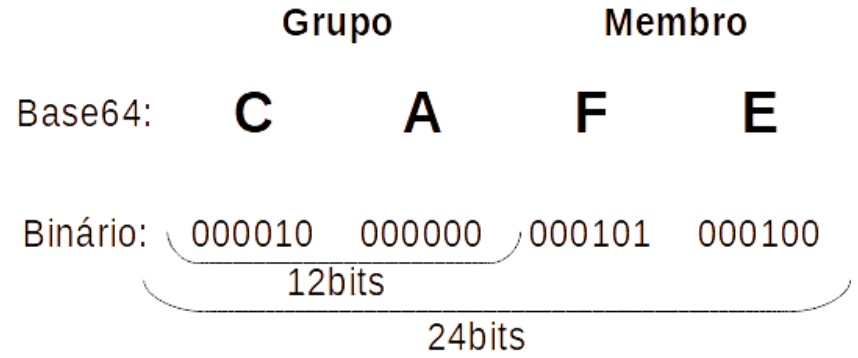
O modo de operação CBC com VI aleatório mostrou-se mais seguro, uma vez que a relação entre um mesmo comando enviado várias vezes, pelo membro ***Base***, equivale à diversos textos cifrados que trafegaram pelo meio de comunicação. Dessa forma, acaba por dificultar a análise do ***Espião***, na tentativa de identificação da relação do comando cifrado com a ação executada pelo ***Veículo***.



João L. M. Freitas, Leonardo da R. Souza, Patrick R. Sardou, Nilson M. Lazarin
{joao.frhb,leorsouza15,ptrcksardou,nilsonmori}@gmail.com
CEFET/RJ

Conclusão

Como trabalhos futuros pode-se considerar adoção completa do protocolo apresentado por [Lazarin et al. 2021], garantindo assim, o envio de mensagens ***unicast***, ***multicast*** e ***broadcast***, através do meio de difusão. Outra possibilidade de trabalho futuro é permitir o uso de tamanhos de texto e chave diferentes de 128bits, pois a biblioteca Securino implementa apenas o AES com chave de 128bits, bloco de texto fixo de 128bits e modos ECB e CBC com VI aleatório.



Destino	Origem	Tamanho	Mensagem
////	CAFE	Aw	Hello!
24 bits	24 bits	12 bits	up to 4095 bits

João L. M. Freitas, Leonardo da R. Souza, Patrick R. Sardou, Nilson M. Lazarin
{joao.frhb,leorsouza15,ptrcksardou,nilsonmori}@gmail.com
CEFET/RJ

Referências

DWORKIN, Morris. NIST Special Publication 800-38: Recommendation for Block Cipher Modes of Operation. **US National Institute of Standards and Technology**, 2001. Disponível em: http://csrc. nist. gov/publications/nistpubs/800-38a/sp800-38a.pdf.

KIM, David; SOLOMON, Michael G. **Fundamentos de segurança de sistemas de informação**. 1a ed. Rio de Janeiro: LTC, 2014.

LAZARIN, Nilson Mori; PANTOJA, Carlos Eduardo. A robotic-agent platform for embedding software agents using raspberry pi and arduino boards. **9th Software Agents, Environments and Applications School**, , p. 13–20, 2015.

LAZARIN, Nilson Mori; PANTOJA, Carlos Eduardo; JESUS, Vinicius Souza de. Um Protocolo para Comunicação entre Sistemas Multi-Agentes Embarcados. **15th Workshop-School on Agents, Environments, and Applications (WESAAC 2021)**, 2021.

MCCAULEY, Mike. Documentation for the VirtualWire communications library for Arduino. 2013. Disponível em: http://www.airspayce.com/mikem/arduino/VirtualWire.pdf.

ROCHA, Igor; SCHOTT, Richard; VERLY, Pedro; LAZARIN, Nilson. Análise de desempenho e conformidade em bibliotecas criptográficas para Internet das Coisas. *In*: ESCOLA REGIONAL DE SISTEMAS DE INFORMAÇÃO DO RIO DE JANEIRO (ERSI-RJ), 2019., event-place: Duque de Caxias. **Anais da VI Escola Regional de Sistemas de Informação do Rio de Janeiro** [...]. Duque de Caxias-RJ: SBC, 2019. Disponível em: https://sol.sbc.org.br/index.php/ersi-rj/article/view/10118.

SABAHI, Farzad. The Security of Vehicular Adhoc Networks. 2011. **2011 Third International Conference on Computational Intelligence, Communication Systems and Networks** [...]. [*S. l.*: *s. n.*], 2011. p. 338–342. https://doi.org/10.1109/CICSyN.2011.77.

SERAFIM, Vinícius da Silveira. Introdução à Criptografia: Cifras de Fluxo e Cifras de Bloco. 2012. Disponível em: http://www.serafim.eti.br/academia/recursos/Roteiro_05-Cifras_de_Fluxo_e_Bloco.pdf. Acesso em: 10 maio 2021.