UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E BIOLÓGICAS
DEPARTAMENTO DE COMPUTAÇÃO

LUCAS DE ARAÚJO

ESTUDO DAS VULNERABILIDADES PRESENTES NOS CARTÕES MIFARE CLASSIC

Ouro Preto, MG
2024

UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E BIOLÓGICAS
DEPARTAMENTO DE COMPUTAÇÃO

LUCAS DE ARAÚJO

ESTUDO DAS VULNERABILIDADES PRESENTES NOS CARTÕES MIFARE CLASSIC

Monografia apresentada ao Curso de Ciência da Computação da Universidade Federal de Ouro Preto como parte dos requisitos necessários para a obtenção do grau de Bacharel em Ciência da Computação.

Orientador: Eduardo José da Silva Luz

Ouro Preto, MG
2024

MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE OURO PRETO
REITORIA
INSTITUTO DE CIENCIAS EXATAS E BIOLOGICAS
DEPARTAMENTO DE COMPUTACAO

FOLHA DE APROVAÇÃO

Lucas de Araújo

Estudo das vulnerabilidades presentes nos cartões MIFARE Classic

Monografia apresentada ao Curso de Ciência da Computação da Universidade Federal
de Ouro Preto como requisito parcial para obtenção do título de Bacharel em Ciência da Computação

Aprovada em 11 de Outubro de 2024.

Membros da banca

Eduardo José da Silva Luz (Orientador) - Doutor - Universidade Federal de Ouro Preto
Augusto Ferreira Guilarducci (Examinador) - Bacharel - PPGCC/UFOP
Guilherme Augusto Lopes Silva (Examinador) - Mestre - Universidade Federal de Ouro Preto

Eduardo José da Silva Luz, Orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 11/10/2024.

Documento assinado eletronicamente por **Eduardo Jose da Silva Luz**, **PROFESSOR DE MAGISTERIO SUPERIOR**, em 15/10/2024, às 11:03, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0 , informando o código verificador **0789778** e o código CRC **BF497ACC**.

R. Diogo de Vasconcelos, 122, - Bairro Pilar Ouro Preto/MG, CEP 35402-163
Telefone: 3135591692 - www.ufop.br

Dedico este trabalho às duas pessoas que tornaram possível que meus pés caminhassem e minhas mãos aguentassem até o fim desta jornada.
Agradeço a Ana Claudia Venuto e a Luis Carlos Araújo.
Obrigado, mãe e pai.

Resumo

O trabalho aborda as vulnerabilidades presentes nos cartões MIFARE Classic, amplamente utilizados em sistemas de controle de acesso e transporte público. Com base em uma análise teórica e experimentos práticos, são descritos ataques como clonagem de UID, Nested, Darkside e Hardnested, explorando falhas no algoritmo de criptografia CRYPTO-1. A pesquisa demonstra a facilidade com que essas vulnerabilidades podem ser exploradas, alertando para a necessidade de medidas de segurança mais eficazes. Conclui-se que o uso contínuo desses cartões representa um risco significativo para a segurança dos sistemas que os utilizam.

Palavras-chave: Computação. Segurança. Radiofrequência. Programação. Criptografia. Arduíno.

Abstract

The paper addresses the vulnerabilities present in MIFARE Classic cards, widely used in access control and public transportation systems. Based on a theoretical analysis and practical experiments, attacks such as UID cloning, Nested, Darkside and Hardnested are described, exploiting flaws in the CRYPTO-1 encryption algorithm. The research demonstrates the ease with which these vulnerabilities can be exploited, alerting us to the need for more effective security measures. It concludes that the continued use of these cards poses a significant risk to the security of the systems that use them.

Keywords: Computing. Security. Radio frequency. Programming. Cryptography. Arduino.

Sumário

1 Introdução

Nos últimos anos, tecnologias de identificação automática vêm sendo empregadas cada vez mais em diferentes sistemas de diferentes segmentos da indústria, tais como logística, transporte e manufatura. Os procedimentos de identificação automática são empregados para prover informações essenciais, as quais permitem a adequada administração e controle do produto ou bem em análise (Finkenzeller 2010).

A tecnologia de identificação automática por radiofrequência (RFID - Radio Frequency Identification) teve seu desenvolvimento inicial na década de 40 e foi utilizada para diferenciar aeronaves aliadas de aeronaves inimigas durante a Segunda Guerra Mundial. Após o conflito, essa tecnologia evoluiu e foi aderida ao uso civil em aplicações específicas, como rastreamento de animais e sistemas de pedágio automotivo. Com os avanços tecnológicos no final da década de 1990, como a miniaturização de circuitos integrados, abriram-se inúmeras novas possibilidades de aplicação desta tecnologia, desde cartões inteligentes sem contato utilizados como bilhetes eletrônicos e etiquetas sensoriais capazes de monitorar a temperatura de alimentos frescos ou medicamentos, facilitando o controle de qualidade (Gampl et al. 2008).

Rádio Frequência (RFID - Radio Frequency Identification) é um método de identificação automática que opera por meio de comunicação sem fio, utilizando a proximidade entre uma tag e um leitor. Essa tecnologia é capaz de transmitir pequenas quantidades de dados a uma distância específica mediante a frequência que está sendo utilizada. Conforme descrito por (Finkenzeller 2010), as tags ativas possuem uma bateria integrada que fornece total ou parcialmente a energia necessária para o funcionamento do microchip. Em contrapartida, as tags passivas não têm fonte de energia própria, dependendo exclusivamente do campo eletromagnético gerado pelo leitor para todas as suas operações.

Os cartões de radiofrequência oferecem a vantagem de não necessitarem de contato direto com a leitora, graças à tecnologia RFID (Identificação por Radiofrequência), que permite a transmissão de dados sem contato físico. Segundo (Finkenzeller 2010), É preferível que a transferência de dados entre o identificador e o leitor ocorra sem contato físico, pois, em algumas aplicações, o contato mecânico pode ser impraticável. Adicionalmente, a ausência de contato físico pode prolongar a durabilidade tanto do leitor quanto do cartão identificador.

Dentre a enorme variedade de cartões disponíveis no mercado, a família MIFARE, um produto da NXP Semiconductors (anteriormente Philips), destaca-se. Ela contém quatro tipos diferentes de cartões: Ultralight, Standard, DESFire e SmartMX, sendo o MIFARE Classic o mais amplamente utilizado e o foco de estudo deste trabalho. O MIFARE Classic, disponível em três tamanhos de memória diferentes (320B, 1KB e 4KB), proporciona um sistema de autenticação mútua com o leitor e sigilo dos dados por meio da cifra de fluxo CRYPTO-1 (Garcia et al. 2009),

um algoritmo proprietário da NXP.

Embora os cartões MIFARE Classic ofereçam vantagens em termos de custo e facilidade de uso, persistem discussões relevantes sobre a segurança e a proteção dos dados armazenados neles (NOHL 2007). A popularidade desses cartões em sistemas críticos, como controle de acesso e transporte público, torna ainda mais urgente a análise das suas vulnerabilidades. Este trabalho, portanto, busca aprofundar o estudo sobre essas falhas de segurança, com o objetivo de identificar e compreender as brechas existentes. Os resultados preliminares obtidos indicam vulnerabilidades exploráveis, demonstrando a importância de desenvolver estratégias mais eficazes para proteger os sistemas que utilizam esses cartões contra potenciais ataques.

1.1 Justificativa

Introduzido no mercado em 1995 pela NXP, anteriormente conhecida como Philips, o MIFARE Classic rapidamente ganhou popularidade devido ao seu baixo custo e à facilidade de implementação. Sua aplicação tem sido ampla, abrangendo sistemas de controle de acesso, como em sistemas de transporte público e em prédios residenciais e comerciais. Até o final de 2008, estima-se que foram produzidos cerca de 3,5 bilhões desses cartões (Souza 2011). Contudo, a grande disseminação do MIFARE Classic levanta preocupações significativas, visto que esses cartões apresentam vulnerabilidades e estão expostos a riscos de fraude, conforme destacado em diversos estudos (NOHL 2007).

Apesar das revelações sobre graves falhas de segurança no MIFARE Classic (NOHL 2007), que comprovam o comprometimento de sua segurança, a indústria parece não demonstrar grande preocupação, continuando a utilizá-lo extensivamente, inclusive em novos projetos (Chiu et al. 2013). Provavelmente, essa persistência no uso deve-se aos investimentos já realizados e ao seu baixo custo.

A transição para um modelo de cartão mais seguro implicaria em custos adicionais na aquisição de novos equipamentos com tecnologias presumivelmente mais seguras. Contudo, o risco de fraudes envolvendo aplicações que usam o MIFARE Classic tende a crescer, especialmente considerando que o custo para realizar certos tipos de ataques é relativamente baixo, necessitando apenas de um leitor autêntico conectado a um computador, tal como um Arduino Uno em conjunto com um módulo MFRC522.

Ademais, na internet, encontram-se disponíveis ferramentas desenvolvidas na linguagem C, de código aberto. Estas podem ser utilizadas ou modificadas por atacantes para atender a propósitos específicos, como no caso da ferramenta MFCUK (Costin 2010).

1.2 Objetivos

Os principais objetivos visados por este trabalho são:

- Realizar uma revisão da literatura sobre as falhas de segurança identificadas no cartão MIFARE Classic;
- Descrever as características do cartão, incluindo sua estrutura interna, seu procolo de comunicação e seu algoritmo de criptografia, visando esclarecer a compreensão dos ataques;
- Destacar vulnerabilidades encontradas e apontar potenciais alvos para ataques;
- Detalhar quais são os possíveis tipos de ataques mais comuns e significativos que o Mifare Classic está vulnerável.
- Demonstrar o impacto de um ataque simples através de um cenário onde é explorado um conjunto de vulnerabilidades presentes na cifra Crypto-1.

2 Revisão Bibliográfica

O avanço tecnológico proveniente das últimas décadas trouxe consigo uma série de inovações que estão presentes em diversos setores do campo da segurança e da identificação eletrônica. Dentre as tecnologias que emergiram neste contexto, encontram-se presentes os cartões baseados em rádio-frequência, como o MIFARE Classic, que tornaram-se amplamente adotados em diversas aplicações, desde sistemas de transporte públicos até sistemas envolvendo o controle de acesso em áreas restritas.

No entanto, mesmo com o notável potencial de inovação e automação dos sistemas RFID, estes apresentam diversas vulnerabilidades. Segundo (18) sistemas estão expostos a uma vasta variedade de ataques maliciosos, desde simples escutas passivas até intervenções ativas.

2.1 Trabalhos Relacionados

A temática das vulnerabilidades em cartões MIFARE Classic tem sido objeto de estudo e análise por diversos pesquisadores ao longo dos anos, dada a sua relevância no campo da segurança e identificação eletrônica.

Em (NOHL 2007), os pesquisadores demonstraram as fraquezas do algoritmo de criptografia proprietário Crypto-1, utilizado nos cartões do tipo Mifare Classic. Para entender como funcionava o algoritmo, os pesquisadores realizaram o processo de engenharia reversa utilizando microscópio eletrônico no chip do cartão. Após analisado todo o processo de funcionamento do algoritmo de criptografia supracitado, os autores demonstraram que o devido a uma fraqueza envolvendo o processo de geração de números pseudo-aleatórios, utilizado durante o processo de autenticação entre o cartão e a leitora, é possível obter todas as chaves de criptografia que estão presentes no cartão, desde de que o atacante possua pelo menos uma chave de algum setor arbitrário. Este ataque ficou conhecido popularmente mais tarde como *Nested Attack*.

Em (10) foi explorado um ataque alternativo onde não se faz necessário a presença de uma chave criptográfica prévia para que se obtenha as demais chaves. os autores descrevem toda arquitetura e o protocolo de comunicação entre o cartão e a leitora e demonstram como funciona o novo ataque proposto. O ataque demonstrado pelo grupo de pesquisadores também se aproveita de uma fraqueza presente no algoritmo de geração de números pseudo-aleatórios da cifra Crypto-1, porém, diferente da pesquisa supracitada, neste estudo é explorada a capacidade de utilizar a própria cifra de fluxo do algoritmo para ler todos os blocos de memória do chip do cartão e obter pelo menos uma chave criptográfica de algum setor arbitrário de memória do cartão.

Em (Courtois 2009), foi realizado em paralelo ao estudo de Gans et al. e este estudo

propõe uma segunda abordagem, menos custosa, para que se obtenha, também, pelo menos uma chave de algum setor arbitrário de memória do cartão. A pesquisa deste autor diferencia-se da pesquisa anterior por utilizar técnicas criptoanalíticas, em particular, a criptoanálise algébrica, para analisar e explorar o algoritmo Crypto-1.

Neste contexto, (Courtois 2009). evidencia que sua metodologia é mais eficiente em termos computacionais, especialmente devido à ausência de um pré-processamento de dados, um passo presente em outras pesquisas. É importante salientar que, neste estudo, (Courtois 2009) enfatiza que o grande desafio não é apenas identificar as vulnerabilidades nos cartões, mas também o fato de muitas instituições dependerem exclusivamente de um cartão de rádio-frequência como único método de acesso a informações sensíveis, colocando em risco toda a sua estratégia operacional.

Em (**?**) os pesquisadores analisam as fragilidades dos sistemas eletrônicos empregados nas infraestruturas dos serviços de transporte público e privado em Brasília. A pesquisa revelou que tais sistemas adotam o cartão Mifare Classic. O objetivo do estudo é produzir conhecimento para apoiar o trabalho de analistas de segurança, propor medidas para remediar as vulnerabilidades encontradas e contribuir para a segurança do ATS (Automatic Ticketing System) e dos usuários finais do sistema de transporte público brasileiro.

Durante a pesquisa, os autores elucidam as formas de explorar as vulnerabilidades já reconhecidas no cartão Mifare Classic para burlar as normativas do sistema de transporte. Como exemplo, demonstram procedimentos para clonar o cartão e alterar seu saldo de créditos. Concluindo, após expor as diversas questões associadas ao uso do referido cartão, o estudo recomenda a remoção iminente dos cartões Mifare Classic dos sistemas de transporte, tanto públicos quanto privados.

2.2 Fundamentação Teórica

Antes de adentrarmos a discussão mais profunda sobre as vulnerabilidades nos cartões MIFARE Classic e os estudos realizados por diversos pesquisadores, é crucial entendermos alguns conceitos fundamentais que norteiam essa área de estudo.

2.2.1 Fundamentos de Criptografia

Este tópico apresenta uma introdução aos conceitos de criptografia que serão necessários para o entendimento de assuntos posteriormente abordados em outros capítulos. A criptografia é a espinha dorsal da segurança da informação, fornecendo os meios para proteger dados contra acessos não autorizados durante a transmissão ou enquanto estão armazenados. Os conceitos aqui discutidos são fundamentais para compreender as vulnerabilidades e as técnicas de proteção associadas aos cartões MIFARE Classic.

2.2.1.1 Introdução à Criptografia

A comunicação segura começa com uma mensagem simples, conhecida como texto claro. A arte de camuflar essa mensagem, escondendo seu verdadeiro significado, é denominada criptografia, transformando-a em um texto cifrado. O processo inverso, que retorna o texto cifrado ao seu formato original, é a descriptografia (Bruce 1996).

Criptografia é, portanto, tanto uma ciência quanto uma arte dedicada à proteção de mensagens, uma especialidade dos criptógrafos. Paralelamente, a criptoanálise ocupa-se de quebrar estas cifras, um desafio assumido pelos criptoanalistas que buscam desvendar os segredos por trás do texto cifrado. Juntas, criptografia e criptoanálise formam a criptologia, uma disciplina que se aprofunda nos fundamentos matemáticos e teóricos para desenvolver códigos e compreender suas vulnerabilidades (Bruce 1996).

Segundo De Souza (2011, p. X):

> "Um algoritmo criptográfico pode ser visto como uma função f que, usando uma chave K, recebe de entrada uma mensagem x, sendo capaz de produzir um texto criptografado y que somente pode ser decriptografado aplicando a inversa da função f utilizando a mesma chave K e o valor y como entrada. Por exemplo, suponha que Alice precise enviar uma mensagem confidencial m para Beto. Então Alice calcula $y = f_K(x)$ e envia para Beto. Tanto Alice como Beto conhecem a chave secreta K. Então Beto calcula $x = f_K^{-1}(y)$ e obtém o texto em claro. Este tipo de criptografia é conhecido como Criptografia de Chave Simétrica."

A moderna criptologia não se restringe apenas a técnicas e práticas; é um campo de estudo avançado que exige dos seus praticantes um sólido conhecimento em matemática teórica. Isso é essencial não apenas para criar sistemas de criptografia robustos, mas também para entender e superar os métodos utilizados na criptoanálise.

2.2.1.2 Criptografia de Chave Simétrica

A criptografia de chave simétrica, também conhecida como criptografia simétrica, funciona utilizando a mesma chave secreta compartilhada para cifrar e decifrar o texto transmitido. Este método de criptografia é um dos mais antigos e mais simples em termos de compreensão e implementação. A chave deve ser conhecida por ambas as partes envolvidas na comunicação, tornando a gestão segura das chaves uma parte crítica do processo de criptografia simétrica (Bruce 1996). A Figura 2.1 ilustra, de maneira simplificada, os processos desta cifra.

Os algoritmos de criptografia simétrica podem ser divididos em duas categorias principais: cifras de bloco e cifras de fluxo. As cifras de bloco operam em blocos de dados de tamanho fixo, transformando-os em um bloco de texto cifrado de tamanho igual usando a chave secreta. Exemplos populares de cifras de bloco incluem AES (Advanced Encryption Standard) e DES

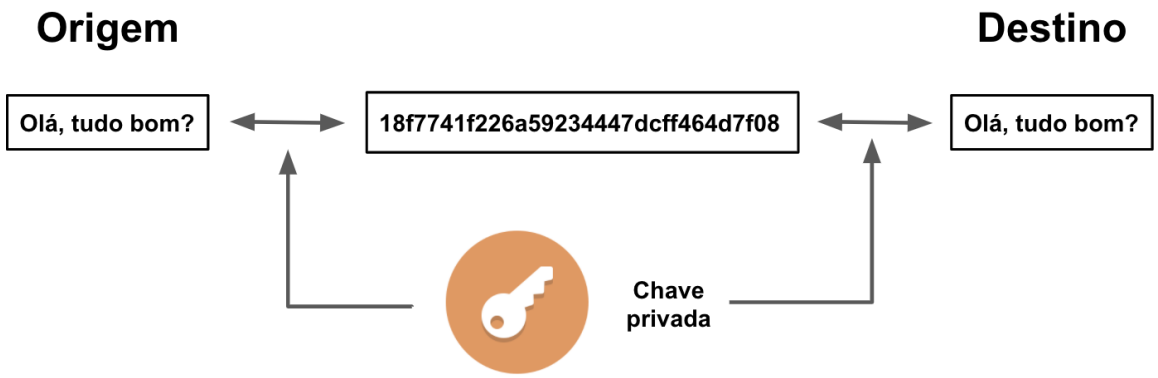

Figura 2.1 – Criptografia Simétrica

(Data Encryption Standard). Por outro lado, as cifras de fluxo criptografam os dados bit a bit (ou byte a byte), o que as torna mais adequadas para aplicações de streaming ou em situações onde o tamanho dos dados é desconhecido ou variável. Um exemplo de cifra de fluxo é o RC4 (Bruce 1996).

A principal vantagem da criptografia simétrica é a eficiência: ela geralmente requer menos recursos computacionais do que a criptografia de chave assimétrica, o que a torna mais rápida para cifrar e decifrar grandes volumes de dados. Isso a torna ideal para uso em sistemas onde o desempenho é crítico, como criptografar dados em disco ou a transmissão de dados em redes de alta velocidade (Stallings 1995)

No entanto, a gestão de chaves apresenta desafios significativos, especialmente em sistemas de comunicação de larga escala. Cada par de comunicantes precisa compartilhar uma chave secreta única, que deve ser trocada por um meio seguro antes da comunicação criptografada. Isso escala mal para grandes redes, uma vez que o número de chaves necessárias cresce exponencialmente com o número de participantes. Além disso, se a chave for comprometida, a confidencialidade de todas as mensagens cifradas com essa chave também será comprometida.

2.2.1.2.1 Cifra de Fluxo

Como citado previamente, as cifras de fluxo operam bit a bit (ou byte a byte) durante o processo de criptografia dos dados. As cifras de fluxo são consideradas mais adequadas para a implementação em hardware quando comparadas às cifras de bloco, pois demandam um menor custo computacional dos circuitos do hardware envolvido (DE SOUZA, 2011). Uma cifra de bloco pode emular o comportamento de uma cifra de fluxo caso esteja sendo utilizada no modo

CFB (*Cipher Feedback*) ou OFB (*Output Feedback*), porém as cifras de fluxo, por natureza, são projetadas de maneira a serem mais rápidas (Souza 2011).

A chave que será utilizada durante o processo de criptografia é carregada em um gerador que utiliza a mesma para produzir uma sequência de bits de natureza aparentemente aleatória. O resultado deste gerador é chamado de *keystream*. Após gerado o *keystream*, o mesmo é combinado bit a bit com o fluxo do texto plano (mensagem original) através de operações de XOR (OU-exclusivo) e, por fim, será obtido o texto criptografado. Para realizar o processo de descriptografia, basta alimentar o gerador novamente com a mesma chave, obter o *keystream* novamente e realizar o processo de XOR com o texto cifrado. Dessa maneira, será obtido o texto plano original. A Figura 2.2 ilustra o funcionamento típico de uma cifra de fluxo.

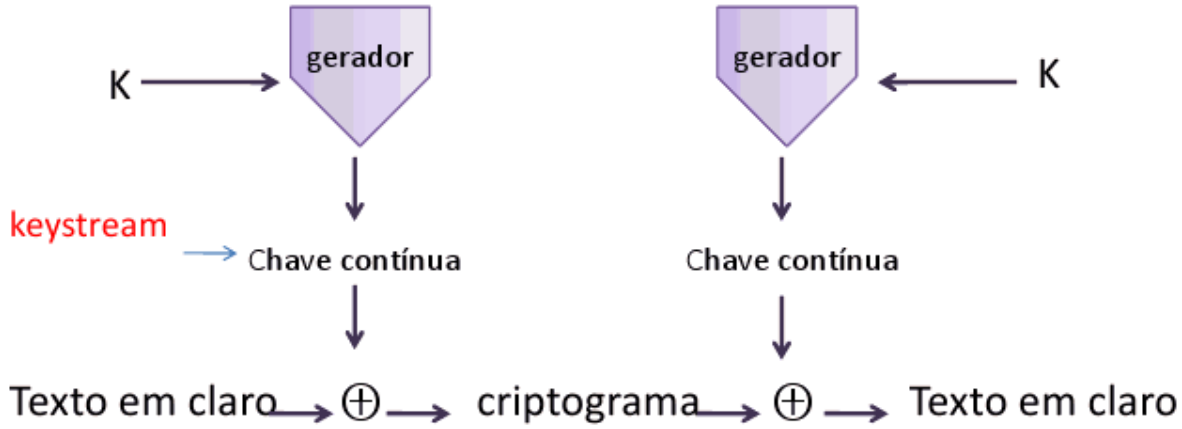

Figura 2.2 – Cifra de Fluxo

Existem diversas maneiras de se implementar um algoritmo responsável pela geração do *keystream*. A qualidade do *keystream* é crucial para a segurança do sistema de criptografia, pois qualquer previsibilidade ou repetição na sequência pode ser explorada por atacantes para quebrar a cifra. Aqui estão algumas das técnicas mais comuns para gerar *keystreams*:

1. **Geradores de Congruência Linear (LCGs)**: Os LCGs são um tipo simples de gerador de números pseudoaleatórios que utiliza uma relação linear para produzir a próxima saída a partir da anterior (16)

2. **Registradores de Deslocamento com Retroalimentação Linear (LFSRs)**: Os LFSRs são estruturas baseadas em registradores de deslocamento que geram sequências pseudoaleatórias. Eles são mais complexos e seguros que os LCGs, especialmente quando configurados corretamente. Os LFSRs podem produzir sequências com boas propriedades estatísticas, mas sua segurança depende do tamanho do registrador e da escolha das taps para feedback (16)

3. **Funções de Hash Criptográficas**: Algumas cifras de fluxo usam funções de hash criptográficas para gerar *keystreams*. Ao alimentar a função hash com uma semente e, potencialmente, um contador, é possível gerar um *keystream* muito difícil de prever. Este método é conhecido por sua robustez e resistência a ataques, pois as propriedades das funções hash criptográficas garantem a imprevisibilidade do *keystream* (16).

4. **Geradores de Números Pseudoaleatórios Criptograficamente Seguros (CSPRNGs)**: Os CSPRNGs são projetados para atender aos requisitos de segurança da criptografia, produzindo sequências de números que são, na prática, indistinguíveis de sequências verdadeiramente aleatórias. Eles são baseados em algoritmos complexos que incluem entradas de entropia do ambiente para garantir a aleatoriedade (16).

No decorrer deste trabalho, será discutido a fundo apenas o **Registrador de Deslocamento com Retroalimentação Linear (LFSRs)**, devido ao fato de o mesmo ser utilizado na cifra de fluxo empregada nos cartões MIFARE Classic.

2.2.1.2.2 Registrador de Deslocamento com Retroalimentação Linear (LFSRs)

O LFSR é um registrador cujo bit de entrada (ou retroalimentação) é uma função linear do seu próprio estado anterior. O estado inicial do Registrador de Deslocamento com Feedback Linear (LFSR) é referido como a chave. Devido à natureza determinística do LFSR, a sequência de bits que ele produz é inteiramente definida pelo seu estado inicial ou atual. Entretanto, dada a limitação no número de estados que o LFSR pode assumir, ele eventualmente entra em um ciclo de repetição.

Os LFSRs são vantajosos para implementação em hardware, devido ao seu custo relativamente baixo, tornando-os ideais para aplicações que demandam uma rápida geração de bits, como é o caso das cifras de fluxo. Porém, como mencionado por Menezes, Van Oorschot e Vanstone (2018), as sequências de saída dos LFSRs são também legíveis em análises utilizando técnicas algébricas, tornando-as facilmente previsíveis e permitindo a violação do algoritmo.

2.2.1.3 Estrutura Lógica do Cartão MIFARE Classic

O cartão MIFARE Classic é classificado como um cartão de memória *Electrically-Erasable Programmable Read-Only Memory* (EEPROM), com algumas funcionalidades de proteção implementadas pela própria fabricante. A unidade básica de memória do cartão consiste em um bloco de 16 bytes, onde, na versão de 1K, cada um dos blocos é agrupado em um setor, totalizando 16 setores. A tabela 2.1 ilustra esta estrutura supracitada:

Número do Setor	**Número do Bloco**	**Conteúdo (16 bytes)**
00	00	BCC, UID, Fabricante (somente Leitura)
	01. Dados/Valor	Dados ou Valor
	02. Dados/Valor	Dados ou Valor
	03. Trailer	Chave A | Condições de Acesso | U | Chave B
01	04. Dados/Valor	Dados ou Valor
	05. Dados/Valor	Dados ou Valor
	06. Dados/Valor	Dados ou Valor
	07. Trailer	Chave A | Condições de Acesso | U | Chave B
...		...
15	60. Dados/Valor	Valor | Valor | Valor | 00 FF 00 FF
	61. Dados/Valor	Valor | Valor | Valor | 00 FF 00 FF
	62. Dados/Valor	Dados ou Valor
	63. Trailer	Chave A | Condições de Acesso | U | Chave B

Tabela 2.1 – Estrutura do Cartão (Adaptado de (Souza 2011))

2.2.1.3.1 Bloco do Fabricante

Este bloco é designado para armazenar a identificação única (UID) do cartão, funcionando como uma marcação distinta atribuída a cada cartão produzido pela fábrica. Essa marcação consiste em um conjunto de 4 bytes.

Adjacente a esse conjunto de bytes, localiza-se o BCC (Byte de Verificação de Controle), um byte encarregado de assegurar a integridade do UID, com o objetivo de confirmar que não houve qualquer forma de adulteração. A verificação é efetuada através de operações de XOR (Operação Lógica "ou exclusivo").

Subsequentemente ao BCC, os dados remanescentes neste bloco referem-se ao fabricante, englobando informações como o modelo do cartão e o nome do fabricante, entre outros aspectos relevantes. É pertinente salientar que este bloco é configurado unicamente para operações de leitura, impedindo, assim, durante o processo de comunicação com o dispositivo leitor, a realização de qualquer alteração nos dados contidos neste bloco. A tabela 2.2 ilustra o fato:

UID	BCC	Dados do Fabricante
0	4	5 - 15

Tabela 2.2 – Bloco do Fabricante (Adaptado de (Souza 2011))

2.2.1.3.2 Bloco de Dados

O armazenamento de dados é feito nos três primeiros blocos de cada setor do cartão, exceto no primeiro por se tratar do bloco do fabricante, como supracitado. Os dados armazenados neste setor são, de forma geral, informações relevantes para a lógica de negócios na qual o cartão está inserido. O cartão MIFARE Classic possui um formato específico para trabalhar com os

valores armazenados no mesmo. Este formato, conhecido como *Value Block* (Bloco de Valor), tem como propósito permitir a detecção e correção de erros.

No início do bloco de dados, os primeiros 12 bytes são dedicados ao armazenamento de um valor inteiro de 4 bytes. Esse procedimento de armazenamento é realizado três vezes, seguindo um padrão específico: inicialmente, o valor é armazenado de forma convencional entre os bytes 0 a 3 e, posteriormente, entre os bytes 8 a 11. Em um processo distinto, entre os bytes 4 a 7, o mesmo valor é registrado após a inversão de seus bits. Essa técnica de redundância e inversão de bits é adotada como uma medida de segurança para verificação e integridade dos dados.

Nos segmentos finais do bloco, especificamente nos últimos 4 bytes, é empregado um método de armazenamento para o endereçamento do bloco, ocupando 1 byte. Esse valor de endereçamento é replicado quatro vezes dentro deste espaço, alternando entre o formato original e um formato onde os bits são invertidos. Tal estratégia é adotada para assegurar a consistência e a recuperação do endereço do bloco, mesmo em casos de falhas parciais dos dados. A tabela 2.3 ilustra o fato supracitado:

Valor	Valor	Valor	End	End	End	End
0	4	8	12	13	14	15

Tabela 2.3 – Bloco de Valor (Adaptado de (Souza 2011))

2.2.1.3.3 Bloco *Trailer*

Último bloco de cada setor, o bloco *trailer* é o bloco responsável por armazenar as chaves utilizadas durante o processo de autenticação. Antes de realizar qualquer operação na memória do cartão (leitura, escrita ou deleção), a leitora irá acessar este bloco para garantir que o processo de autenticação aconteça.

Neste bloco estão presentes duas chaves, sendo cada uma com tamanho de 6 bytes. A primeira chave, chave A, é obrigatória e não é passível de leitura. Em seguida, no setor, são definidos os bits referentes às **Condições de Acesso** (CA). Estas condições de acesso serão explicadas com maiores detalhes posteriormente.

Prosseguindo, mediante a como estão definidas as condições de acesso, pode haver uma chave B no setor, sendo esta passível ou não de leitura. Caso a mesma não esteja definida, o espaço dela pode ser utilizado para armazenar outras informações que sejam relevantes para o contexto do uso do cartão. Por fim, existe um byte restante denominado de **U** que pode ser utilizado com o intuito de armazenar dados. A tabela 2.4 ilustra esta descrição supracitada.

Chave A	CA	U	Chave B (opcional)	
0	6	9	10	15

Tabela 2.4 – Bloco *Trailer* (Adaptado de (Souza 2011))

Condições de Acesso Para cada bloco de dados e para o bloco *trailer* de cada setor, são definidas condições de acesso através de 3 bits armazenados de forma normal e de forma com os bits invertidos. Os bits controlam as permissões de acesso à memória utilizando a chave A e/ou a chave B.

Para cada acesso na memória do cartão, é realizada uma verificação destas condições de acesso. Caso aconteça uma violação como, por exemplo, uma violação no formato do bloco, o mesmo será bloqueado de maneira irreversível. A figura 2.3 ilustra a arquitetura básica do funcionamento destes bits de acesso:

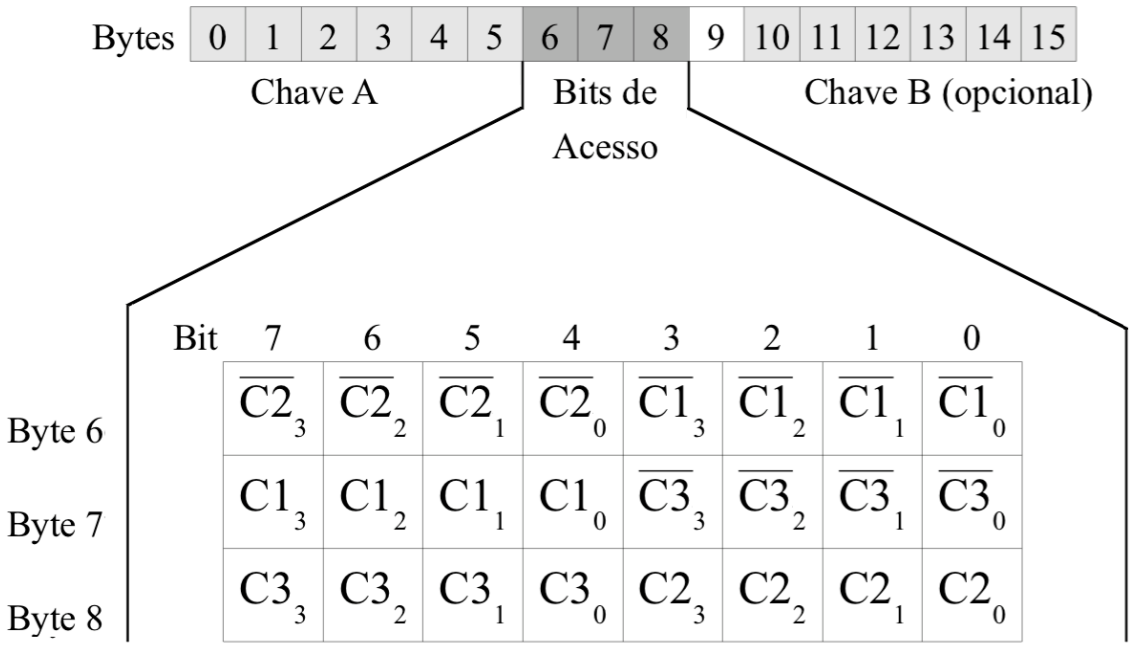

Figura 2.3 – Arquitetura das Condições de Acesso (Adaptado de (Souza 2011))

As regras que regem o acesso ao bloco trailer de um setor, que representa o último bloco de cada setor, são definidas com base nos bits de acesso e determinam as permissões de leitura e escrita. As permissões são categorizadas de acordo com os seguintes critérios: acesso não permitido ("nunca"), acesso permitido somente com a utilização da chave A ("chave A"), acesso permitido somente com a utilização da chave B ("chave B"), ou acesso permitido com o uso de qualquer uma das chaves A ou B ("chave A|B").

Em relação aos cartões recém-emissionados, as configurações padrão do bloco trailer do setor são estabelecidas para possibilitar o transporte, e a chave A é pré-configurada para este fim. É importante salientar que, nesse estágio inicial, a chave B pode ser acessada caso o sistema esteja configurado para o modo de transporte. Portanto, a autenticação de novos cartões deve ser realizada utilizando-se a chave A para garantir a segurança do sistema.

Doravante, temos a definição das permissões de acesso aos blocos de dados. Tais permissões são estipuladas com base nos bits de acesso e podem ser classificadas nas seguintes categorias: proibição total de acesso ("nunca"), acesso exclusivo mediante a chave A ("chave A"), acesso exclusivo mediante a chave B ("chave B") e acesso permissível com qualquer uma das chaves A ou B ("chave A|B"). A tabela 2.5 a seguir ilustra as condições de acesso para o bloco de dados do setor.

C1	C2	C3	Leitura	Escrita	Incremento	Decremento Transferência Restauração	Aplicação
0	0	0	chave A|B	chave A|B	chave A|B	chave A|B	config. de transporte
0	0	1	chave A|B	nunca	nunca	nunca	bloco de leitura/escrita
0	1	0	chave A|B	B	nunca	nunca	bloco de leitura/escrita
0	1	1	chave A|B	B	chave A|B	chave A|B	bloco de valor
1	0	0	chave A|B	nunca	chave A|B	chave A|B	bloco de valor
1	0	1	B	nunca	nunca	nunca	bloco de leitura/escrita
1	1	0	nunca	nunca	nunca	nunca	bloco de leitura/escrita
1	1	1	nunca	nunca	nunca	nunca	bloco de leitura/escrita

Tabela 2.5 – Condições de Acesso - Bloco de Dados (Adaptado de (Souza 2011))

2.2.1.4 Comunicação

Esta seção é dedicada a explicar o funcionamento do processo de comunicação entre a leitora e uma tag do tipo MIFARE Classic. Serão demonstrados o processo de autenticação e autorização, bem como o uso das chaves (A e B) dos setores, mencionadas anteriormente, durante esses processos. Posteriormente, também nesta seção, será explicado o funcionamento da cifra proprietária Crypto-1, utilizada pelo cartão MIFARE Classic.

1. **Primeiro Contato:** O processo de comunicação entre a leitora e a tag inicia-se assim que esta é aproximada da leitora. Após isso, o chip da tag é inicializado devido à descarga elétrica proveniente da alteração do campo magnético, e uma sequência de ações é iniciada.

2. **Ativação do Campo:** Momento onde o cartão encontra-se presente em um determinado raio que seja suficiente para que a leitora possa detectá-lo. Os tipos de requisições enviadas pela leitora são do tipo A ou B.

3. **Resposta do Cartão:** Assim que o cartão recebe a requisição, ele responde à leitora para informar sobre qual tipo de cartão ele é e permitindo que a leitora, após receber esta informação, prossiga de maneira adequada com o processo de comunicação.

4. **Anti-Colisão:** Após receber a resposta do cartão, a leitora envia mais algumas requisições para detectar se algum outro cartão encontra-se presente no raio de distância, pois, caso o mesmo ocorra, isto pode acarretar em interferências na comunicação com o cartão desejado. Caso apenas um cartão responda os sinais, a leitora então selecionará este cartão e dará prosseguimento com o processo de comunicação.

5. **Autenticação:** Antes de realizar a troca de informações com o cartão, a leitora necessita autenticar-se utilizando a chave (A ou B) apropriada para a operação desejada. A leitora e o cartão realizam um processo conhecido como *challenge-response handshake* através do mecanismo de cifra proprietária Crypto-1. Este passo é responsável por garantir que somente leitoras autorizadas, ou seja, aquelas que possuem as chaves corretas, possam acessar as informações presentes no cartão.

6. **Troca de Informações:** Após a autenticação bem-sucedida, a leitora e o cartão passarão a trocar informações de maneira criptografada utilizando as chaves que foram solicitadas durante o passo anterior.

7. **Finalização:** Por fim, assim que a leitora finalizar as operações desejadas, a mesma irá enviar um comando chamado de *Halt* para que o cartão seja colocado em um estado de "cochilo"e não responda mais os comandos da leitora até que seja aproximado novamente e todo processo se repita.

A imagem 2.4 ilustra todo o processo supracitado:

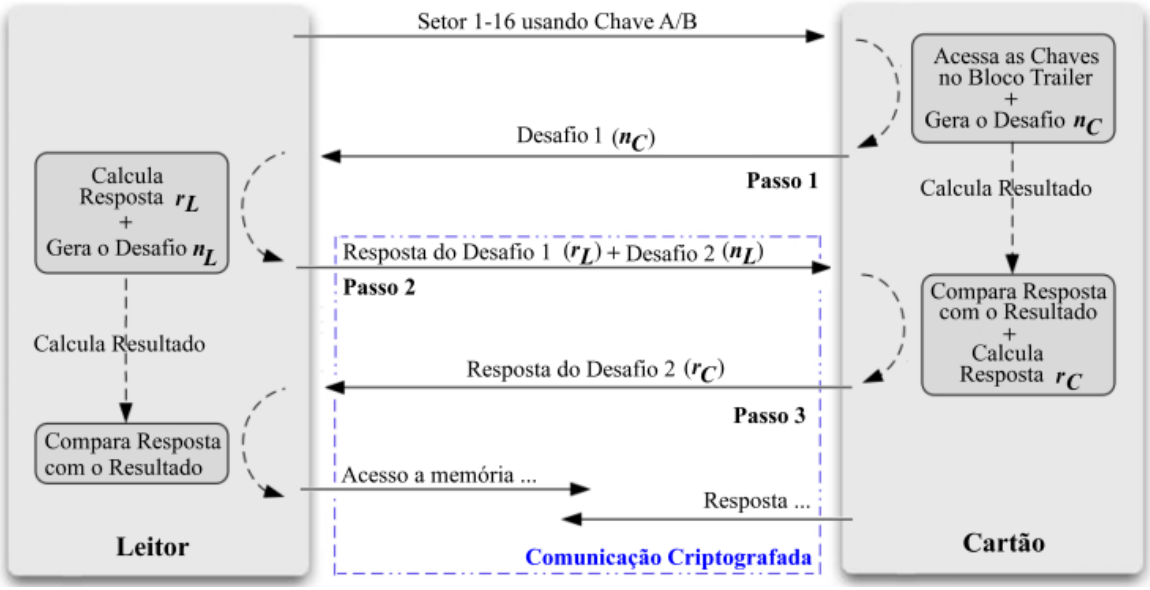

Figura 2.4 – Comunicação entre cartão e leitora

(Adaptado de (Souza 2011))

2.2.1.4.1 Crypto-1

Desenvolvida pela NXP durante a fase de concebimento do cartão MIFARE Classic, o Crypto-1 é o algoritmo proprietário que tinha como objetivo garantir a segurança entre o cartão e a leitora durante a transmissão das informações. O funcionamento do algoritmo foi mantido em segredo de indústria por muitos anos, técnica esta conhecida como "segurança por

obscuridade"para garantir que pessoas indesejadas não fossem capazes de estudar descobrir vulnerabilidades presentes na cifra.

O algoritmo Crypto-1 é uma cifra de fluxo essencialmente. Seu principal componente é o gerador de bits pseudoaleatórios que produz, como resultado, os bits do *keystream* que será utilizado para cifrar e decifrar as mensagens. Este gerador consiste basicamente em um *linear feedback shift-register* (LFSR) de 48 bits que segue a estrutura do polinômio 2.1:

$$x^{48}+x^{43}+x^{39}+x^{38}+x^{36}+x^{34}+x^{33}+x^{31}+x^{29}+x^{24}+x^{23}+x^{21}+x^{19}+x^{13}+x^{9}+x^{7}+x^{6}+x^{5}+1 \tag{2.1}$$

Junto ao referido LFSR, existem funções auxiliares que realizam operações com determinados bits específicos dentre os 48 bits mencionados visando aumentar a natureza de complexidade do algoritmo e garantir um grau maior de imprevisibilidade. Estas funções auxiliares podem ser descritas através das fórmulas 2.2, 2.3 e 2.4:

$$f_a(a,b,c,d) := ((a \vee b) \oplus (a \wedge d)) \oplus (c \wedge ((a \oplus b) \vee d)) \tag{2.2}$$

$$f_b(a,b,c,d) := ((a \wedge b) \vee c) \oplus ((a \oplus b) \wedge (c \vee d)) \tag{2.3}$$

$$f_c(a,b,c,d,e) := (a \vee ((b \vee e) \wedge (d \oplus e))) \oplus ((a \oplus (b \wedge d)) \wedge ((c \oplus d) \vee (b \wedge e))) \tag{2.4}$$

Reunindo todos estes componentes mencionados, o LFSR final do algoritmo Crypto-1 pode ser representado pelo seguinte diagrama da figura 2.5:

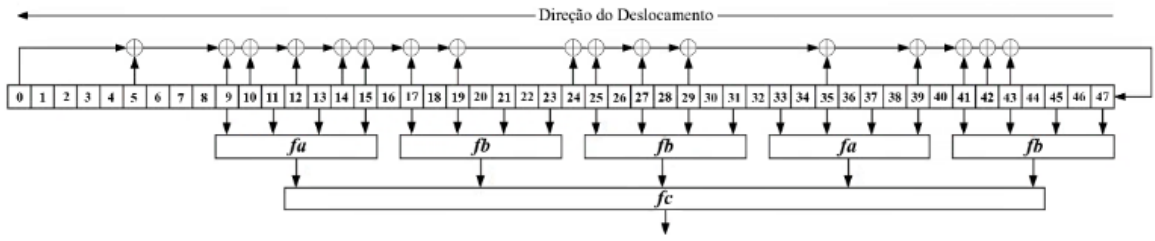

Figura 2.5 – LFSR e funções auxiliares do Crypto-1 (Adaptado de (Souza 2011))

Geração do *Keystream* Conhecida como *Keystream*, trata-se de uma sequência de bits utilizada para cifrar e decifrar informações. O *Keystream* possui um tamanho fixo, e cada bit de sua sequência é gerado após uma quantidade de *clocks* (operações) realizadas pelo gerador, que, neste caso, é o LFSR e suas funções auxiliares. No cenário da cifra Crypto-1, ocorre a seguinte sequência de instruções: a cada ciclo de *clock*, o LFSR produz exatamente um bit do *keystream*. Em seguida, o LFSR desloca-se um bit para a esquerda e insere um novo bit à direita.

Gerador Pseudoaleatório O gerador pseudoaleatório é o responsável pela criação dos desafios (*nonce*) que são utilizados durante o processo de autenticação entre o cartão e a leitora, fazendo parte do processo de *challenge-response handshake*.

Este gerador consiste em um segundo LFSR que trabalha com 32 bits e que tem como estado inicial sempre a mesma sequência de bits ilustrada pela figura 2.5.

1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0

(2.5)

Este mesmo LFSR utiliza, como gerador para o novo bit que será inserido à direita, a fórmula presente na figura (2.6):

$$L_{16}(x_0 x_1 \dots x_{15}) := x_0 \oplus x_2 \oplus x_3 \oplus x_5. \tag{2.6}$$

É importante notar que, embora o estado inicial contenha 32 bits, o gerador realiza operações apenas com 16 bits. Esse fato é relevante, pois facilita a exploração de vulnerabilidades relacionadas ao grau de previsibilidade dos desafios gerados.

2.2.2 Conceitos Adicionais

Radiofrequência (RF): É a faixa de frequência que compreende de 20 kHz a 300 GHz e é utilizada para a transmissão de ondas de rádio Essa tecnologia é fundamental para diversas aplicações, como comunicação sem fio, transmissão de TV e rádio, e sistemas RFID (Identificação por Radiofrequência). O principal objetivo da radiofrequência é possibilitar a comunicação e a transmissão de dados à distância, sem a necessidade de fios ou cabos, garantindo eficácia e amplitude de sinal (Coleman 2004).

RFID (Identificação por Radiofrequência): A tecnologia de identificação por radiofrequência (RFID) é um método de transferência de dados sem fios entre uma leitora e uma etiqueta que está ligada a qualquer objeto que seja necessário identificar. O principal objetivo do RFID é proporcionar identificação automática, rápida e sem necessidade de contato direto, sendo amplamente utilizado em controle de acesso, logística, varejo e outras aplicações para melhorar a eficiência e a gestão de recursos (Shields et al. 2015).

Engenharia Reversa: Processo de análise de um produto para determinar seus componentes e seu funcionamento (Eilam 2011). No contexto dos cartões Mifare Classic, a engenharia reversa foi aplicada para entender o algoritmo de criptografia Crypto-1.

Números Pseudo-aleatórios: São sequências de números que parecem aleatórios, mas são geradas a partir de um processo determinístico (Gentle 2003). Muitos sistemas de criptografia dependem de geradores de números pseudo-aleatórios para funções como a geração de chaves.

Criptoanálise Algébrica: É o processo de decifrar códigos através da resoluções de sistemas de equações polinomiais (Bard 2009). Esta técnica pode ser empregada para encontrar falhas em algoritmos ou sistemas de criptografia.

Nested Attack: O ataque Nested explora uma falha no protocolo de autenticação dos cartões MIFARE Classic, permitindo a recuperação de chaves criptográficas conhecidas a partir de uma chave inicial. Este ataque utiliza a análise de tempo e o comportamento previsível dos números pseudoaleatórios (nonces) gerados pelo cartão, permitindo a recuperação de chaves adicionais após a obtenção de uma chave de setor inicial (Garcia et al. 2008).

Darkside Attack: O ataque Darkside tira vantagem do código de resposta NACK (Negative Acknowledgment) emitido pelo cartão quando uma chave incorreta é enviada com bits de paridade corretos. Ao realizar um ataque de força bruta nos 4 bits criptografados do código NACK, o atacante consegue obter partes do keystream utilizado na criptografia da comunicação, permitindo a recuperação completa das chaves de autenticação sem a necessidade de uma chave inicial (NOHL 2007)

Hardnested Attack: O ataque Hardnested é uma evolução do ataque Nested, que foi projetado para contornar as proteções implementadas após a descoberta das vulnerabilidades do MIFARE Classic. Utilizando um número elevado de nonces criptografados e técnicas de criptoanálise avançadas, o ataque reduz o espaço de busca da chave e permite sua descoberta com maior eficiência, mesmo em cartões que implementaram medidas de proteção adicionais (Meijer e Verdult 2015)

3 Metodologia

Este capítulo detalha a metodologia empregada para atingir os objetivos propostos no estudo das vulnerabilidades dos cartões MIFARE Classic, destacando-se pela aplicação de técnicas de engenharia reversa, criptoanálise e experimentação prática. Serão exploradas as características técnicas dos cartões, as metodologias para identificar e explorar suas vulnerabilidades, e o desenvolvimento de estratégias de proteção.

3.1 Tipo de Pesquisa

A pesquisa adota uma abordagem mista, combinando métodos qualitativos e quantitativos. Inicialmente, será realizada uma análise documental e bibliográfica extensiva sobre as tecnologias RFID, com foco específico nos cartões MIFARE Classic. Paralelamente, a pesquisa se dedica à engenharia reversa e criptoanálise para compreender as vulnerabilidades desses cartões. Finalmente, experimentos controlados serão conduzidos para verificar as vulnerabilidades e seus respectivos impactos.

3.2 Instrumentos e Procedimentos Metodológicos

3.2.1 Experimentação Prática

A fase experimental envolverá a configuração de um ambiente controlado, no qual cartões MIFARE Classic serão submetidos a diversos tipos de ataques, replicando as condições descritas na literatura revisada. Para tal, serão utilizados leitores RFID e equipamentos especializados em segurança de sistemas de identificação por rádio frequência, como o Arduino em conjunto com o módulo MFRC522, o Proxmark3 RDV2 e o Proxmark3 RDV4.

3.2.2 Arduíno e RC522

Determinados ataques foram viáveis através da utilização de um Arduíno UNO junto ao módulo RC522 responsável por fornecer a capacidade de interação com dispositivos de radiofrequência. Para realizar as operações de manipulação sobre o cartão, foi utilizado a biblioteca **RFID** do autor *miguelbalboa* (miguelbalboa) que, contém um conjunto de códigos escritos na linguagem C++. As imagens 3.1 e 3.2 a seguir ilustram, respectivamente, como ficou a montagem final do dispositivo e a esquemática que foi utilizada durante o processo de construção:

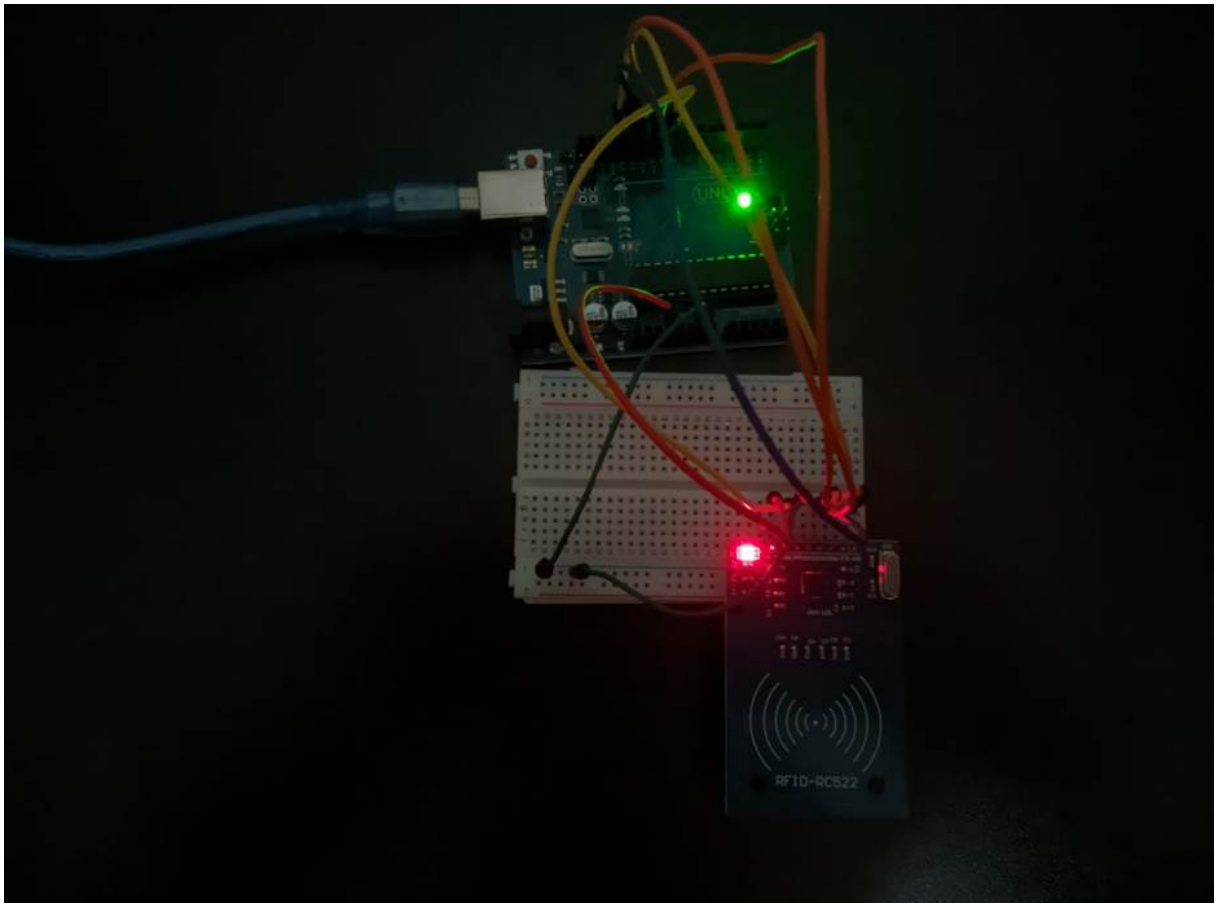

Figura 3.1 – Arduíno UNO e Módulo RC522

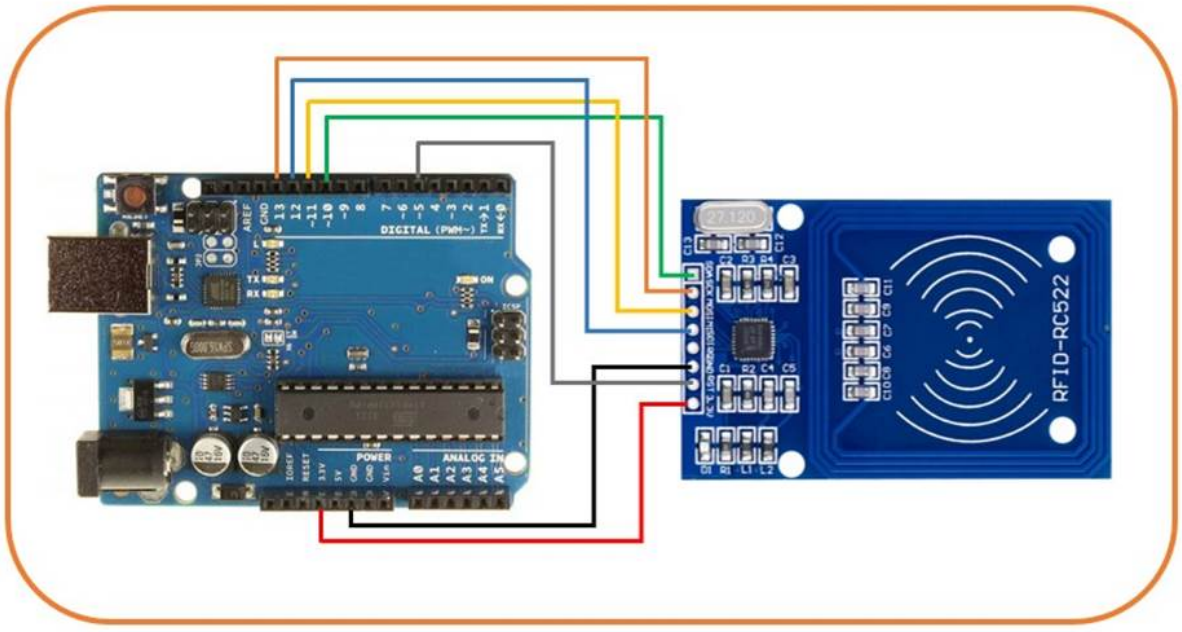

Figura 3.2 – Esquemática

3.2.3 Proxmark3 RDV2 e RDV4

Criado originalmente em 2007, o *Proxmark3* é uma ferramenta de hardware multipropósito desenvolvida para análise de segurança, pesquisa e desenvolvimento de sistemas de identificação por radiofrequência (*RFID*). O dispositivo foi projetado como uma plataforma de código aberto, permitindo a leitura, emulação e manipulação de diversos protocolos RFID, tanto de alta (13,56 MHz) quanto de baixa frequência (125/134 kHz).

A arquitetura do Proxmark3 é baseada em uma combinação de FPGA (*Field-Programmable Gate Array*) e um microcontrolador, permitindo a implementação de processamento analógico de sinais em nível avançado, modulação e demodulação. A FPGA é responsável por funções de baixo nível, enquanto o microcontrolador gerencia a codificação e decodificação dos protocolos, além da comunicação via USB com o cliente de software no PC.

O dispositivo conta com antenas independentes para as frequências de 125 kHz e 13,56 MHz, um conversor analógico-digital (ADC) de 8 bits e memória flash para armazenar o firmware, que inclui código ARM e a imagem FPGA. A evolução do firmware e do código resultou em versões otimizadas do dispositivo, como o Proxmark3 RDV4, que trazem melhorias em estabilidade e funcionalidade.

Ao contrário do exemplo com o Arduíno, certos sistemas podem utilizar recursos presentes no próprio cartão para garantir a autenticação completa do usuário. Frequentemente, esses sistemas armazenam dados sensíveis na memória do cartão, os quais serão utilizados pelo usuário após a autenticação no sistema. Por exemplo, em bilhetes eletrônicos de transporte público, o saldo do usuário é armazenado na memória do cartão e verificado no sistema da leitora antes de autorizar ou negar o acesso do usuário ao veículo.

No entanto, para obtermos acesso às informações contidas no cartão, é necessário descobrir as chaves presentes em cada setor, a fim de descriptografar, modificar e escrever os dados. Devido às limitações do Arduíno, executar todo esse processo pode ser bastante custoso e ineficiente, uma vez que o hardware do Arduíno não foi projetado especificamente para essa finalidade. É nesse contexto que o Proxmark3 surge para suprir essa necessidade de forma eficaz.

Tanto o Proxmark3 RDV2 quanto o Proxmark3 RDV4 serão utilizados como ferramentas centrais nos experimentos devido a capacidade avançada de ambos para realizar análises e execuções de ataques em sistemas RFID. Esses dispositivo permitem uma inspeção detalhada dos protocolos de comunicação e da segurança dos cartões MIFARE Classic, oferecendo funcionalidades como:

- **Clonagem de Cartões:** Para testar a facilidade com que as informações de um cartão MIFARE Classic podem ser copiadas e reproduzidas em outro cartão.
- **Análise de Protocolos:** Para examinar as trocas de mensagens entre o cartão e o leitor, identificando potenciais vulnerabilidades no processo de autenticação e na transferência

de dados.

- **Execução de Ataques Conhecidos:** Para aplicar ataques específicos, como o *Nested Attack* e ataques que exploram vulnerabilidades na geração de números pseudo-aleatórios do algoritmo Crypto-1, verificando a susceptibilidade dos cartões a essas técnicas.
- **Teste de Estratégias de Proteção:** Para avaliar a eficácia das medidas de segurança propostas, aplicando os mesmos ataques após a implementação das estratégias de proteção.

A utilização de ambos permitirá uma análise abrangente e detalhada das vulnerabilidades dos cartões MIFARE Classic, fundamentando a pesquisa com dados experimentais sólidos. A capacidade do dispositivo de simular diversos cenários de ataque e testar a efetividade das soluções de segurança propostas é crucial para o desenvolvimento de estratégias de proteção eficazes e viáveis. As figuras 3.3 e 3.4 ilustram, respectivamente, os dispositivos supracitados e que foram utilizados durante os experimentos neste estudo:

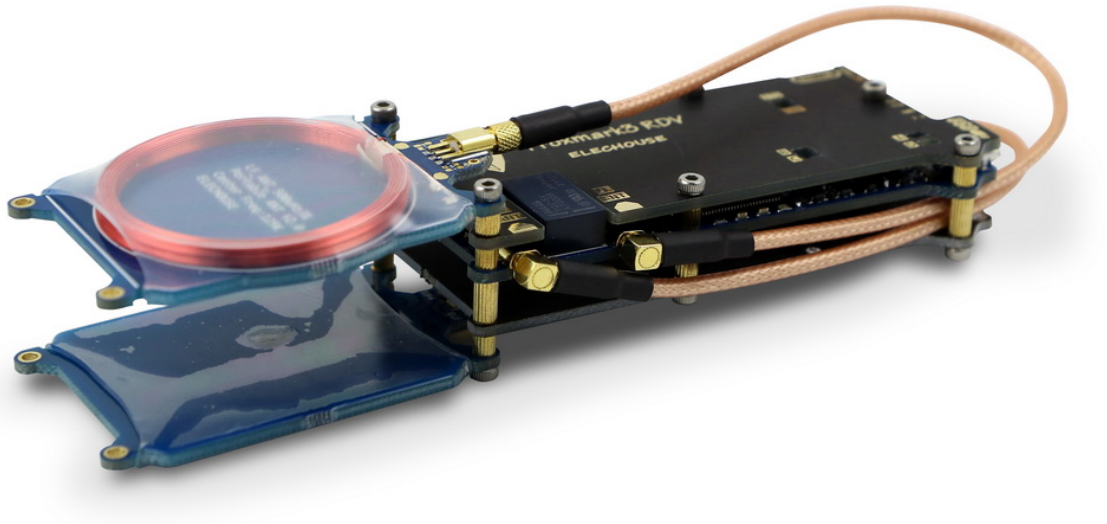

Figura 3.3 – Proxmark3 RDV2
Fonte: <https://hackerwarehouse.com/product/proxmark3-rdv2-kit/>

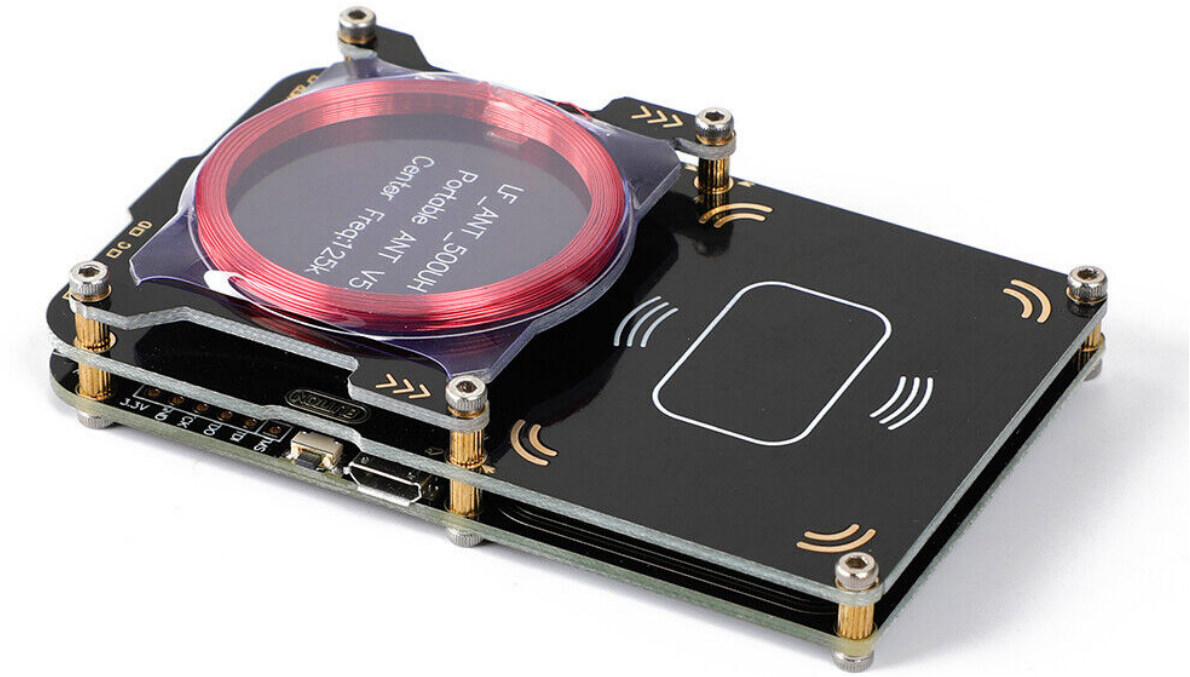

Figura 3.4 – Proxmark3 RDV4 (Easy)
Fonte: <https://www.digitalkey.it/it/sensori-prossimita/144-proxmark3-v3-easy-512m-kit-nfc-rfid-5-tag-di-test-793596617942.html>

3.3 Ataques Estudados

De maneira geral, os ataques demonstrados aqui podem ser separados em dois grupos. O primeiro inclui apenas o ataque no qual não é necessária a obtenção de nenhuma chave criptográfica (A ou B), visando apenas realizar o clone de UID para outro cartão. Para este ataque será utilizado somente o Arduino UNO junto ao módulo RC522. Já o segundo grupo inclui três tipos de ataques (*Nested*, *Darkside* e *Hardnested*) que visam explorar vulnerabilidades presentes no sistema de criptografia do cartão e a obtenção das chaves criptográficas dos setores. Para este grupo, será utilizado as versões RDV2 RDV4 do Proxmark3

3.3.1 Clone de UID

Conforme mencionado anteriormente, cada cartão MIFARE Classic possui, atrelado ao seu chip, especificamente no EPROM, um UID que serve como código de identificação do cartão. Esses códigos são atribuídos normalmente logo após a fabricação do cartão e, por padrão, não podem ser alterados, pois o setor de memória onde se encontram é exclusivamente para leitura.

Os UIDs utilizados pelos cartões MIFARE Classic seguem a norma ISO/IEC 14443-3 (Identification cards 2018) e são armazenados, de maneira não criptografada, em formato hexadecimal. Os primeiros cartões MIFARE Classic disponíveis no mercado operavam com um UID de 4 bytes, conhecido em inglês como "Single Size UID". Após determinado período, a NXP começou a fabricar novos cartões que utilizavam UIDs de 7 bytes ("Double Size UID"), devido ao fato de o número máximo de possibilidades de 4 bytes ter sido ultrapassado pelo número de cartões fabricados. Por fim, na última atualização, alguns cartões fabricados passaram a operar com UIDs de 10 bytes, conhecidos também como "Triple Size UID".

Porém, apesar de os cartões originais MIFARE Classic não permitirem a adulteração de seu UID, alguns fabricantes comercializam cartões que seguem a exata mesma estrutura do cartão original, mas com o chip configurado para que o setor de memória onde o UID se encontra esteja passível de escrita, permitindo, assim, a sua alteração. Esses cartões são conhecidos informalmente como *Chinese Magic Cards* e podem ser encontrados em diversas lojas virtuais, como ilustrado na figura 3.5:

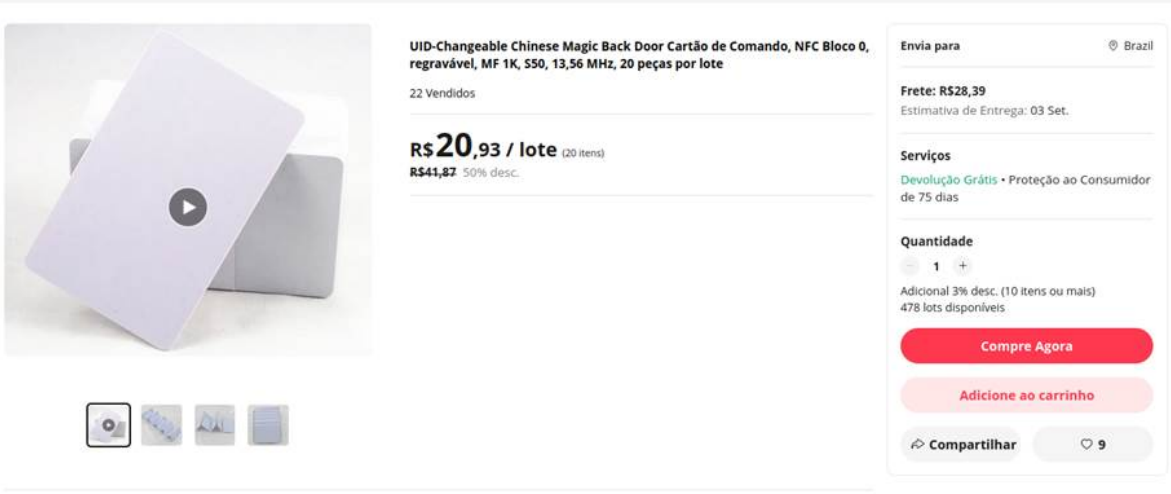

Figura 3.5 – Loja virtual comercializando cartões MIFARE Classic adulteráveis

O cerne do impacto desse ataque reside no fato de que muitos sistemas utilizam apenas o UID como forma de verificar a identidade do cartão e, com isso, fornecer acesso a locais autorizados, documentos sensíveis e outros. Como descrito anteriormente, o UID dos cartões não é criptografado; portanto, obter acesso ao mesmo pode ser feito de maneira simples, bastando apenas o contato com um dispositivo que seja capaz de operar nas frequências necessárias, como,

por exemplo, um celular que possua a tecnologia NFC.

Após obtido o UID de um cartão vítima, para realizar o clone do mesmo, basta utilizar-se de um cartão adulterável e escrever sobre o setor de memória o novo UID. Após o clone do UID realizado com sucesso, um atacante seria capaz de explorar sistemas que se baseiam exclusivamente na verificação da identidade do usuário pelo UID presente no cartão. Empresas de cibersegurança brasileiras relatam que este tipo de cenário de ataque é bastante comum (e efetivo) em sistemas de controle de acesso, como em edifícios corporativos e áreas restritas em fábricas, indústrias e escritórios corporativos.

3.3.2 Ataque *Nested*

Apresentado pela primeira vez ao público através do estudo de *Dismantling MIFARE Classic* (Garcia et al. 2008), este tipo de ataque é utilizado no cenário em que o atacante possui pelo menos uma chave de algum setor arbitrário do cartão. Esse ataque explora o mecanismo de *nonce* (desafio), descrito na seção 2.2.2, no qual o *nonce* é sempre gerado a partir de uma mesma sequência, sendo que cada iteração subsequente possui um tempo médio determinado. Esse ataque pode ser resumido nos seguintes passos:

1. O atacante inicia uma autenticação no setor do qual ele conhece a chave e, por conta disso, o cartão gera um primeiro *nonce*.

2. Em seguida, o atacante realiza outra autenticação em outro setor, no qual ele não conhece a chave.

3. O cartão responde a essa nova autenticação com um *nonce* criptografado. Calculando a "distância" entre os dois *nonces* através do monitoramento do tempo entre as duas autenticações consecutivas, o atacante consegue prever qual era o *nonce* descriptografado que foi gerado na segunda autenticação.

4. De posse do *nonce* descriptografado e criptografado, o atacante realiza uma operação de *XOR* (Ou Exclusivo), recuperando 32 bits do *keystream* que foi utilizado para cifrar o *nonce*.

5. Após esse processo, o atacante tem um espaço de busca muito menor e pode recorrer a ataques de força bruta para descobrir os bits restantes do *keystream* e determinar qual é a respectiva chave do setor que está sendo atacado.

6. Finalmente, para descobrir as chaves dos outros setores, basta repetir o processo com as chaves obtidas anteriormente.

O diagrama 3.6 ilustra, de maneira simplificada, os passos supracitados:

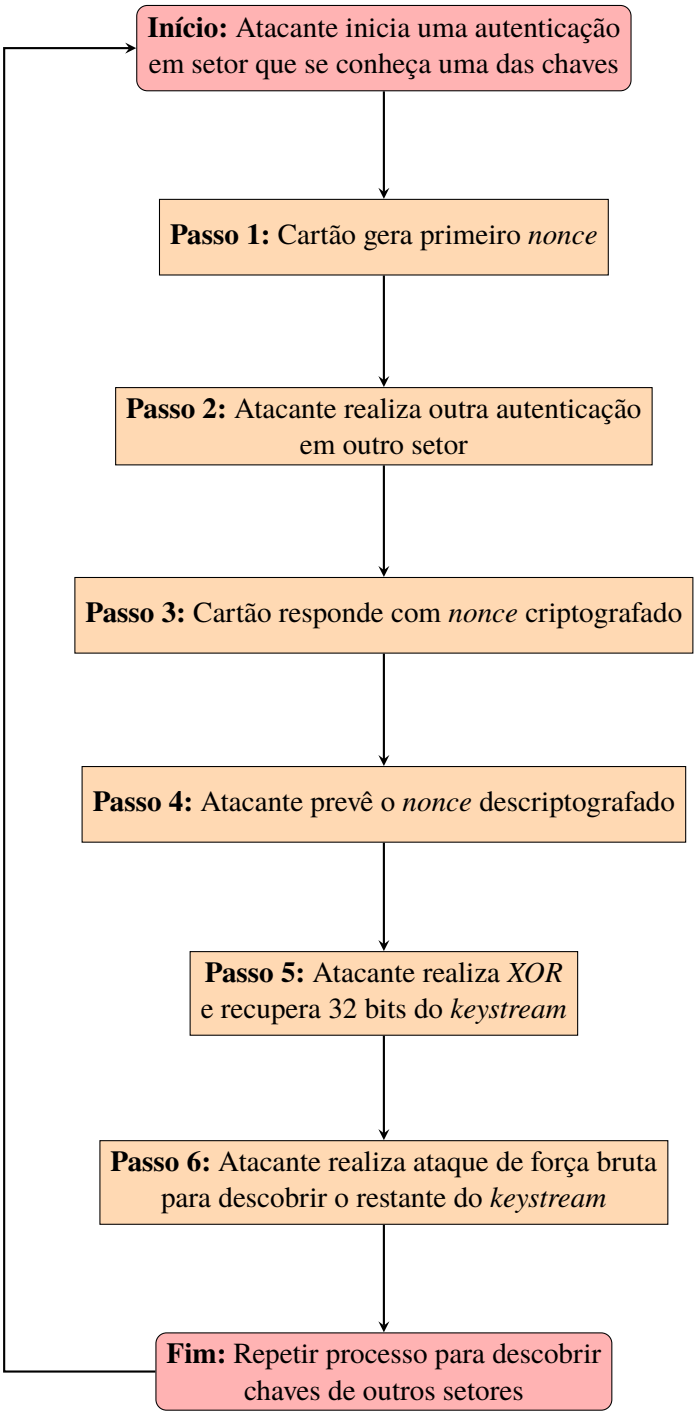

Figura 3.6 – Diagrama do processo de autenticação explorado pelo Nested Attack.

É importante pontuar que esse processo pode ser combinado com outras análises que visam reduzir o número de possibilidades do *keystream*, permitindo ao atacante se aproximar cada vez mais da respectiva chave do setor.

3.3.3 Ataque *Darkside*

Em cenários em que o atacante não possui nenhuma chave de nenhum setor do cartão, é necessário utilizar outro tipo de ataque para se obter êxito. O ataque utilizado nesse cenário é popularmente conhecido como *Darkside*.

O ataque *Darkside* foi inicialmente descrito no artigo intitulado *The Dark Side of Security by Obscurity: And Cloning MiFare Classic Rail and Building Passes, Anywhere, Anytime* (NOHL 2007). Nesse artigo, o autor explora diversos aspectos relacionados ao processo de autenticação e discute questões associadas à segurança baseada em obscuridade.

O ataque *Darkside* tem como lógica central, aproveitar-se do código de resposta *NACK* (*Negative Acknowledgment*), que é gerado quando os bits de paridade enviados ao cartão estão corretos, mesmo que a chave selecionada não seja a correta. Nessa situação, o cartão responde com um código *NACK* (4 bits) criptografado com a *keystream*. Como o código *NACK* é pequeno, pode-se realizar um ataque de força bruta rapidamente até que o código *NACK* antes de ser cifrado seja descoberto. Após isso, é possível realizar uma operação *XOR* entre o código descriptografado e o código criptografado, permitindo a obtenção de parte da *keystream* utilizada. De forma simplificada, este ataque pode ser resumido da seguinte maneira:

1. Posiciona-se o Proxmark 3 próximo ao cartão. Em seguida, inicia-se o processo de ataque.
2. Envia-se um criptograma gerado pelo dispositivo do atacante, que possui os bits de paridade corretos, de modo que, quando o cartão realiza a checagem, seja emitido o código *NACK*.
3. Devido à paridade correta dos bits, o cartão responde com o código *NACK* (4 bits) cifrado com a *keystream*.
4. Realiza-se o ataque de força bruta para descobrir quais são os respectivos 4 bits de texto plano do *NACK*. Como 4 bits permitem apenas 2^4 possibilidades, o código em texto plano pode ser obtido rapidamente.
5. Após a descoberta do código em texto plano, realiza-se o processo de *XOR* com o código cifrado para obter 4 bits da *keystream*.
6. O processo deve ser repetido, junto às outras análises de predição, até que se obtenha a chave do setor atacado.

O diagrama 3.7 ilustra, de maneira simplificada, os passos supracitados:

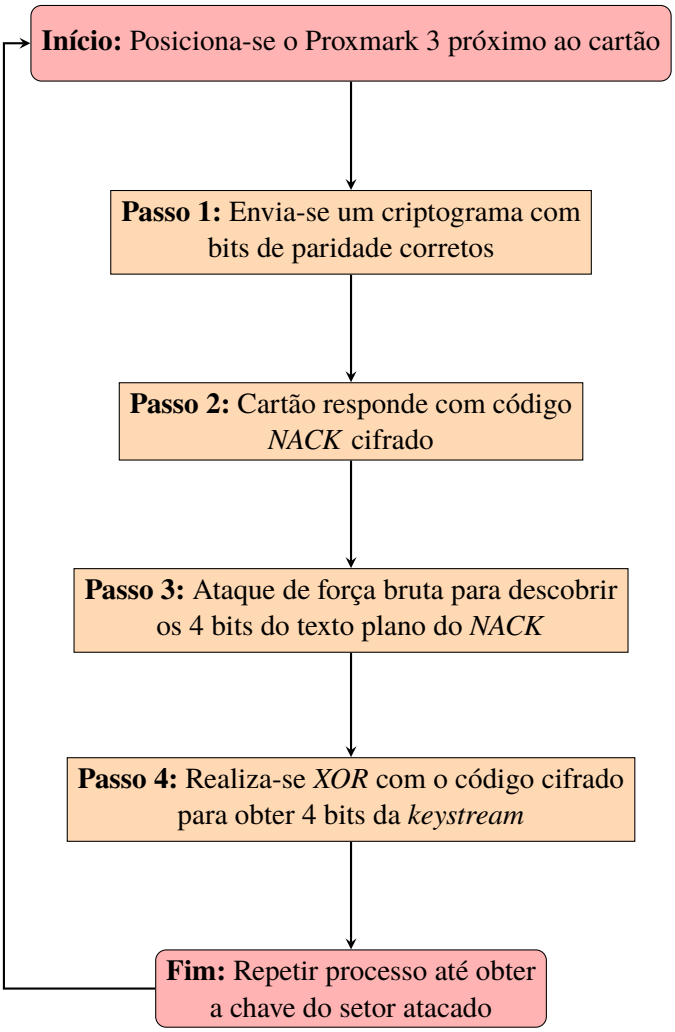

Figura 3.7 – Diagrama simplificado do ataque Darkside.

Após a descoberta de uma chave, esta pode ser utilizada para realizar o ataque do tipo *Nested*, já descrito, para obter as chaves correspondentes dos outros setores.

3.3.4 Ataque *Hardnested*

Após a repercussão dos dois tipos de ataques mencionados anteriormente, principalmente do ataque do tipo *Nested*, a fabricante dos cartões MIFARE (NXP) identificou a necessidade de mitigar as vulnerabilidades associadas ao seu produto.

Em resposta a essa situação, a solução adotada na época foi a implementação de mudanças no algoritmo de geração de números pseudoaleatórios, responsável por gerar os *nonces* utilizados durante o processo de autenticação. A alteração aplicada tinha como principal objetivo dificultar ao máximo a predição do valor do *nonce* em texto plano, elemento-chave para a execução do ataque do tipo *Nested*.

Entretanto, em 2015, foi demonstrado no estudo (Meijer e Verdult 2015) que, por meio da coleta de diversos valores de *nonces* criptografados, aliado a algumas técnicas de análises criptográficas, é possível, em tempo hábil, obter a chave de um setor arbitrário.

Esse processo é consideravelmente mais complexo do que os ataques mencionados anteriormente. Uma breve síntese dos passos envolvidos neste tipo de ataque pode ser apresentada:

1. O atacante coleta entre 2000 e 4000 *nonces* gerados pelo cartão por meio de tentativas consecutivas de autenticação.
2. Em seguida, realiza-se uma série de análises bit a bit a respeito da natureza desses *nonces* para que o escopo da chave original (2^{48}) seja reduzido ao máximo.
3. Após a redução do escopo, é realizado um ataque de força bruta para se obter a chave original.

O diagrama do ataque Hardnested está ilustrado na Figura 3.8.

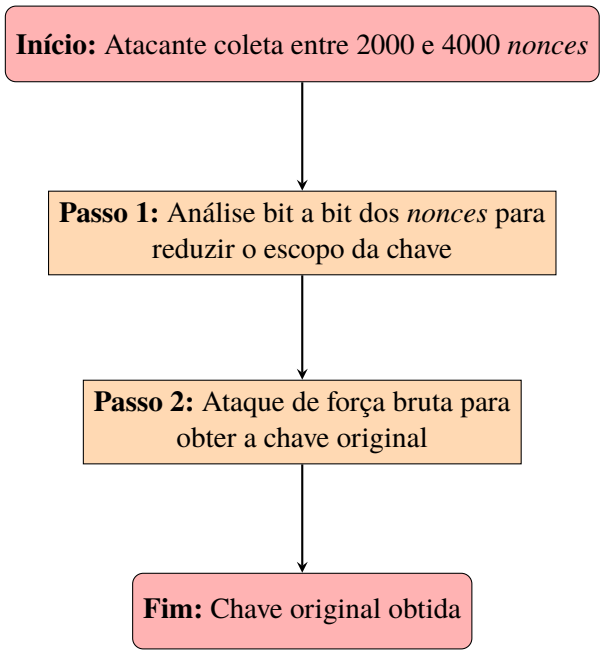

Figura 3.8 – Diagrama simplificado do ataque Hardnested.

Cabe destacar que este ataque foi o único para o qual não foi possível obter informações suficientes que permitissem sua realização prática. Durante o processo de pesquisa, não foi possível obter um cartão que possuísse o novo mecanismo de PRNG mencionado. Portanto, a compreensão apresentada baseia-se exclusivamente em estudos de casos e experiências relatadas por outros pesquisadores e entusiastas da área.

3.4 Análise dos Resultados

Os dados coletados durante a experimentação serão analisados quantitativamente para determinar a eficácia dos ataques e qualitativamente para compreender o impacto das vulnerabilidades no contexto de segurança dos cartões. A análise qualitativa se concentrará na interpretação dos dados obtidos, buscando identificar padrões, semelhanças e diferenças nas vulnerabilidades exploradas. Também será realizado uma comparação entre os resultados obtidos entre ambos Proxmark3 para evidenciar vantagens e desvantagens de um modelo em relação ao outro.

4 Experimentos

Neste capítulo, serão demonstrados os experimentos realizados envolvendo os ataques descritos no capítulo anterior. Os ataques foram implementados utilizando duas plataformas distintas: o Arduino UNO com o módulo RC522 e o dispositivo Proxmark3 nas versões RDV2 e RDV4. Os experimentos visam comprovar na prática as vulnerabilidades presentes nos cartões MIFARE Classic e a eficácia dos métodos de ataque estudados.

4.1 Arduino UNO e RC522

O módulo RC522 é um leitor/gravador RFID de baixo custo que, em conjunto com o Arduino UNO, permite a comunicação com cartões MIFARE Classic. Para a realização dos experimentos, utilizou-se a biblioteca MFRC522, que fornece funções para leitura e escrita de dados nos cartões.

A figura 4.1 ilustra o processo de clonagem de UID (Identificador Único) de um cartão MIFARE Classic. Este tipo de ataque não requer a obtenção das chaves criptográficas (A ou B) dos setores, pois atua diretamente no UID do cartão. O código utilizado faz parte da biblioteca mencionada, adaptado para sobrescrever um novo UID em um cartão compatível:

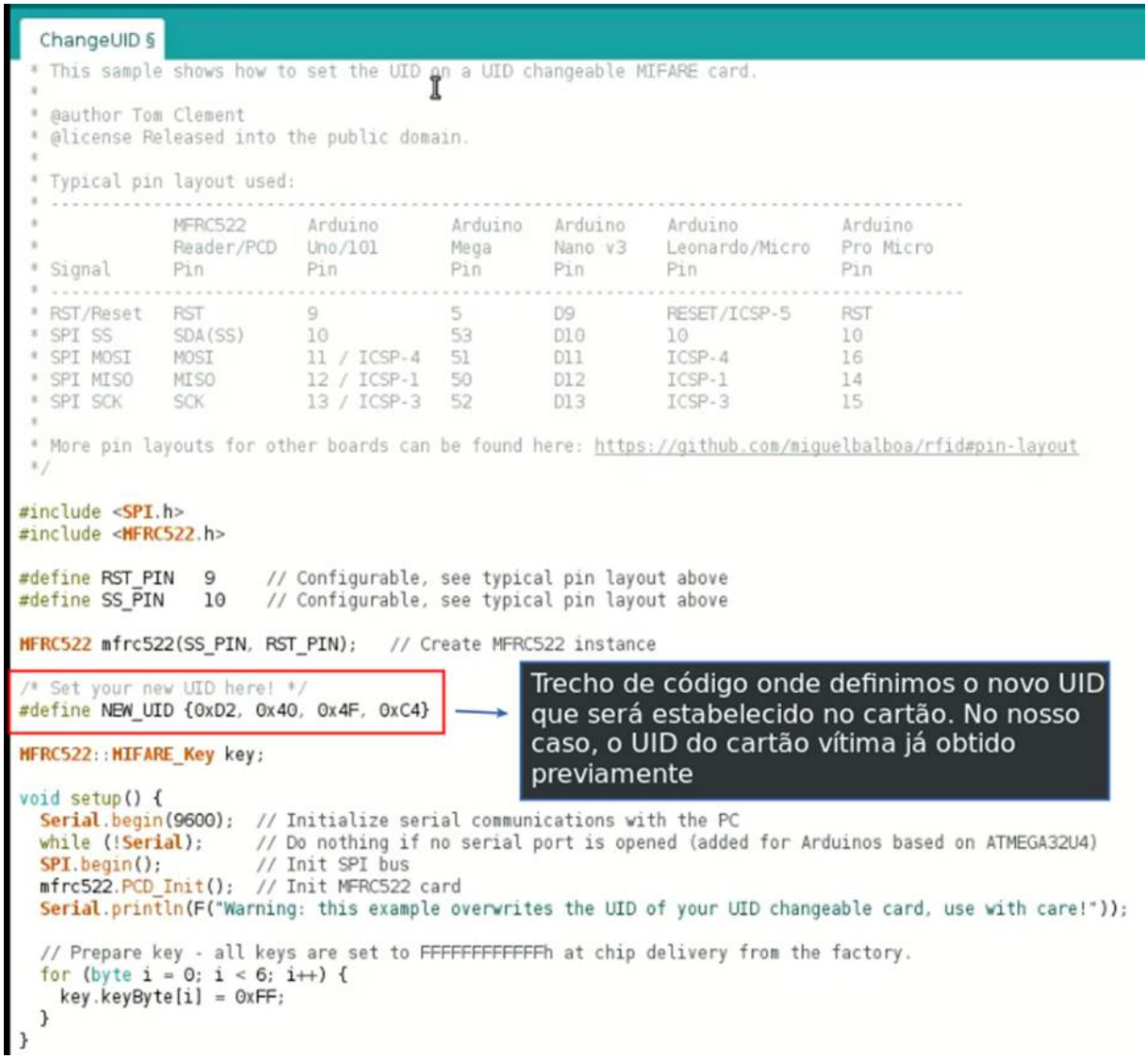

Figura 4.1 – Trecho do código utilizado para sobrescrever novo UID

Após a execução do código, o cartão alvo passa a responder com o novo UID programado, efetivamente clonando a identidade do cartão original.

4.2 Proxmark3

O Proxmark3 é uma ferramenta avançada para análise e manipulação de dispositivos RFID, suportando diversas frequências e protocolos. Nos experimentos seguintes, foram utilizados os modelos RDV2 e RDV4 para realizar ataques mais complexos aos cartões MIFARE Classic, explorando vulnerabilidades criptográficas para obter as chaves dos setores.

4.2.1 Ataque *Nested*

O ataque *Nested* explora uma falha no protocolo de autenticação dos cartões MIFARE Classic, permitindo recuperar chaves criptográficas conhecidas a partir de uma chave inicial. Nos experimentos, utilizou-se a chave de fábrica (**FFFFFFFFFFFF**) como ponto de partida, comum em cartões que não foram devidamente configurados.

A seguir, os pseudocódigos buscam evidenciar a implementação a nível computacional deste tipo de ataque. Este pseudocódigo é uma adaptação do código original implementado no *firmware Iceman*, utilizado em ambas as versões do Proxmark3:

Algorithm 1 Ataque *Nested*

Data: Conjunto de valores pNK, tamanho $tamanhoPNK$, identificador $authuid$
Result: Lista de chaves recuperadas e unificadas

Inicializar $keyCount \leftarrow 0$
Definir número de threads conforme necessário
Inicializar lista de chaves possíveis `possibleKeys` vazia
for *cada thread* **do**
    Inicializar parâmetros para a thread
    Iniciar a thread com a função abaixo para recuperação de chaves
    `// Função executada por cada thread para recuperar chaves (nested_recover)`
    for *cada posição designada* **do**
        Calcular nt_probe e $ks1$
        Recuperar o estado inicial do LFSR usando `lfsr_recovery32`
        while *existirem estados possíveis* **do**
            Retroceder o LFSR com `lfsr_rollback_word`
            Armazenar a chave possível em `possibleKeys`
        end
    end
    Atualizar $keyCount$
end
Esperar todas as threads terminarem
if $keyCount = 0$ **then**
    Imprimir "Nenhuma chave foi recuperada." **return** NULO
end
`// Ordenar e unificar chaves`
Ordenar `possibleKeys`
Contar a frequência de cada chave
Ordenar as chaves por frequência decrescente
Selecionar as chaves com maior frequência
return chaves selecionadas

As figuras 4.2 e 4.3 a seguir ilustram a execução do ataque *Nested* utilizando o Proxmark3 RDV2 e o RDV4 em um cartão MIFARE Classic arbitrário. O bloco atacado foi o bloco 0, que contém informações críticas como o UID e os dados de manufatura.

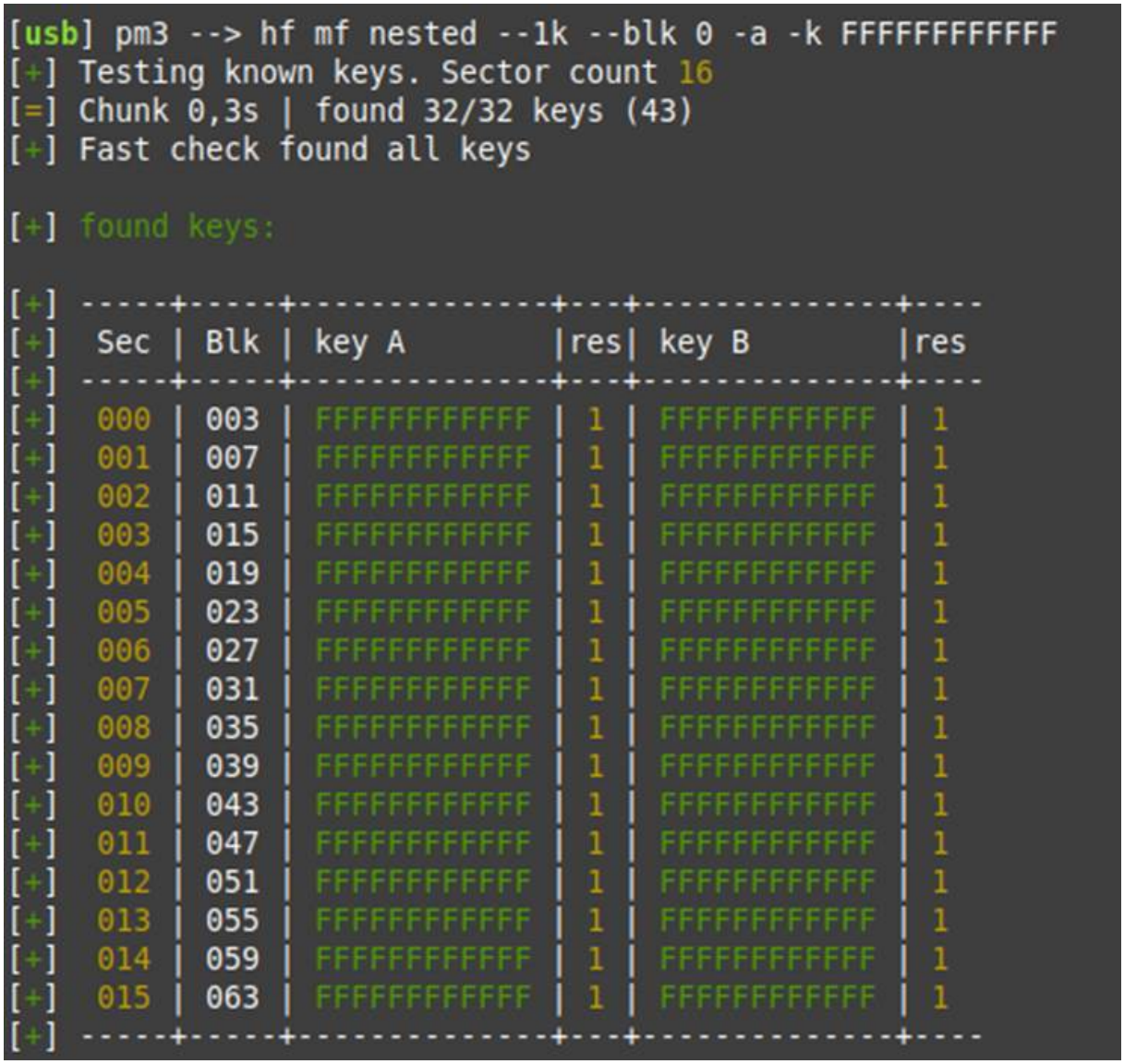

Figura 4.2 – Execução do ataque *Nested* através do Proxmark3 RDV2

```
[usb] pm3 --> hf mf nested --1k --blk 0 -a -k FFFFFFFFFFFF
[+] Testing known keys. Sector count 16
[+] Fast check found all keys

[+] found keys:

[+] -----+-----+--------------+---+--------------+----
[+]  Sec | Blk | key A        |res| key B        |res
[+] -----+-----+--------------+---+--------------+----
[+]  000 | 003 | FFFFFFFFFFFF | 1 | FFFFFFFFFFFF | 1
[+]  001 | 007 | FFFFFFFFFFFF | 1 | FFFFFFFFFFFF | 1
[+]  002 | 011 | FFFFFFFFFFFF | 1 | FFFFFFFFFFFF | 1
[+]  003 | 015 | FFFFFFFFFFFF | 1 | FFFFFFFFFFFF | 1
[+]  004 | 019 | FFFFFFFFFFFF | 1 | FFFFFFFFFFFF | 1
[+]  005 | 023 | FFFFFFFFFFFF | 1 | FFFFFFFFFFFF | 1
[+]  006 | 027 | FFFFFFFFFFFF | 1 | FFFFFFFFFFFF | 1
[+]  007 | 031 | FFFFFFFFFFFF | 1 | FFFFFFFFFFFF | 1
[+]  008 | 035 | FFFFFFFFFFFF | 1 | FFFFFFFFFFFF | 1
[+]  009 | 039 | FFFFFFFFFFFF | 1 | FFFFFFFFFFFF | 1
[+]  010 | 043 | FFFFFFFFFFFF | 1 | FFFFFFFFFFFF | 1
[+]  011 | 047 | FFFFFFFFFFFF | 1 | FFFFFFFFFFFF | 1
[+]  012 | 051 | FFFFFFFFFFFF | 1 | FFFFFFFFFFFF | 1
[+]  013 | 055 | FFFFFFFFFFFF | 1 | FFFFFFFFFFFF | 1
[+]  014 | 059 | FFFFFFFFFFFF | 1 | FFFFFFFFFFFF | 1
[+]  015 | 063 | FFFFFFFFFFFF | 1 | FFFFFFFFFFFF | 1
[+] -----+-----+--------------+---+--------------+----
[+] ( 0:Failed / 1:Success )
```

Figura 4.3 – Execução do ataque *Nested* através do Proxmark3 RDV4

Como pode ser observado, o ataque foi bem-sucedido em ambos os dispositivos, resultando na recuperação das chaves dos setores do cartão. Isso permite acesso completo aos dados armazenados, possibilitando leitura, escrita e clonagem do cartão.

4.2.2 Ataque *Darkside*

O ataque *Darkside* é uma técnica que explora uma vulnerabilidade específica no algoritmo de criptografia dos cartões MIFARE Classic, permitindo a recuperação das chaves sem conhecimento prévio. Diferentemente do ataque *Nested*, o *Darkside* não requer uma chave inicial conhecida.

A seguir, os pseudocódigos buscam evidenciar a implementação a nível computacional deste tipo de ataque. Este pseudocódigo é uma adaptação do código original implementado no *firmware Iceman*, utilizado em ambas as versões do Proxmark3:

Algorithm 2 Ataque *Darkside*

Data: Número do bloco `blockno`, tipo da chave `key_type`
Result: Chave recuperada `key`

Inicializar variáveis
`first_run` ← `verdadeiro`
while *verdadeiro* **do**
    if *tecla de abortar pressionada* **then**
        Enviar comando de interrupção e retornar erro
    end
    Preparar e enviar comando para iniciar o ataque Darkside
    Imprimir "Executando ataque Darkside"
    while *aguardando resposta* **do**
        if *tecla de abortar pressionada ou erro de comunicação* **then**
            Enviar comando de interrupção e retornar erro
        end
        if *resposta recebida* **then**
            Extrair `isOK` e dados da resposta
            if *`isOK` indica erro* **then**
                Tratar erros com base em `isOK` e retornar erro
            end
            else
                Extrair `uid`, `nt`, `par_list`, `ks_list`, `nr`, `ar`
            end
        end
    end
    if *`par_list` == 0 e `first_run` == `verdadeiro`* **then**
        Imprimir mensagem sobre paridade zero
    end
    `first_run` ← `falso`
    Chamar `nonce2key` para obter lista de chaves `keylist`
    if *nenhuma chave encontrada* **then**
        Imprimir mensagem e
    end
    if *`par_list` == 0* **then**
        Atualizar `keylist` com a interseção de `last_keylist` **if** *nenhum candidato encontrado* **then**
            Atualizar `last_keylist` e continuar
        end
    end
    Imprimir número de chaves candidatas encontradas
    `key` ← `nulo`
    for *cada lote de chaves em `keylist`* **do**
        Converter chaves para bytes e armazenar em `keyBlock`
        if *`mfCheckKeys` com `keyBlock` for bem-sucedido* **then**
            `key` ← chave encontrada
        end
    end
    if *`key` não é nulo* **then**
    end
    else
        Imprimir mensagem de falha dos candidatos
        Atualizar `last_keylist`, definir `first_run` ← `verdadeiro` e continuar
    end
end
Liberar recursos e retornar sucesso

A seguir, as figuras 4.4 e 4.5 ilustram a execução do ataque *Darkside* através da utilização do Proxmark3 RDV2 e RDV4. O bloco alvo do ataque foi novamente o bloco 0:

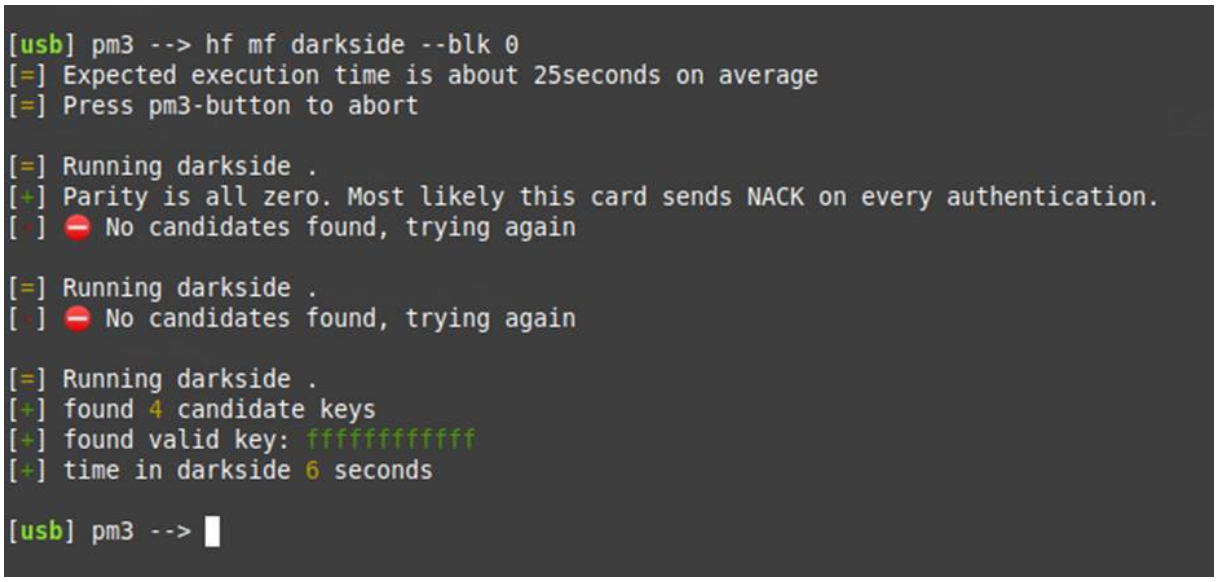

Figura 4.4 – Execução do ataque *Darkside* através do Proxmark3 RDV2

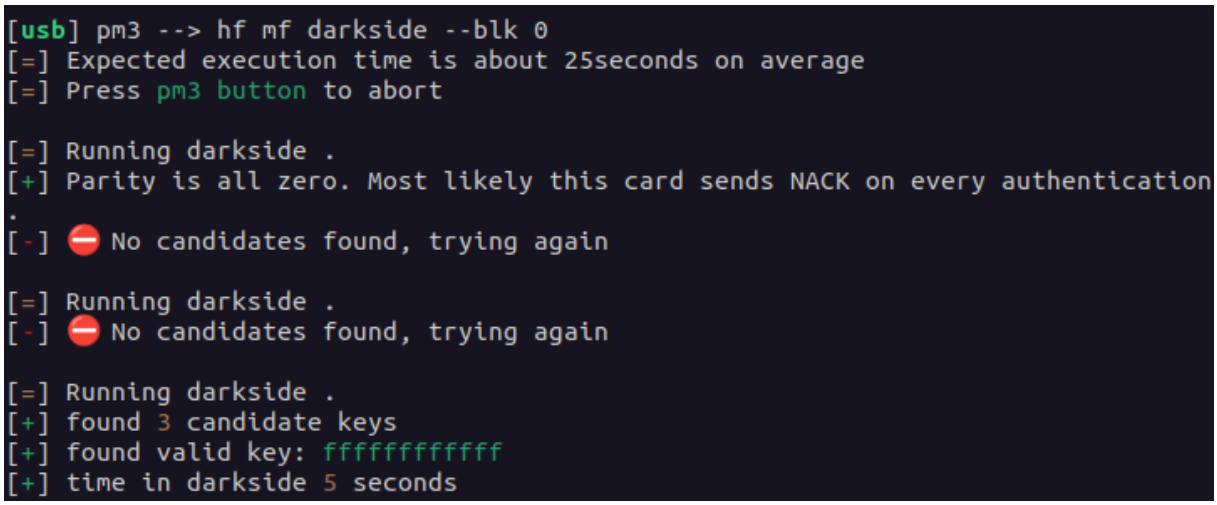

Figura 4.5 – Execução do ataque *Darkside* através do Proxmark3 RDV4

Os resultados confirmam a eficácia do ataque *Darkside* em ambos os modelos do Proxmark3, permitindo a extração das chaves criptográficas sem informações prévias.

5 Resultados

Neste capítulo, são apresentados os resultados obtidos através dos experimentos realizados com os cartões MIFARE Classic, utilizando duas plataformas distintas: Arduino UNO com o módulo RC522 e Proxmark3 nas versões RDV2 e RDV4. O objetivo principal foi comprovar a possibilidade de execução dos ataques estudados e quais os requisitos para executar os mesmos, além de avaliar as vulnerabilidades mencionadas previamente.

5.1 Clonagem de UID

Os experimentos iniciais focaram na clonagem do UID dos cartões MIFARE Classic utilizando o Arduino UNO com o módulo RC522. O processo de clonagem foi bem-sucedido, permitindo sobrescrever o UID de um cartão compatível. Além de ser um ataque relativamente simples de implementar, seu custo é extremamente baixo. Os componentes necessários, como o Arduino, o módulo RC522 e os cartões adulteráveis, podem ser adquiridos facilmente em lojas de eletrônica ou pela internet por valores acessíveis. Essa acessibilidade torna o ataque particularmente perigoso, uma vez que até mesmo pessoas com conhecimentos técnicos básicos e baixo investimento financeiro podem realizar clonagens de UID. A simplicidade e o custo reduzido deste ataque demonstram uma grave vulnerabilidade, especialmente em sistemas que utilizam apenas o UID para autenticação.

5.2 Ataque Nested

O ataque Nested foi executado com sucesso utilizando as versões RDV2 e RDV4 do Proxmark3. Esse ataque explora uma falha significativa no protocolo de autenticação dos cartões MIFARE Classic, permitindo a recuperação de todas as chaves criptográficas de cada setor a partir de uma chave inicial conhecida. Nos testes realizados, a chave de fábrica padrão (**FFFFFFFFFFFF**) foi usada como ponto de partida, o que demonstrou a eficácia do ataque, especialmente em cartões que não foram devidamente configurados com medidas de segurança robustas.

Durante os experimentos, foi possível recuperar todas as chaves dos setores do cartão, permitindo acesso completo a todos os blocos de dados armazenados. Desta maneira, o ataque permitiu a manipulação de dados em todos os setores, revelando a total vulnerabilidade do cartão ao ataque. A capacidade de acessar e modificar esses dados, aliados à simplicidade da execução do ataque, demonstra o alto risco de segurança para sistemas que utilizam cartões MIFARE Classic sem camadas adicionais de proteção

Em ambos os modelos do Proxmark3, RDV2 e RDV4, o ataque Nested foi realizado com eficiência, comprovando que ferramentas disponíveis comercialmente podem explorar essa falha de maneira simples. Essa vulnerabilidade ressalta a necessidade urgente de substituição dos sistemas que dependem exclusivamente dos cartões MIFARE Classic ou de, pelo menos, atualizar seus mecanismos de autenticação para dificultar ataques dessa natureza.

5.3 Ataque Darkside

O ataque Darkside também foi realizado com sucesso nos mesmos modelos de Proxmark3, RDV2 e RDV4. Ao contrário do ataque Nested, que requer o conhecimento de pelo menos uma chave criptográfica inicial, o Darkside é ainda mais perigoso, pois permite a recuperação das chaves sem qualquer conhecimento prévio das mesmas. Esse ataque explora vulnerabilidades relacionadas ao código NACK (Negative Acknowledgment) gerado pelos cartões durante o processo de autenticação e também falhas presentes no algoritmo pseudo-aleatório responsável por gerar os desafios (*nonces*) durante o processo de autenticação.

Durante o processo de autenticação, o cartão responde com o código NACK quando a chave enviada está incorreta, mas os bits de paridade estão corretos. O ataque Darkside se aproveita dessa resposta para quebrar a criptografia do cartão, ao realizar uma força bruta eficiente no código NACK criptografado. Como o código NACK tem apenas 4 bits, a combinação de possibilidades é limitada, facilitando o trabalho de força bruta. Ao comparar o código NACK criptografado com a versão em texto claro, é possível recuperar partes do keystream utilizado para criptografar a comunicação, o que, eventualmente, leva à recuperação completa da chave de autenticação.

O ataque Darkside foi capaz de quebrar a criptografia de múltiplos setores do cartão sem a necessidade de uma chave inicial conhecida, revelando a extensão da vulnerabilidade. Este ataque demonstra a gravidade da falha de segurança, já que permite que um atacante, sem qualquer conhecimento prévio ou acesso autorizado, consiga acessar dados sensíveis armazenados no cartão. Esse cenário é particularmente perigoso em sistemas de controle de acesso que utilizam os cartões MIFARE Classic, pois um atacante pode comprometer a segurança do sistema com um esforço relativamente pequeno.

5.4 Comparação de Resultados

Os resultados dos ataques Nested e Darkside foram comparados entre as versões RDV2 e RDV4 do Proxmark3. Embora ambos os modelos tenham se mostrado extremamente eficazes na execução dos ataques, utilizando o mesmo firmware, não foram observadas diferenças significativas no desempenho entre as duas versões durante a realização dos experimentos. A principal distinção entre os modelos reside nas melhorias de hardware presentes no RDV4 (Easy), como o

tamanho reduzido e uma interface mais amigável, o que pode facilitar seu uso em determinadas situações. No entanto, quando se trata especificamente da execução dos ataques Nested e Darkside, ambas as versões entregaram resultados similares em termos de tempo de execução e eficácia na recuperação das chaves criptográficas e no acesso aos dados dos cartões MIFARE Classic.

Por outro lado, ao comparar os resultados obtidos com o Proxmark3 (em ambas as versões) aos obtidos com o Arduino UNO, utilizando o módulo RC522, ficou evidente que o hardware especializado do Proxmark3 é mais eficiente para a realização de determinados ataques. Embora o Arduino tenha sido capaz de executar alguns ataques, como a clonagem de UID, ele não foi projetado especificamente para análises de segurança RFID. Consequentemente, seu desempenho foi consideravelmente inferior ao do Proxmark3, por não possuir a capacidade de realizar ataques mais complexos, como os ataques Nested e Darkside. Além disso, o Proxmark3 se destaca por acessar e manipular dados de forma muito mais aprofundada, graças ao seu firmware robusto e ao design focado na segurança RFID.

Essa comparação ressalta a importância de utilizar ferramentas especializadas, como o Proxmark3, para a exploração de vulnerabilidades complexas em sistemas RFID. Embora o Arduino ofereça uma solução de baixo custo para ataques mais simples, como a clonagem de UID, ele é limitado em sua capacidade de realizar ataques avançados de maneira eficiente. Isso demonstra que, com as ferramentas certas e o firmware apropriado, as vulnerabilidades dos cartões MIFARE Classic podem ser facilmente exploradas, reforçando a necessidade de implementar proteções mais avançadas e robustas para mitigar esses riscos.

6 Considerações Finais

Neste trabalho, foram exploradas de maneira aprofundada as vulnerabilidades presentes nos cartões MIFARE Classic, com ênfase nos ataques Nested e Darkside, amplamente utilizados para comprometer a segurança desses sistemas. Além disso, foi explicada em detalhes a engenharia por trás do cartão MIFARE Classic, sua estrutura lógica e a cifra de fluxo Crypto-1, incluindo os conceitos iniciais de criptografia necessários para compreender o funcionamento dessa cifra e as vulnerabilidades exploradas. A análise e experimentação realizadas demonstraram que as fragilidades dos cartões MIFARE Classic podem ser exploradas de maneira relativamente simples, utilizando-se ferramentas acessíveis como o Arduino UNO junto a módulos externos complementares e dispositivos próprios para esta finalidade, tais como o Proxmark3.

Nossos achados ressaltam a necessidade urgente de rever a utilização dos cartões MIFARE Classic em aplicações críticas. Organizações que dependem dessa tecnologia para controle de acesso, pagamento eletrônico e outras funções sensíveis estão expostas a riscos significativos de segurança. Além disso, a ampla disponibilidade de ferramentas e tutoriais na internet facilita a execução desses ataques por indivíduos com conhecimento técnico limitado, ampliando o potencial de ameaças.

6.1 Conclusão

Esta pesquisa iniciou-se com uma extensa revisão da literatura, evidenciando as principais vulnerabilidades associadas ao protocolo de autenticação do MIFARE Classic. Também foi detalhado o funcionamento do cartão, sua estrutura interna e a cifra Crypto-1, um componente fundamental para entender como os ataques exploram as fraquezas no processo de autenticação. A execução dos ataques Nested e Darkside confirmou a eficiência das técnicas de quebra de segurança, recuperando as chaves criptográficas dos setores dos cartões, revelando dados sensíveis e expondo as falhas de segurança do sistema.

Os resultados experimentais mostraram que, embora o Arduino UNO com o módulo RC522 seja capaz de realizar ataques mais simples, como a clonagem de UID, seu desempenho é significativamente inferior ao do Proxmark3, uma ferramenta especificamente projetada para esse tipo de análise. A facilidade de acesso aos componentes e à tecnologia, com baixo custo e disponibilidade pela internet, ressalta a urgência em desenvolver e implementar medidas de proteção mais robustas para mitigar os riscos.

Embora o MIFARE Classic tenha sido uma solução popular devido ao seu baixo custo e facilidade de implementação, a descoberta de suas vulnerabilidades significativas levou à recomendação de migração para tecnologias mais seguras. Cartões como o MIFARE Plus e o MI-

FARE DESFire EV2 oferecem algoritmos de criptografia avançados, como AES, proporcionando níveis mais altos de segurança. A transição para essas tecnologias é essencial para organizações que buscam proteger seus sistemas contra ataques cada vez mais sofisticados.

Com base nas experimentações com o Proxmark3 RDV2 e RDV4, verificou-se que o desempenho de ambas as versões é similar, sendo o principal diferencial as melhorias no hardware da versão RDV4, que facilitam a execução dos ataques. Tanto o ataque Nested quanto o Darkside foram implementados com sucesso, reforçando a gravidade das vulnerabilidades presentes nos cartões MIFARE Classic, especialmente em sistemas que dependem exclusivamente da autenticação por UID.

Este estudo não apenas destaca as falhas críticas no MIFARE Classic, mas também serve como um alerta para a necessidade contínua de monitoramento e atualização das medidas de segurança em sistemas RFID. Com o avanço constante das técnicas de ataque, é fundamental que os sistemas de autenticação evoluam para incorporar protocolos mais robustos e resistentes a vulnerabilidades conhecidas e emergentes.

6.2 Trabalhos Futuros

A continuidade deste trabalho pode seguir diversas direções, considerando o campo vasto das pesquisas em segurança de sistemas RFID. Propostas para trabalhos futuros incluem:

- A análise e experimentação com outros tipos de cartões RFID, especialmente os que utilizam algoritmos de criptografia mais robustos que o Crypto-1, visando identificar se as mesmas vulnerabilidades estão presentes;
- O desenvolvimento de contramedidas práticas que possam ser implementadas nos sistemas existentes para mitigar ataques baseados em clonagem e recuperação de chaves, especialmente em setores críticos, como transporte público e controle de acesso;
- A investigação das dificuldades e melhores práticas na migração de sistemas baseados em MIFARE Classic para tecnologias mais seguras, como MIFARE Plus ou DESFire, incluindo análises de custo-benefício e impactos operacionais.

Referências

BARD, G. *Algebraic cryptanalysis*. [S.l.]: Springer Science & Business Media, 2009.

BRUCE, S. *Applied Cryptography: Protocols, Algorthms, and Source Code in C.-2nd*. [S.l.]: John Wiley & Sons, 1996.

CHIU, Y.-H.; HONG, W.-C.; CHOU, L.-P.; DING, J.; YANG, B.-Y.; CHENG, C.-M. A practical attack on patched mifare classic. p. 150–164, 2013.

COLEMAN, C. *An introduction to radio frequency engineering*. [S.l.]: IET, 2004.

COSTIN, A. *GitHub - nfc-tools/mfcuk: MiFare Classic Universal toolKit (MFCUK) — github.com*. 2010. <https://github.com/nfc-tools/mfcuk>. [Accessed 17-11-2023].

COURTOIS, N. T. The dark side of security by obscurity - and cloning mifare classic rail and building passes, anywhere, anytime. *IACR Cryptol. ePrint Arch.*, v. 2009, p. 137, 2009. Disponível em: <https://api.semanticscholar.org/CorpusID:11438350>.

EILAM, E. *Reversing: secrets of reverse engineering*. [S.l.]: John Wiley & Sons, 2011.

FINKENZELLER, K. *RFID handbook: fundamentals and applications in contactless smart cards, radio frequency identification and near-field communication*. [S.l.]: John Wiley & Sons, 2010.

GAMPL, B.; GABRIEL, P.; WOLFRAM, G.; WALK, E.; GAUBY, A.; NEUBAUER, F.; LANGE, S.; CORDUANT, V.; MORTERA-MARTINEZ, C.; METZ, G.; PAVLIK, K. *The RFID Roadmap: The Next Steps for Europe*. [S.l.: s.n.], 2008. ISBN 978-3-540-71018-9.

GANS, G.; HOEPMAN, J.-H.; GARCIA, F. *A practical attack on the MIFARE classic*. [S.l.: s.n.], 2008. ISBN 978-3-540-85892-8.

GARCIA, F. D.; GANS, G. K.; MUIJRERS, R.; ROSSUM, P.; VERDULT, R.; SCHREUR, R. W.; JACOBS, B. Dismantling mifare classic. In: *Computer Security – ESORICS 2008*. [S.l.]: Springer Berlin Heidelberg, 2008.

GARCIA, F. D.; ROSSUM, P. V.; VERDULT, R.; SCHREUR, R. W. Wirelessly pickpocketing a mifare classic card. In: IEEE. *2009 30th IEEE Symposium on Security and Privacy*. [S.l.], 2009. p. 3–15.

GENTLE, J. E. *Random number generation and Monte Carlo methods*. [S.l.]: Springer, 2003. v. 381.

IDENTIFICATION cards. [S.l.], 2018. Disponível em: <https://www.iso.org/standard/73595.html>.

MEIJER, C.; VERDULT, R. Ciphertext-only cryptanalysis on hardened mifare classic cards. In: ACM. *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. [S.l.], 2015. p. 1212–1223.

MENEZES, A. J.; OORSCHOT, P. C. V.; VANSTONE, S. A. *Handbook of applied cryptography*. [S.l.]: CRC press, 2018.

MIGUELBALBOA. *RFID*. <https://github.com/miguelbalboa/rfid>. Repositório GitHub. Acesso em: 15 out. 2023.

MITROKOTSA, A.; RIEBACK, M. R.; TANENBAUM, A. S. Classifying rfid attacks and defenses. *Information Systems Frontiers*, v. 12, n. 5, p. 491–505, Nov 2010. ISSN 1572-9419. Disponível em: <https://doi.org/10.1007/s10796-009-9210-z>.

NOHL, K. Mifare, little security, despite obscurity. *the 24th Congress of the Chaos Computer Club in Berlin, December 2007*, 2007. Disponível em: <https://cir.nii.ac.jp/crid/1571135651252094848>.

SHIELDS, A.; CARTHY, U. M.; RIORDAN, D.; DOODY, P.; WALSH, J.; UYSAL, I. Radio frequency identification (rfid). In: ____. [s.n.], 2015. p. 1–14. ISBN 9780471346081. Disponível em: <https://doi.org/10.1002/047134608X.W8155>.

SOUZA, W. B. de. *Cartão MIFARE classic: ataques e medidas de contorno*. Dissertação (MSc Thesis) — Universidade de São Paulo, São Paulo, nov 2011.

STALLINGS, W. *Network and internetwork security: principles and practice*. [S.l.]: Prentice-Hall, Inc., 1995.