# AMAS: Anonimato e Autenticação Mútua em Sistemas RFID com Protocolos Anticolisão baseados em Árvore

**Bruno Gentilini D'Ambrosio, Paulo André da S. Gonçalves**

Centro de Informática (CIn)- Universidade Federal de Pernambuco (UFPE)
50.740-560 – Recife – PE – Brasil

{bgda,pasg}@cin.ufpe.br

***Abstract.*** *The most recent tag-reader mutual authentication schemes for RFID systems based on passive tags, SAMA and SEAS, are able to maintain anonymity of the tags. However, a minimum requirement is that the tag's real ID must never be transmitted in clear text during any message exchange with the reader(s). Moreover, the real ID is commonly used and transmitted in plain text during the execution of tree-based anticollision protocols. This execution precedes the authentication process, so that the anonymity of the tags can not be guaranteed. This paper proposes a scheme to be used with tree-based anticollision protocols that allows tag-reader mutual authentication and preserves the anonymity of the tags. The proposed scheme, namely AMAS (Anonymous Mutual Authentication Scheme), is designed with focus on systems that use passive tags, which have limited computing capabilities. The proposal introduces the use of random and temporary IDs since the execution of the tree-based anticollision protocol, not allowing an attacker to correlate these IDs with the real IDs.*

***Resumo.*** *Os esquemas mais atuais de autenticação mútua etiqueta-leitor para sistemas RFID baseados em etiquetas passivas, o SEAS e o SAMA, são capazes de manter o anonimato das etiquetas. Mas para isso, um requisito mínimo necessário é nunca transmitir em claro o ID real delas durante qualquer troca de mensagem com o(s) leitor(es). Por outro lado, o ID real é comumente utilizado e transmitido em claro durante a execução de protocolos anticolisão baseados em árvore. Essa execução precede o processo de autenticação, fazendo com que o anonimato das etiquetas não possa ser garantido. Este artigo propõe um esquema para ser utilizado com protocolos anticolisão baseados em árvore que permite a autenticação mútua etiqueta-leitor e preserva o anonimato das etiquetas. O esquema proposto, denominado AMAS (Anonymous Mutual Authentication Scheme), é projetado com foco em sistemas com etiquetas passivas, as quais possuem recursos computacionais limitados. A proposta introduz o uso de IDs aleatórios e temporários desde a execução do protocolo anticolisão baseado em árvore, não permitindo a um atacante correlacionar tais IDs com os IDs reais.*

## 1. Introdução

Em sistemas RFID (*Radio Frequency IDentification*), o controle de acesso das etiquetas ao meio é arbitrado pelo leitor através do uso de um protocolo anticolisão [Klair et al. 2010]. Nos protocolos anticolisão baseados em árvore, o leitor envia uma requisição às etiquetas. Cada etiqueta que satisfaz à requisição responde com seu identificador único (ID).

Quando duas ou mais etiquetas respondem ao mesmo tempo, ocorre uma colisão. O leitor faz, então, uma série de requisições, dividindo recursivamente essas etiquetas em subgrupos até que apenas uma etiqueta responda. O processo descrito pode ser representado por uma árvore. A raiz representa a população de etiquetas a ser identificada. Os nós intermediários representam subgrupos de etiquetas que colidiram ao responderem a uma mesma requisição do leitor. Cada folha representa a resposta de uma única etiqueta a uma requisição do leitor, permitindo sua identificação pelo ID, ou ainda, sua seleção para qualquer outra comunicação exclusiva com essa etiqueta e prevista pela aplicação, como por exemplo, um processo de autenticação mútua etiqueta-leitor [Klair et al. 2010].

Prover a autenticação mútua etiqueta-leitor é um dos maiores desafios em sistemas RFID baseados em etiquetas passivas devido à impossibilidade de se utilizar primitivas criptográficas como aquelas baseadas na função SHA-256 e no AES (*Advanced Encryption Standard*) [Feldhofer and Rechberger 2006]. Isso ocorre devido às limitações de recursos computacionais dessas etiquetas, as quais possuem no máximo 4.000 portas lógicas dedicadas aos mecanismos de segurança [Myneni et al. 2011, Misra et al. 2009]. Além disso, por limitações de tempo de processamento e de consumo de energia, a quantidade máxima de ciclos de relógio utilizada por tais mecanismos está limitada em 220 [Myneni et al. 2011, Misra et al. 2009].

Diversos esquemas buscam prover autenticação em sistemas RFID com etiquetas passivas [Peris-Lopez et al. 2006, Misra et al. 2009, Myneni et al. 2011]. Dentre eles, o SAMA (*Serverless Anonymous Mutual Authentication*) [Myneni et al. 2011] e o SEAS (*Secure and Efficient Anonymity Scheme*) [Misra et al. 2009] são os mais atuais para prover autenticação mútua etiqueta-leitor. Esses dois esquemas são capazes de manter o anonimato das etiquetas. Mas para isso, um requisito mínimo necessário é nunca transmitir em claro o ID real dessas etiquetas durante qualquer troca de mensagem com o(s) leitor(es). Por outro lado, o ID real das etiquetas é comumente utilizado e transmitido em claro durante a execução de protocolos anticolisão baseados em árvore. Essa execução precede o processo de autenticação, fazendo com que o anonimato das etiquetas não possa ser garantido.

Este artigo propõe um mecanismo para ser utilizado com protocolos anticolisão baseados em árvore que permite a autenticação mútua etiqueta-leitor e preserva o anonimato das etiquetas. O esquema proposto, denominado AMAS (*Anonymous Mutual Authentication Scheme*), é projetado com foco em sistemas que utilizam etiquetas passivas. Assim sendo, as limitações de recursos computacionais dessas etiquetas são levadas em consideração. A proposta introduz o uso de IDs aleatórios e temporários desde a execução do protocolo anticolisão baseado em árvore, não permitindo a um atacante correlacionar tais IDs com os IDs reais. Adicionalmente, este artigo provê uma análise da segurança do esquema proposto e avalia o seu custo em termos de quantidade de portas lógicas e de ciclos de relógio.

O restante deste artigo está organizado da seguinte forma: a Seção 2 apresenta os trabalhos relacionados. A Seção 3 descreve o modelo de sistema e o modelo de ameaça considerados para o desenvolvimento deste trabalho. A Seção 4 apresenta o esquema proposto de anonimato e autenticação mútua. Em seguida, a Seção 5 apresenta uma análise da segurança do esquema proposto e avalia o seu custo em termos de quantidade de portas lógicas e ciclos de relógio. Por fim, a Seção 6 apresenta a conclusão do trabalho.

## 2. Trabalhos Relacionados

Existem diversas propostas de mecanismos que lidam com requisitos de autenticação e anonimato em sistemas RFID. Essas propostas podem ser divididas em três classes distintas [Misra et al. 2009]: as baseadas em funções *hash* [Dimitriou 2005, Lee et al. 2005, Weis et al. 2004, Yang et al. 2005]; as baseadas em algoritmos criptográficos [Dimitriou 2006, Dominikus et al. 2005, Feldhofer et al. 2004, Feldhofer and Rechberger 2006] e; as baseadas em operações lógicas e bit a bit [Peris-Lopez et al. 2006, Misra et al. 2009, Myneni et al. 2011].

As propostas da primeira classe se baseiam em funções *hash* como: SHA-1, SHA-256, MD4 e MD5. Um problema das propostas dessa classe é a utilização de um número significativo de portas lógicas e de ciclos de relógio. Em particular, a função MD4 é a menos custosa entre as citadas e requer 7.350 portas lógicas e 456 ciclos de relógio para que possa ser implementada [Feldhofer and Rechberger 2006]. As propostas da segunda classe se baseiam em algoritmos criptográficos como o AES, o qual é capaz de prover um nível adequado de segurança ao sistema. As propostas dessa classe também consomem uma quantidade significativa de postas lógicas e ciclos de relógio. Em [Feldhofer and Rechberger 2006], por exemplo, a implementação mais otimizada do AES requer 3.400 portas lógicas e 1.032 ciclos de relógio.

Como dito anteriormente, as etiquetas passivas possuem no máximo 4.000 portas lógicas para uso dedicado dos mecanismos de segurança e a quantidade máxima de ciclos de relógio utilizada por tais mecanismos está limitada em 220 [Myneni et al. 2011, Misra et al. 2009]. Por causa disso, surgiram propostas baseadas em operações lógicas e bit a bit [Peris-Lopez et al. 2006, Misra et al. 2009, Myneni et al. 2011]. Essas propostas buscam prover anonimato e autenticação utilizando mecanismos menos custosos em termos de portas lógicas e ciclos de relógio para viabilidade de uso em etiquetas passivas.

O $M^2AP$ (*Minimalist Mutual-Authentication Protocol*) [Peris-Lopez et al. 2006] é um protocolo de autenticação mútua etiqueta-leitor que busca preservar o anonimato das etiquetas. Nesse protocolo, a etiqueta possui dois *IDs*: o primeiro é o ID real e o segundo é conhecido como IDS (*index-pseudonym*). Este último é um pseudônimo que funciona como um índice e permite ao leitor encontrar numa tabela no banco de dados onde estão armazenadas as informações da etiqueta correspondente.

O $M^2AP$ é realizado em duas fases: autenticação e atualização. Na fase de autenticação, o leitor e a etiqueta trocam duas mensagens. Essas mensagens são construídas de forma pré-estabelecida a partir da realização de operações bit a bit entre o IDS, o ID real da etiqueta, quatro chaves de segurança pré-compartilhadas e dois números aleatórios gerados pelo leitor. Na fase de atualização, a etiqueta e o leitor atualizam de forma sincronizada o IDS e as quatro chaves de segurança utilizadas na fase anterior. Isso objetiva manter o anonimato das etiquetas e impedir que um atacante identifique as chaves de segurança que serão utilizadas na próxima rodada de autenticação.

Em [Barasz 2007] é mostrado um ataque contra o $M^2AP$. Esse ataque permite a uma entidade maliciosa ter acesso às chaves de segurança utilizadas e ao ID real da etiqueta. Para isso, basta que o atacante consiga capturar poucas rodadas de trocas de mensagens entre uma etiqueta-alvo e o leitor durante a execução do $M^2AP$. O acesso a essas informações permite a um atacante clonar essa etiqueta.

Em [Misra et al. 2009] é apresentado um esquema de autenticação mútua denominado SEAS (*Secure and Efficient Anonymity Scheme*). Nesse esquema, cada etiqueta possui, além do ID real, um segredo exclusivo que é pré-compartilhado com um servidor conectado ao leitor. Cada leitor também possui um segredo pré-compartilhado com o servidor. Para construir as mensagens trocadas durante o processo de autenticação mútua, o SEAS faz operações bit a bit utilizando três números aleatórios, uma função de deslocamento circular à esquerda e os segredos pré-compartilhados. Essas mensagens são repassadas ao servidor, o qual as utiliza para autenticar tanto o leitor quanto a etiqueta. A proposta do SEAS assume que a etiqueta já tenha passado pelo processo anticolisão para iniciar o processo de autenticação e que seu ID real nunca seja transmitido em claro a fim de manter o anonimato da mesma.

Em [Myneni et al. 2011] é proposto o SAMA (*Serverless Anonymous Mutual Authentication*). O SAMA se propõe a oferecer um esquema de autenticação mútua etiqueta-leitor sem a necessidade de uso de um servidor conectado ao leitor. A vantagem alegada é a não necessidade de haver conexões persistentes entre o leitor e o servidor para a realização do processo de autenticação. Os contras da eliminação do servidor incluem: 1) o aumento do custo e do consumo de energia do leitor, já que o mesmo precisa armazenar toda a base de informações da aplicação e; 2) a necessidade de um mecanismo que sincronize a base de dados de todos os leitores do sistema.

O processo de autenticação do SAMA utiliza dois componentes: o NLFSR (*Non-Linear Feedback Shift Register*)[Dubrova et al. 2008] e uma função de perturbação. O NLFSR é um registrador que realiza deslocamentos e que possui uma função de transição composta por uma quantidade pré-definida de portas lógicas do tipo $XOR$ e $AND$. Essa função de transição faz com que cada NLFSR modifique a sua entrada de uma maneira única. Maiores detalhes sobre o NLFSR serão apresentados na Seção 4. A função de perturbação é um mecanismo auxiliar que visa manter o anonimato das mensagens transmitidas durante o processo de autenticação. No processo de autenticação da etiqueta perante o leitor, a etiqueta utiliza o NLFSR para gerar uma espécie de assinatura que é transmitida para o leitor. Essa assinatura muda toda vez que o processo de autenticação é executado uma vez que números pseudoaleatórios compõem a entrada para o NLFSR. A assinatura é utilizada pelo leitor para identificar e autenticar a etiqueta sem a necessidade de trasmissão do ID real. A autenticação do leitor perante a etiqueta é feita de forma análoga, passando o leitor a utilizar o NLFSR para gerar a assinatura a ser autenticada pela etiqueta. O SAMA não leva em consideração como é realizado processo anticolisão, mas requer o sigilo do ID real para manutenção do anonimato das etiquetas.

Este artigo se diferencia dos trabalhos relacionados por propor um esquema de autenticação mútua etiqueta-leitor para ser utilizado em conjunto com protocolos anticolisão baseados em árvore e ao mesmo tempo preservar o anonimato das etiquetas. O esquema proposto, denominado AMAS (*Anonymous Mutual Authentication Scheme*), reutiliza mecanismos empregados no processo de autenticação para que as etiquetas gerem IDs aleatórios e temporários a serem utilizados durante a execução do protocolo anticolisão baseado em árvore. Essa abordagem busca garantir o anonimato das etiquetas desde o processo de anticolisão, não permitindo a um atacante correlacionar os IDs aleatórios e temporários com os IDs reais. O AMAS também é projetado com foco em sistemas que utilizam etiquetas passivas.

## 3. Modelo do Sistema e Modelo de Ameaça

Esta seção detalha o modelo do sistema RFID adotado neste trabalho e o modelo de ameaça contra o esquema de autenticação e anonimato proposto.

### 3.1. Modelo do Sistema

O sistema RFID analisado nesse trabalho é composto por três componentes: um servidor $S$, leitores e etiquetas. As etiquetas são passivas e possuem memória necessária para armazenar as informações utilizadas pelo esquema proposto. Adicionalmente, as etiquetas possuem um gerador de números pseudoaleatórios. Toda comunicação entre o leitor e o servidor é feita através de uma conexão segura e inviolável. A comunicação entre leitor e etiquetas é realizada através de radiofrequência (RF) e pode ser facilmente capturada. O leitor utiliza um protocolo anticolisão baseado em árvore para controle de acesso das etiquetas ao meio de comunicação.

Cada etiqueta $T_i$ possui espaço de memória reservado para um identificador aleatório e temporário ($IDT_i$) de $n$ *bits*. Esse identificador substitui seu identificador real ($IDR_i$) de $n$ *bits* durante a execução do protocolo anticolisão baseado em árvore. Para a geração do $IDT_i$ e a realização do processo de autenticação, cada etiqueta possui uma chave atualizável ($K_i$) e um $NLFSR_i$ único, sendo ambos de $n$ *bits*.

### 3.2. Modelo de Ameaça

Um modelo de ameaça tem como objetivo definir o que uma entidade maliciosa pode fazer para tentar quebrar a segurança de um sistema. Neste artigo, é assumido que uma entidade maliciosa pode ameaçar o sistema RFID das seguintes formas:

1. Capturando quaisquer mensagens trocadas entre o leitor e as etiquetas durante a execução do protocolo de anticolisão;
2. Capturando quaisquer mensagens entre as etiquetas e o leitor durante o processo de autenticação;
3. Realizando ataques de *replay*;
4. Utilizando dados capturados para tentar obter informações sigilosas;
5. Realizando ataques de dessincronização.

Com a *ameaça 1*, o atacante tem acesso aos identificadores que estejam sendo utilizados pelas etiquetas durante a execução do protocolo anticolisão baseado em árvore. Essa ameaça também permite a um atacante que ele tente se passar por um leitor e realize o processo de anticolisão de etiquetas. Isso significa que o atacante pode obter os identificadores aleatórios e temporários utilizados pelas etiquetas a qualquer momento. A *ameaça 2* permite a um atacante tentar obter alguma informação relevante que o auxilie em algum outro tipo de ataque. As informações capturadas nas *ameaças 1* e *2* podem ser utilizadas para que um atacante tente rastrear ou mesmo clonar uma ou mais etiquetas do sistema.

A *ameaça 3* permite a uma entidade maliciosa reutilizar as mensagens capturadas nas *ameaças 1* e *2* para tentar se passar por uma etiqueta autêntica do sistema. A *ameaça 4* permite a um atacante utilizar as mensagens capturadas nas *ameaças 1* e *2* para tentar obter o identificador real, a chave atualizável ou o $NLFSR_i$ das etiquetas (vide Seção 3.1). Na *ameaça 5*, a entidade maliciosa pode bloquear a recepção de mensagens trocadas entre qualquer etiqueta e o leitor. Isso pode impedir a etiqueta de atualizar sua chave ($K_i$).

Existem ameaças inerentes às redes sem fio que não são tratadas neste artigo, como por exemplo: comprometer um dispositivo sem fio fisicamente; ataques de homem no meio (*man-in-the-middle*) e; outros tipos de negação de serviço (DoS - *Denial of Service*) além da dessincronização. Proteger um sistema de forma efetiva contra essas ameaças ainda é um desafio atualmente.

## 4. AMAS

.O AMAS (*Anonymous Mutual Authentication Scheme*) é um esquema para ser utilizado com protocolos anticolisão baseados em árvore. Ele é realizado em duas etapas: a primeira é responsável pela geração do $IDT_i$ e a segunda é reponsável pela autenticação mútua etiqueta-leitor. No AMAS, cada etiqueta possui um $NLFSR_i$ único pré-compartilhado com o servidor $S$. Esse NLFSR é utilizado no processo de geração do $IDT_i$ para garantir que apenas um leitor autorizado consiga identificar a etiqueta. O NLFSR também é utilizado no processo de autenticação mútua etiqueta-leitor, garantindo que tal processo só possa ser feito por etiquetas e leitores autorizados. Todas as operações realizadas durante a execução do AMAS utilizam números de $n$ bits, onde $n$ é um múltiplo de 8.

O esquema de autenticação mútua etiqueta-leitor do AMAS é baseado no do SAMA e suas diferenças são discutidas nesta seção. A seguir, são apresentados: uma descrição detalhada do funcionamento do NLFSR, o processo de geração do $IDT_i$, o processo de autenticação mútua etiqueta-leitor e, por fim,as diferenças entre o AMAS, o SEAS e o SAMA.

### 4.1. NLFSR

O NLFSR [Dubrova et al. 2008] é um registrador de $x$ *bits* que realiza $y$ rodadas de deslocamento à direita, onde $y$ é o tamanho da saída do NLFSR. Cada NLFSR possui uma função de transição única composta por uma quantidade pré-definida de portas lógicas do tipo $XOR$ ou $AND$. Cada uma das portas lógicas pode receber como entrada a valor do *bit* que se encontra em uma das posições do registrador ou mesmo a saída de outra porta lógica. Antes de cada rodada de deslocamento, o NLFSR substitui o valor de cada *bit* do registrador pelo resultado de um $XOR$ entre o valor atual do *bit* e o resultado da função de transição. A cada rodada de deslocamento, o *bit* mais a direita do registrador é concatenado aos *bits* de saída do NLFSR. É importante ressaltar que quaisquer modificações realizadas na função de transição como, por exemplo, a alteração no número de portas lógicas, faz com que a saída do NLFSR seja completamente modificada. A Figura 1 apresenta um exemplo de *NLFSR* de 32 *bits* que possui uma função de transição composta por sete portas lógicas, sendo quatro do tipo $XOR$ e três do tipo $AND$.

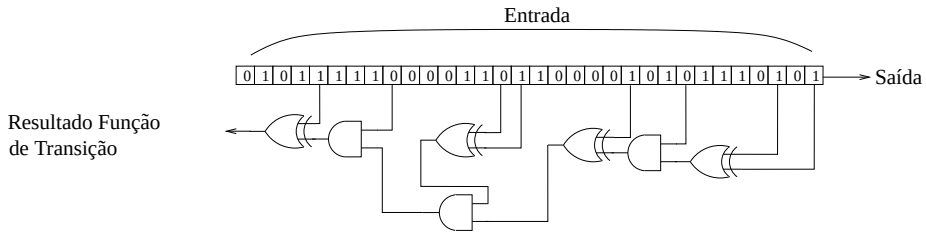


**Figura 1. Exemplo de NLFSR de 32 bits com sete portas lógicas.**

### 4.2. Geração do $IDT_i$

A Figura 2 apresenta o processo de geração do $IDT_i$ realizado por cada uma das etiquetas a serem identificadas. O leitor $R_j$ precisa que as etiquetas gerem os seus respectivos $IDT_i$ antes do início da execução do protocolo anticolisão. Para isso, ele gera um número aleatório $r_j^1$ e o transmite em *broadcast* para as etiquetas. Cada uma das etiquetas recebe $r_j^1$ e o utiliza como entrada para o seu respectivo $NLFSR_i$, passando por uma sequência de $n$ deslocamentos. A saída desse processo é um número $r_j^1$. Em seguida, cada etiqueta gera um número aleatório $r_{ij}^2$ e faz um XOR entre $r_j^1$, $r_{ij}^2$, o seu identificador real $IDR_i$ e sua chave $K_i$, gerando o número $p_{ij}^1$. Esse número é fornecido como entrada para o $NLFSR_i$ que, após os $n$ deslocamentos, gera o $IDT_i$.

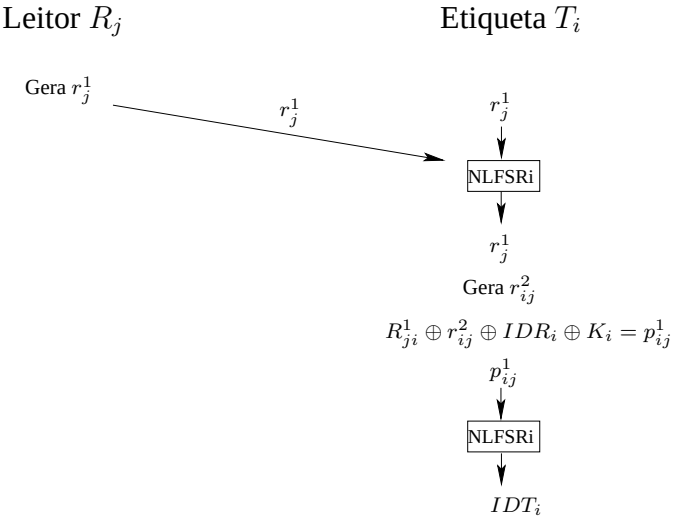


**Figura 2. Geração do $IDT_i$ por uma etiqueta $T_i$.**

Após o fim da primeira etapa do AMAS, o leitor executa o protocolo anticolisão. É importante observar que existe a possibilidade de que duas etiquetas gerem o mesmo $IDT_i$. No entanto, dado que até 500 etiquetas com IDs de 32 *bits* estejam sendo identificadas ao mesmo tempo, a probabilidade de que pelo menos duas gerem dois $IDT_i$ iguais é de aproximadamente $2,91 \times 10^{-5}$ conforme o paradoxo do aniversário. O procedimento a ser adotado pelo protocolo anticolisão para tratar esse problema está fora do escopo deste artigo.

### 4.3. Autenticação da Etiqueta perante o Leitor

A Figura 3 apresenta o processo de autenticação de uma etiqueta $T_i$ perante o leitor $R_j$. O processo se inicia assim que a mesma é selecionada durante a execução do protocolo anticolisão baseado em árvore. O leitor $R_j$ armazena o $IDT_i$ da etiqueta e inicia a verificação da autenticidade da mesma. Para isso, o leitor requisita o $r_{ij}^2$ gerado pela etiqueta durante o processo de geração de seu $IDT_i$. Após o recebimento de $r_{ij}^2$, o leitor envia $IDT_i$, $r_j^1$ e $r_{ji}^2$ para o servidor $S$. O servidor busca no seu banco de dados uma terna ($IDR_i$, $K_i$, $NLFSR_i$) que consiga gerar o $IDT_i$ quando combinada com $r_j^1$ e $r_{ij}^2$. Em outras palavras, o servidor verifica um a um os dados das etiquetas presentes no banco de dados até que a terna correta seja encontrada. Quando isso acontece, ele envia a terna para o leitor que, por sua vez, autentica a etiqueta. Caso nenhuma terna correspondente seja encontrada, a etiqueta é considerada falsa e a comunicação com essa etiqueta é encerrada.

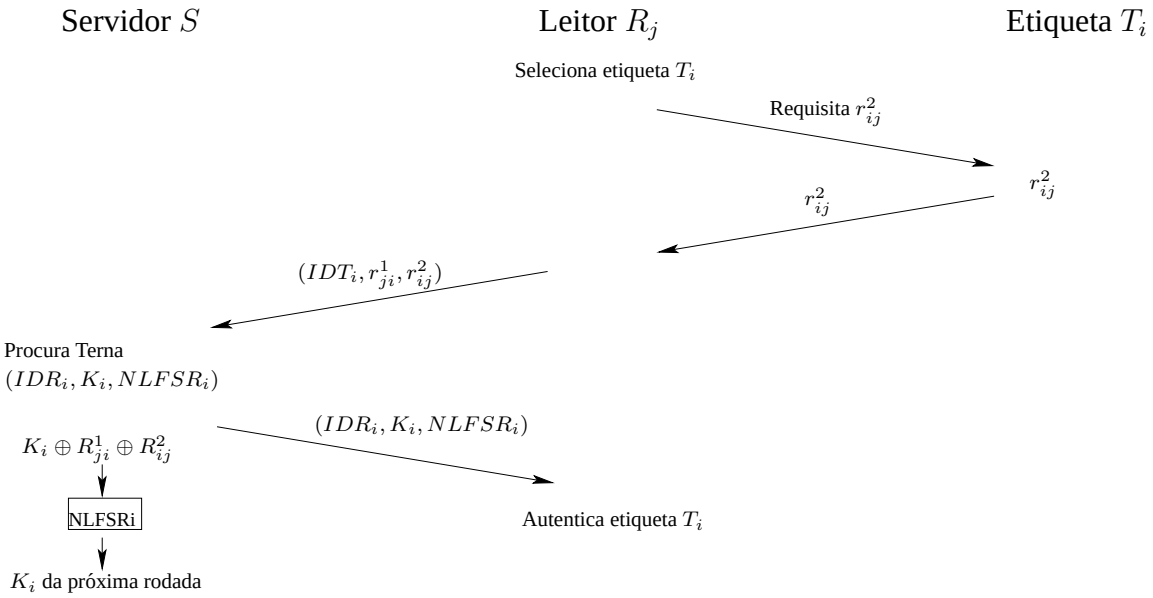


**Figura 3. Autenticação bem sucedida de uma etiqueta $T_i$ perante o leitor $R_j$.**

Para garantir uma maior segurança do mecanismo de geração do $IDT_i$, a chave $K_i$ é atualizada pelo servidor após a autenticação da etiqueta. Para isso, é feito um XOR entre $K_i$, $r_j^1$ e $R_{ij}^2$. Este último é o valor de $r_{ij}^2$ após ele ser modificado pelo $NLFSR_i$. O resultado do XOR é fornecido como entrada para o $NLFSR_i$, o qual gera como saída o valor de $K_i$ que será utilizado na próxima geração do $IDT_i$. O servidor guarda o valor antigo de $K_i$ para evitar que ataques de dessincronização possam afetar o sistema RFID.

### 4.4. Autenticação do Leitor perante à Etiqueta

A Figura 4 apresenta o processo de autenticação do leitor $R_j$ por uma etiqueta $T_i$. Para isso, o leitor fornece $r_{ij}^2$ como entrada para o $NLFSR_i$, o qual gera a saída $R_{ij}^2$. Em seguida, o leitor gera um novo número aleatório $r_{ji}^3$ e faz um XOR desse número com $R_{ji}^2$, gerando $x_{ji}^1$. Este último número é fornecido como entrada ao $NLFSR_i$, o qual gera a saída $m_{ji}^1$. Os valores de $m_{ji}^1$ e $r_{ji}^3$ são concatenados, gerando a mensagem $M_{ji}^1$. Essa mensagem é enviada para a etiqueta que verifica a sua validade utilizando $r_{ij}^2$, $r_{ji}^3$ e o $NLFSR_i$. Caso $M_{ji}^1$ seja válida, o processo de autenticação do leitor é finalizado.

Após autenticar o leitor, a chave $K_i$ é atualizada na etiqueta da mesma forma que foi feita no servidor $S$. Em seguida, $T_i$ faz um XOR entre $R_{ij}^2$, $r_j^1$ e $r_j^3 i$ e fornece o resultado dessa operação como entrada para o $NLFSR_i$, gerando como saída o número $m_{ij}^2$. Este último número é enviado para o leitor e serve como confirmação para o servidor de que o processo foi concluído corretamente. Após o recebimento, o servidor utiliza $R_{ij}^2$, $r_j^1$ e $r_j^3 i$ para verificar o valor de $m_{ij}^2$, verificando se foi realmente a etiqueta que enviou o número.

### 4.5. Diferenças entre o AMAS, o SEAS e o SAMA

O AMAS e o SEAS são esquemas diferentes. O AMAS utiliza uma chave de segurança atualizável para cada etiqueta e baseia sua segurança no NLFSR. Já o SEAS utiliza uma chave de segurança fixa para cada etiqueta e baseia sua segurança apenas em operações

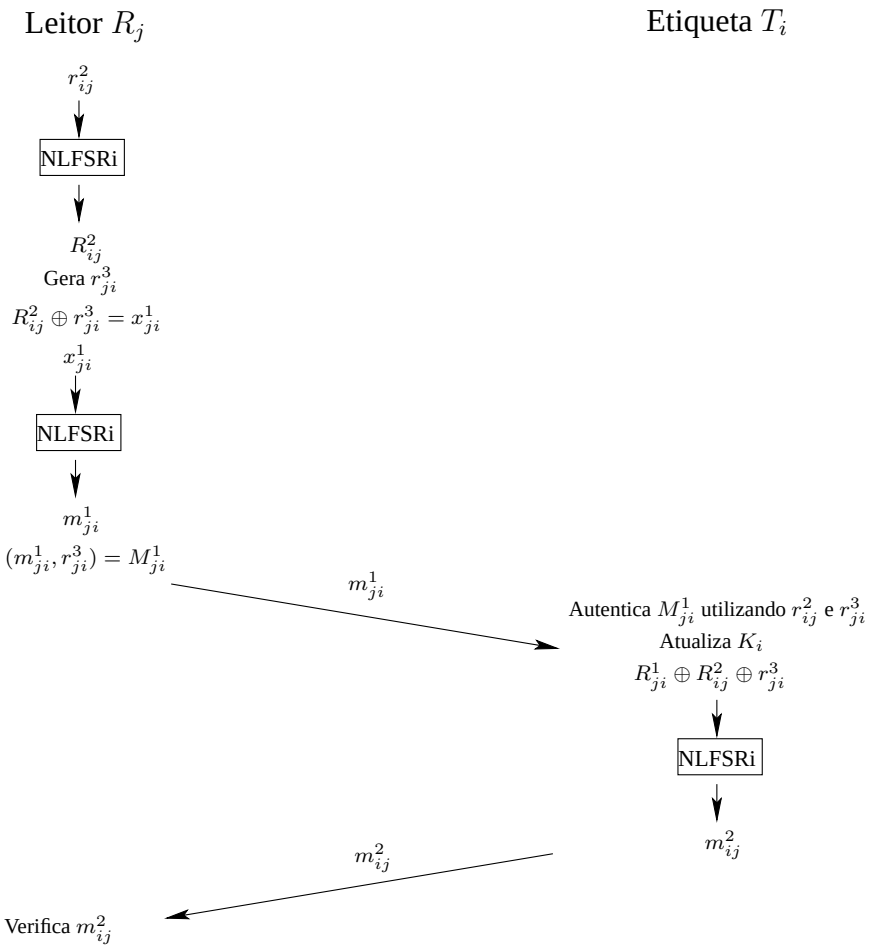


**Figura 4. Autenticação bem sucedida de um leitor $R_j$ perante uma etiqueta $T_i$.**

XOR e em uma função linear de deslocamento à esquerda. O SEAS não leva em conta como o processo anticolisão é feito nem as vulnerabilidades desse processo.

O esquema de autenticação do AMAS é uma versão modificada daquele usado pelo SAMA [Myneni et al. 2011]. Primeiramente, o esquema utilizado pelo SAMA se difere pela não utilização de um servidor no processo de autenticação, fazendo com que cada leitor tenha que armazenar os dados de todas as etiquetas do sistema para realizar tal processo. No AMAS, o servidor é utilizado para armazenar e buscar todas as informações relativas às etiquetas. Isso reduz significativamente a quantidade de memória que cada leitor precisa ter disponível para realizar o processo de autenticação.

A segunda diferença entre o AMAS e o SAMA ocorre no processo de autenticação da etiqueta perante o leitor. No AMAS, esse processo inclui a utilização do $IDT_i$, do ID real da etiqueta ($IDR_i$) e da chave atualizável ($K_i$). Essa modificação permite que o $IDT_i$ seja utilizado tanto pelo processo anticolisão baseado em árvore como na autenticação da etiqueta perante o leitor, recebendo a segurança adicional fornecida pela chave atualizável ($K_i$).

Outra diferença entre o AMAS e o SAMA é a não utilização da função de perturbação pelo AMAS, sendo substituída, quando necessário, pelo $NLFSR_i$. Isso é feito porque a função de perturbação pode ser invertida facilmente. Assim sendo,

a utilização dessa função não traria nenhum ganho para a segurança do esquema de autenticação do AMAS e resultaria em um consumo desnecessário de recursos para sua implementação e execução. Assim como o SEAS, o SAMA não leva em conta como o processo anticolisão é realizado nem as fraquezas desse processo.

## 5. Análise da Segurança e do Custo do AMAS

Essa seção apresenta, inicialmente, uma criptoanálise do NLFSR e uma análise da segurança do AMAS perante as ameaças definidas na Seção 3. Em seguida, é apresentada uma avaliação de seu custo em termos de quantidade de portas lógicas e ciclos de relógio, comparando-os com os quantitativos obtidos pelo SEAS, SAMA, SHA-1, MD4 e AES.

### 5.1. Análise da Segurança

#### 5.1.1. Criptoanálise do NLFSR

O $NLFSR$ é o dispositivo de segurança básico utilizado pelo AMAS. Assim sendo, um atacante precisa primeiramente identificar o $NLFSR_i$ da etiqueta $T_i$ que ele pretende atacar para conseguir burlar o AMAS. No entanto, de acordo com o corolário apresentado e provado em [Myneni et al. 2011], a probabilidade de um atacante conseguir identificar um $NLFSR$ de 32 bits com 9 portas lógicas é igual a $\frac{1}{6{,}36X10^{25}}$, sendo um valor desprezível. A partir desse valor podemos afirmar que um atacante não consegue identificar que uma entrada $x$ foi utilizada pelo $NLFSR$ para gerar uma saída $x'$.

#### 5.1.2. Rastreamento

Uma etiqueta $T_i$ não é rastreável se um atacante não conseguir o seguinte:

- **caso 1 -** correlacionar seus identificadores aleatórios e temporários, gerados em rodadas distintas, como pertencentes à etiqueta $T_i$.
- **caso 2 -** relacionar seu identificador aleatório e temporário atual com o seu identificador real;

Em ambos os casos, um atacante poderia tentar rastrear uma etiqueta $T_i$ utilizando dois métodos. No primeiro, o atacante apenas escuta mensagens trocadas entre a etiqueta $T_i$ e o leitor. No segundo, o atacante tenta se passar por um leitor legítimo e iniciar o processo anticolisão posteriormente ao envio por ele de um $r_j^1$ para que a etiqueta gere seu $IDT_i$.

O primeiro método não funciona no **caso 1** pelo seguinte: um atacante precisa identificar o $NLFSR_i$ e a chave atualizável ($K_i$) da etiqueta $T_i$ que foram utilizados no processo de geração de dois ou mais identificadores aleatórios e temporários. Como apresentado na Seção 5.1.1, a probabilidade dele identificar o $NLFSR_i$ é desprezível. Além disso, a chave $K_i$ é mantida em sigilo durante todo o processo e é atualizada a cada rodada de autenticação.

O segundo método não funciona no **caso 1** pelo seguinte: as etiquetas utilizam um número aleatório $r_{ij}^2$ para gerar o $IDT_i$. Toda vez que o processo anticolisão for rodado, a etiqueta gerará um novo número aleatório $r_{ij}^2$ e, consequentemente, um novo $IDT_i$. Para conseguir identificar que dois ou mais identificadores $IDT_i$ foram gerados por uma

mesma etiqueta $T_i$, um atacante precisaria obter o seu $NLFSR_i$ e sua chave atualizável ($K_i$), os quais ele não tem acesso.

Ambos os métodos não funcionam para o **caso 2**. Isso ocorre porque o atacante precisa obter o $NLFSR_i$ e a chave atualizável ($K_i$) da etiqueta $T_i$, os quais ele não tem acesso, para conseguir relacionar o $IDT_i$ atualmente utilizado pela etiqueta com o seu identificador real.

### 5.1.3. Clonagem

Para que um atacante consiga clonar corretamente uma etiqueta, ele precisa conseguir burlar o mecanismo de autenticação do AMAS. Para isso, a etiqueta falsa $T_i'$ teria que conseguir gerar corretamente o $IDT_i$ a partir de um $r_j^1$ enviado pelo leitor. A única forma de gerar $IDT_i$ corretamente é utilizando o $NLFSR_i$ e a chave atualizável $K_i$ da etiqueta $T_i$. Assim sendo, $T_i'$ não consegue burlar esse mecanismo, pois ela não consegue identificar qual é o $NLFSR_i$ utilizado por $T_i$ e não tem acesso à chave $K_i$. É importante observar que se um atacante conseguir acessar fisicamente uma etiqueta, ele terá acesso a todos os segredos contidos dentro da mesma e poderá cloná-la com facilidade. No entanto, essa ameaça não foi considerada no modelo de ataque pelo motivo já citado na Seção 3.

### 5.1.4. Ataque de Replay

Mesmo que um atacante consiga capturar diversas mensagens de autenticação transmitidas entre um leitor e uma etiqueta, ele não conseguirá reutilizá-las posteriormente em um ataque de *replay*. Isso ocorre devido à utilização do $IDT_i$ e dos números $r_j^1$, $r_{ij}^2$ e $r_{ji}^3$, os quais são gerados aleatoriamente a cada rodada da autenticação.

### 5.1.5. Acesso a Informações Sigilosas

As únicas informações sigilosas mantidas pelo AMAS são a chave $K_i$, o ID real $IDR_i$ e o $NLFSR_i$. No entanto, o $NLFSR_i$ nunca é transmitido e a chave $K_i$ e o $IDR_i$ não são transmitidos sem terem sido combinados com outras variáveis e modificados pelo $NLFSR_i$. Portanto, um atacante não consegue obtê-los somente capturando mensagens trocadas entre a etiqueta e o leitor. Mesmo que ele conseguisse adivinhar a chave $K_i$ de uma determinada rodada por força bruta, a mesma já não teria mais valor na rodada seguinte devido à atualização da chave no processo de autenticação.

### 5.1.6. Ataques de Dessincronização

O único parâmetro atualizado pelo AMAS a cada rodada é a chave $K_i$. Ele consegue garantir que um ataque de dessincronização simples não seja capaz de afetar a atualização sincronizada desse parâmetro pela etiqueta e pelo servidor. Isso ocorre porque o servidor utiliza o número $m_{ij}^2$ como garantia de que uma etiqueta $T_i$ tenha conseguido atualizar $K_i$ corretamente. Se o atacante conseguir evitar que o processo seja concluído com algum ataque de dessincronização mais elaborado, isso não afetará a geração do $IDT_i$ e o

processo de autenticação mútua etiqueta-leitor. Isso ocorre porque o servidor armazena para cada etiqueta do sistema, tanto a $K_i$ atualizada quanto o valor de $K_i$ utilizado na rodada anterior do processo autenticação. Dessa forma, ele garante que o leitor consegue autenticar a etiqueta em uma certa rodada mesmo que a chave $K_i$ não seja atualizada corretamente na rodada anterior. A aleatoriedade de $IDT_i$ também não é afetada por conta da utilização de um novo $r_{ij}^2$ a cada vez que ele é gerado.

### 5.2. Avaliação de Custos

Esta seção avalia os custos do AMAS em termos de quantidade de portas lógicas e ciclos de relógio, comparando-os com os custos do SEAS, do SAMA, da SHA-1, da MD4 e do AES. A comparação com a SHA-1 e a MD4 foi realizada por elas serem as funções *hash* que possuem o menor custo computacional em termos de portas lógicas e ciclos de relógio. O AMAS foi implementado como uma máquina de estados combinacional na linguagem de *hardware System Verilog*. Essa implementação foi utilizada para calcular seus custos através do programa de síntese de *hardware ALTERA Quartus II*. A implementação utilizou $n$ igual a 32, ou seja, todos os parâmetros e mecanismos utilizados foram de 32 bits. Isso foi feito para que fosse possível comparar o AMAS com os trabalhos relacionados, os quais também utilizam 32 bits.

O SAMA também foi implementado no ambiente utilizado para a implementação do AMAS. Os custos obtidos para o SAMA através dessa implementação foram semelhantes aos obtidos em [Myneni et al. 2011]. A versão da implementação do SAMA no *ALTERA Quartus II* utiliza 1.420 portas lógicas e 70 ciclos de relógio, ao passo que, os autores do SAMA informam que o esquema utiliza 1.393 portas lógicas e 70 ciclos de relógio. A pequena diferença se deve ao fato dos resultados serem dependentes da forma como o algoritmo é codificado. Para fins de comparação, este artigo adota os custos do SAMA obtidos no ambiente *ALTERA Quartus II*. Os custos da SHA-1, da MD4 e do AES foram obtidos em [Feldhofer and Rechberger 2006] enquanto os custos do SEAS foram obtido em [Misra et al. 2009]. Esses custos devem ser vistos como uma aproximação para comparação com o SAMA e o AMAS. A implementação no mesmo ambiente do AMAS não traria potencialmente mudanças significativas a ponto de alterar as conclusões deste trabalho.

A Figura 5(a) apresenta uma comparação do número necessário de portas lógicas para a implementação dos esquemas estudados. Note que a quantidade de portas lógicas do AMAS (1214 portas lógicas) é a menor de todas. A diferença de portas lógicas entre o AMAS e o SAMA ocorre por dois motivos. O primeiro é a não utilização pelo AMAS da função de perturbação do SAMA, a qual utiliza aproximadamente 300 portas lógicas em sua implementação. O segundo é fato de que os mecanismos de geração do $IDT_i$ e de atualização da chave $K_i$ do AMAS reutilizam o mesmo $NLFSR$ do esquema de autenticação para prover IDs para o processo anticolisão. Com a reutilização, esses mecanismos contribuem na quantidade adicional de aproximadamente 100 portas lógicas para serem implementados.

A Figura 5(b) apresenta o resultado de ciclos de relógio para os esquemas estudados. O custo do AMAS (150 ciclos) ficou abaixo do limite de 220 ciclos para etiquetas passivas embora tenha sido mais elevado do que o cilco de relógio do SAMA (70 ciclos) e do SEAS (44 ciclos). A necessidade de se utilizar mais ciclo de relógio no AMAS em

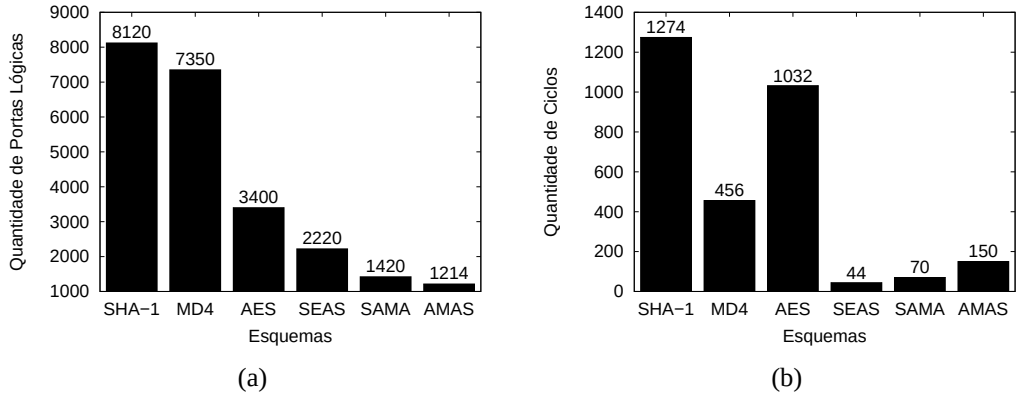


**Figura 5. Comparação entre os diversos esquemas.**

relação ao SAMA se deve ao mecanismo de geração do $IDT_i$ e ao uso, por mais vezes, do $NLFSR$, gerando o consumo de 80 ciclos de relógio a mais do que o SAMA. No entanto, quando comparado com os mecanismos tradicionais de segurança como o AES (1032 ciclos), o AMAS possui um custo significativamente menor.

## 6. Considerações Finais

Prover autenticação mútua é um dos maiores desafios de sistemas RFID que utilizam etiquetas passivas. Isso ocorre devido às limitações de recursos computacionais e memória inerentes a esse tipo de etiqueta. O SAMA e o SEAS são as propostas mais atuais para prover autenticação mútua etiqueta-leitor em sistemas RFID baseados em etiquetas passivas. Esses dois esquemas são capazes de manter o anonimato das etiquetas. Mas para isso, um requisito mínimo necessário é nunca transmitir em claro o ID real dessas etiquetas durante qualquer troca de mensagem com o(s) leitor(es). Por outro lado, o ID real das etiquetas é comumente utilizado e transmitido em claro durante a execução de protocolos anticolisão baseados em árvore. Essa execução precede o processo de autenticação, fazendo com que o anonimato das etiquetas não possa ser garantido.

Este artigo propôs um esquema de autenticação mútua etiqueta-leitor para ser utilizado em conjunto com protocolos anticolisão baseados em árvore e ao mesmo tempo preservar o anonimato das etiquetas. Essa é uma das principais contribuições deste artigo. O esquema proposto, denominado AMAS, reutilizou mecanismos empregados no processo de autenticação para que as etiquetas gerassem IDs aleatórios e temporários a serem utilizados durante a execução do protocolo anticolisão baseado em árvore. Essa abordagem buscou garantir o anonimato das etiquetas desde o processo de anticolisão, não permitindo a um atacante correlacionar os IDs aleatórios e temporários com os IDs reais.

Este artigo também apresentou uma análise da segurança do AMAS, demonstrando que o esquema consegue defender o sistema de todos os ataques presentes no modelo de ameaça adotado. Adicionalmente, os custos do AMAS, em termos de quantidade de portas lógicas e ciclos de relógio, foram avaliados. Os resultados demonstraram que o AMAS atende aos requisitos necessários para que ele possa ser utilizado em sistemas RFID baseados em etiquetas passivas.

## Referências


Barasz, M. (2007). Passive Attack Against the M2AP Mutual Authentication Protocol for RFID Tags. In *Proceedings of the First Int'l Workshop RFID Technology (EURASIP)*.

Dimitriou, T. (2005). A Lightweight RFID Protocol to Protect Against Traceability and Cloning Attacks. In *Proceedings of the First International Conference on Security and Privacy for Emerging Areas in Communications Networks (SecureComm)*, pages 59–66.

Dimitriou, T. (2006). A Secure and Efficient RFID Protocol that could make Big Brother (partially) Obsolete. In *Proceedings of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications (PERCOM)*, pages 269–275.

Dominikus, S., Oswald, E., and Feldhofer, M. (2005). Symmetric Authentication for RFID Systems in Practice. In *Proceedings of the Ecrypt Workshop on RFID and Lightweight Cryptography*, pages 14–15.

Dubrova, E., Teslenko, M., and Tenhunen, H. (2008). On Analysis and Synthesis of (n, k)-Non-linear Feedback Shift Registers. In *Proceedings of the Conference on Design, Automation and Test in Europe (DATE)*, pages 1286–1291.

Feldhofer, M., Dominikus, S., and Wolkerstorfer, J. (2004). Strong Authentication for RFID Systems Using the AES Algorithm. *Lecture Notes In Computer Science*, pages 357–370.

Feldhofer, M. and Rechberger, C. (2006). A Case Against Currently Used Hash Functions in RFID Protocols. *Lecture Notes In Computer Science*, 4278:372–381.

Klair, D., Chin, K.-W., and Raad, R. (2010). A Survey and Tutorial of RFID Anti-Collision Protocols. *IEEE Communications Surveys Tutorials*, 12(3):400–421.

Lee, S. et al. (2005). Efficient Authentication for Low-Cost RFID Systems. In *Proceedings of the International Conference on Computational Science and its Applications (ICCSA)*, pages 619–627.

Misra, S. et al. (2009). SEAS: A Secure and Efficient Anonymity Scheme for Low-Cost RFID tags. In *Proceedings of the IEEE International Conference on Communications (ICC)*, pages 1–6.

Myneni, S., Misra, S., and Xue, G. (2011). SAMA: Serverless Anonymous Mutual Authentication for Low-Cost RFID Tags. In *Proceedings of the IEEE International Conference on Communications (ICC)*, pages 1–5.

Peris-Lopez, P. et al. (2006). M2AP: A Minimalist Mutual-Authentication Protocol for low-cost RFID tags. In *Proceedings of the Third Int'l Conf. Ubiquitous Intelligence and Computing (UIC)*.

Weis, S. et al. (2004). Security and Privacy Aspects of Low-Cost Radio Frequency Identification Systems. *Lecture Notes In Computer Science*, 2802:201–212.

Yang, J. et al. (2005). Mutual Authentication for Low-Cost RFID Systems. In *Proceedings of the Ecrypt Workshop on RFID and Lightweight Cryptography*, pages 17–24.