Análise de Protocolos de Autenticação PUF para IoT

Universidade Lusófona

Mestrado de Engenheira Informática e Sistemas de Informação

João Vás e Lima

Nº21502526

Abstract

In my master's thesis I had the idea of testing authentication through Physically Unclonable Function (PUF). However, several authors have demonstrated that this authentication has several weaknesses, especially since it started testing attacks with machine learning. Other authors, however, have developed approaches with protocols to circumvent these attacks. From these protocols I am investigating which are possible to test in a controlled environment and with the means I can acquire, to select some to test and create the question that my thesis will try to answer.

Introdução

Uma Physically Unclonable Function (PUF), ou seja, uma função incorporada em uma estrutura física, contém informações aleatórias e únicas que se originam de variações incontroláveis do processo durante a fabricação em circuitos integrados (CI).

A ideia básica é usar essa "impressão digital" para servir como âncora de segurança em várias aplicações. O uso de PUFs permite o design de aplicações criptográficos sem armazenar informações confidenciais, como chaves na memória.

Para serem práticos, os PUFs devem ser fáceis de avaliar porque é extremamente difícil criar um clone de hardware, um modelo matemático do comportamento da estrutura ou um programa de software que pode calcular a resposta a um desafio em um período de tempo razoável.

Os circuitos PUF recebem uma sequência de bits (supostos desafios) como entrada e geram uma sequência de bits (chamadas respostas) como saída.

Dois chips não geram respostas idênticas para um desafio específico. A combinação de um desafio e sua resposta correspondente é chamada de Challenge Response Pair (CRP).

Os circuitos PUF são implementados através de diferentes tecnologias CMOS, normalmente num chip de memória, como um Application-Specific Integrated Circuit (ASIC) ou em um FPGA. Existem dois tipos de circuitos PUF, PUFs Weak e Strong. Para PUFs Strong, aumentar o tamanho do circuito PUF leva a um crescimento exponencial no número de CRPs. Para PUFs Weaks, aumenta linearmente.

Devido a variações de tempo, temperatura e tensão, alguns bits tendem a variar.

São usados fuzzy extractors para realizar correções para que as trocas de bits existentes sejam corrigidas.

Fuzzy extractors são uma ferramenta biométrica que permite a autenticação do usuário, através de um modelo biométrico construído a partir dos dados biométricos do utilizador como chave.

Eles extraem uma sequência uniforme e aleatória de uma entrada com tolerância a ruído (noise).

Além da característica noisy dos PUFs, também os efeitos tempo de vida devem ser levado em consideração no desenvolvimento de soluções baseadas em PUF. Sabe-se que é provável que o comportamento de resposta de uma instanciação PUF se altere levemente ao longo de sua vida útil. Portanto, os níveis de noisy aumentam com o tempo na ausência de protocolos "anti-envelhecimento".

Protocolos de autenticação PUF para IoT

A seguir, apresento uma visão geral dos protocolos existentes, levando em consideração os diferentes recursos que eles suportam. Foco principalmente nos protocolos compatíveis com IoT porque é o tipo de device que escolhi para realizar os testes para a dissertação.

Por outro lado, os protocolos geralmente não são independentes do tipo de PUF (weak ou strong) que eles usam. Protocolos baseados em PUFs weak normalmente requerem métodos criptográficos (por exemplo, hash, criptografia, etc.) para compensar a escassez de CRP. Isso nega a principal a vantagem de usar um PUF no IoT. Por isto concentrei-me em investigar protocolos para PUF strong.

Funcionalidades essenciais nos Protocolos de autenticação

1) Rubustez contra ataques de machine learning

Os primeiros protocolos de autenticação de um PUF (strong) eram baseado na suposição de que um PUF nunca pode ser clonado ou modelado. No entanto, várias pesquisas mostram que essa suposição é incorreta. Especialmente quando confrontados com análises de CRPs feitas por machine learning.

2) Lidar com PUFs parcialmente instáveis

Ao contrário das definições iniciais de PUF, as respostas geralmente não são 100% estáveis e os protocolos precisam de métodos de correção de erros para alinhar os resultados

3) Autenticação mútua, nos dois sentidos

Ambos os elementos da comunicação devem ser autenticados um contra o outro. Caso contrário, por exemplo, os sensores de IoT podem enviar dados pessoais sobre utilizadores a servidores não autenticados, e os servidores podem aceitar medições falsas de sensores de atacantes (não autenticados).

Como os dispositivos IoT normalmente comunicam-se diretamente, o ideal é que a autenticação mútua seja fornecida, não apenas entre um dispositivo e um servidor, mas também entre dois dispositivos.

Protocolos PUF

Early Protocols (Protocolos inicias)

Nenhum dos primeiros protocolos foi projetado especificamente para uso na IoT.

Uma vez conhecidos os ataques de Machine Learning, vários autores desenvolveram abordagens para proteger os PUF contra eles

Todos esses protocolos seguem o mesmo processo: depois do dispositivo receber um challange e gerar uma resposta, os protocolos aplicam criptografia, como algoritmos de hash ou algoritmos encriptados na response e enviam de volta ao servidor.

Alguns protocolos usam Nonvolatile Memory (NVM) nestes processos]. No entanto, implementar até mesmo um algoritmo criptográfico simples, adiciona custos indiretos de hardware, e o NVM contradiz uma das motivações originais dos PUFs, não armazenar segredos no dispositivo.

Isso é especialmente problemático para os sistemas de IoT, uma vez que têm poucos recursos e são vulneráveis a ataques físicos.

Mutual Authentication Protocol

Protocolo de autenticação baseado em PUF fornece autenticação mútua entre dispositivos na IoT abrange dois cenários de autenticação: autenticação mútua dispositivo-servidor e autenticação mútua dispositivo-dispositivo. O cenário de autenticação do dispositivo-servidor é o seguinte: cada dispositivo IoT tem um ID.

Primeiro, o dispositivo envia seu ID (vamos assumir o IDA) mais um random nonce N1 para o servidor. O servidor seleciona um CRP (Ci, Ri) e gera um número aleatório Rs. Em seguida, o servidor gera uma mensagem criptografada MA = (IDA, N1, Rs) Ri.

Em seguida, envia esta mensagem MA juntamente com o desafio Ci e um código de autenticação de mensagem MACA para o dispositivo IoT. O dispositivo IoT aplica Ci ao PUF local para regenerar Ri, desencripta a mensagem MA com ele e verifica a autenticidade, integridade e atualização da mensagem com o MAC fornecido.

De seguida, o dispositivo IoT gera uma nova CRP (Ci + 1, Ri + 1) através do novo desafio (Ci + 1 = H (Rs'|NA), sendo o H a função de distância de Hamming e NA outro número aleatório gerado. O dispositivo gera outra vez uma mensagem criptografada Ms = (IDA, NA, Ri + 1) Ri. Ms e a envia com o seu MACs para o servidor.

O servidor verifica a mensagem se é recente com o MAC, regenera Ci + 1 e verifica se o Ri + 1 está correto. O protocolo de autenticação mútua dispositivo-dispositivo é semelhante a esta abordagem e usa o servidor para autenticar os dois dispositivos.

A principal vantagem deste protocolo é o recurso de autenticação mútua e a possibilidade de autenticação dispositivo a dispositivo, o que é muito útil para muitos aplicativos de IoT.

Obfuscated Challenge Response Protocol

O protocolo ofuscado de resposta ao desafio não requer criptografia ou outras primitivas criptográficas e é robusto contra ataques de Machine Learning. No entanto, ele não fornece autenticação mútua e é restrito a dispositivos IoT que se autenticam em um servidor.

A principal ideia do protocolo é ofuscar a relação direta entre desafios e respostas, transferindo apenas parte do desafio para o chamado PUF Obfuscated (OB).

Um OB-PUF consiste em um gerador de números aleatórios, um bloco de controle de challenge e um PUF padrão, neste exemplo, um PUF arbitrário.

Quando um desafio parcial COB é aplicado ao OB-PUF, o bloco de controle de challange solicita ao gerador um número aleatório e o combina com COB para criar um desafio completo C. Esse desafio é aplicado ao PUF e a resposta é dada de volta.

Como um invasor não vê o CRPs completo e nunca sabe que número aleatório foi adicionado a um desafio parcial para gerar uma determinada resposta, este esquema torna muito mais difícil para um ataque de machine learning criar um modelo funcional do PUF.

Lockdown Protocol

O Lockdown Protocol concentra-se em fornecer resiliência contra ataques de machine learning sem a necessidade de criptografia. Fornece autenticação mútua, mas apenas entre um dispositivo e um servidor, não entre dispositivos.

O protocolo possui duas fases:

Na fase de inscrição, o servidor aplica todos os desafios possíveis ao PUF, colocando o PUF em um modo de bypass XOR especial. No entanto, em vez de armazenar as respostas diretamente como em outras abordagens, o servidor usa machine learning para gerar um modelo de verificação de autenticação SPUFiˆ, que simula o PUF. Este modelo SPUFiˆ vai ser usado para outros processos de autenticação.

No final da fase de inscrição, o PUF é bloqueado por meio de um fusível irreversível ou num armazenamento inviolável para garantir que os XORs ignorados não sejam mais possíveis.

Na fase de autenticação, o dispositivo envia primeiro o seu ID e um challange CD ao servidor. O servidor seleciona um segundo desafio CS e aplica CS‖CD a um gerador de números pseudoaleatórios, que gera um novo desafio <C>. O CS atua essencialmente como um contador e aumenta sempre que uma autenticação é concluída.

Isso impossibilita que os invasores obtenham dados suficientes para treinar seus modelos com eficiência. O servidor aplica <C> ao SPUFiˆ e recebe uma resposta r, que é dividida em duas partes r1 e r2, de modo que r = r1‖r2. Em seguida, o servidor envia CS e r1 para o dispositivo. Usando o CS recebido, o dispositivo reconstrói <C> e aplica-o ao PUF.

O resultado é r~ = r~1‖r~2. Então, o dispositivo calcula a distância fracionária de Hamming entre r1 e r~1. Se estiver além de um determinado limite, a autenticação será abortada. Caso contrário, o dispositivo envia r~2 para o servidor.

Finalmente, o servidor calcula a distância fracionária de Hamming entre r2 e r~2. Novamente, se eles diferirem demasiado, o servidor interrompe a autenticação. Caso contrário, a autenticação termina com sucesso.

Conclusão

Destes três protocolos apresentados, o Mutual Authentication Protocol, Obfuscated Challenge Response Protocol e o Lockdown Protocol, vou testar um ou os três para a tese do mestrado.

A questão a fazer será baseada no sucesso da implementação destes ou na comparação dos resultados obtidos durante os testes destes protocolos.

Referencias

1. Jin Y., Xin W., Sun H., Chen Z. Puf-based rfid authentication protocol against secret key leakage;

2. Kocabaş Ü., Peter A., Katzenbeisser S., Sadeghi A.R. Converse puf-based authentication

3. Van Herrewege A., Katzenbeisser S., Maes R., Peeters R., Sadeghi A.-H., Verbauwhede I., Wachsmann C. Reverse fuzzy extractors: Enabling lightweight mutual authentication for puf-enabled rfids;

4. Bolotnyy L., Robins G. Physically unclonable function-based security and privacy in rfid systems

5. Öztürk E., Hammouri G., Sunar B. Towards robust low cost authentication for pervasive devices;

6. Kulseng L., Yu Z., Wei Y., Guan Y. Lightweight mutual authentication and ownership transfer for rfid systems

7. Xu Y., He Z. Design of a security protocol for low-cost rfid;

8. Jung S.W., Jung S. Hrp: A hmac-based rfid mutual authentication protocol using puf;

9. He Z., Zou L. High-efficient rfid authentication protocol based on physical unclonable function;

10. Lee Y.S., Lee H.J., Alasaarela E. Mutual authentication in wireless body sensor networks (wbsn) based on physical unclonable function (puf

11. Gao Y., Li G., Ma H., Al-Sarawi S.F., Kavehei O., Abbott D., Ranasinghe D.C. Obfuscated challenge-response: A secure lightweight authentication mechanism for puf-based pervasive devices

12. Yu M.-D., Hiller M., Delvaux J., Sowell R., Devadas S., Verbauwhede I. A lockdown technique to prevent machine learning on pufs for lightweight authentication.