|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Trabalhos selecionados | | | | | |
| Ref | Título | Abordagem de computação confiável utilizada | Solução que utiliza computação confiável analisada no trabalho | Metodologia | Principais resultados |
| [1] | *ETS-TEE: An Energy-Efficient Task Scheduling Strategy in a Mobile Trusted Computing Environment* | Abordagem de computação confiável de *hardware* e *software*. Foi utilizado o TEE | O artigo busca diminuir a sobrecarga produzida por aplicativos no TEE e aumentar a eficiência energética por meio de estratégia de agendamento, no qual eles nomeiam ETS-TEE | Foi utilizado um Raspberry Pi 3B equipado com o processador A53 de quatro núcleos de 64 bits e 1 GB de RAM executando o OP-TEE compilado pelo aarch64-linux-gnu para simular um dispositivo móvel local. O servidor de bordar é um NVIDIA Jetson TX2 equipado com o Denver2 dual-core e ARM Cortex-A57 de quatro núcleos. | Durante a pesquisa, foram testados 4 algoritmos de aprendizado de máquina sendo eles o K-Nearest Neighbor (KNN), o Decision Tree (DT), o Naive Bayesian (NB) e o Support Vector Machine (SVM). O algoritmo KNN e SVM demonstraram bons resultados. A estratégia de agendamento confiável em servidor ficou, em média, 3,6 vezes mais rápida e reduzindo em 73,2% o consumo de energia comparado a execução local. |
| [2] | *Hardware-Based Trusted Computing Architectures for Isolation and Attestation* | Abordagens hibridas e em *hardware*. | Analisar a eficiência das soluções de computação confiável existentes para indústrias e acadêmicos. Avaliar seu desempenho para atestação e isolamento. | Foi analisado 12 soluções de computação confiável, verificando se possuem propriedades de segurança e características arquitetônicas. Ao final, as 12 soluções foram comparadas. | Ao final foi constatado que poucas aplicações suportam todos os possíveis mecanismos de computação confiável, sendo que nem todas as arquiteturas suportam certos recursos arquitetônicos. Os autores também citam que soluções em *hardware* são mais confiáveis. |
| [3] | *Mobile-only Solution for Server-based Qualified Electronic Signatures* | Abordagem de computação confiável de *hardware*. Foi utilizado o SE e o *secure enclave*. | Fornecer uma nova solução utilizando apenas um dispositivo móvel para autenticação em um provedor de acesso para admissão em servidores de assinaturas eletrônicas qualificadas. | Foi desenvolvido um aplicativo para Android e iOS que utiliza o SIM SE em androids e o secure enclave para iOS para armazenar informações sigilosas. | Os resultados das pesquisas foram satisfatórios, permitindo registros dinâmicos dos fatores de autenticação, no qual é possível integrar com qualquer método de autenticação existente no *smartphone*. A solução atendendo normas e regulações europeias (eIDAS e CEN). |
| [4] | *Offloading cryptographic services to the SIM card* | Abordagem de computação confiável de *hardware*. Foi utilizado o SE. | Transferir funções de armazenamento de chaves e processamento criptográfico para o SE, para entender a segurança caso o *smartphone* seja seguro. | Para o experimento foi utilizado o Sm@rtCafe Expert 6.0 80K JC, utilizando um *smartphone* Samsung Galaxy S5 (G900F). Para o ambiente de desenvolvimento foi utilizado o Netbeans IDE e o Android Developer Studio. | A abordagem forneceu uma arquitetura capaz de transferir serviços criptográficos para cartões SIM SE em *smartphone*s. O protótipo foi testado demonstrando que a solução funciona. Porém foi constatado algumas limitações de desempenho e um pequeno impacto no consumo de bateria |
| [5] | *On the Use of TEE for Mission Critical Public Safety Use Cases* | Abordagem de computação confiável de *hardware* e *software*. Foi utilizado o TEE | Desenvolver mecanismos de segurança para o acesso a Long-Term Evolution (LTE) utilizando o TEE. | Desenvolveram uma arquitetura de rede que utiliza o TEE e Aplicações confiáveis para realizar a autenticação dos *smartphone*s na rede. Por ser desenvolvida para segurança em projetos reais omitiram algumas informações. | O projeto atendeu às expectativas, mitigando alguns problemas de segurança. Porém, é necessário que o TEE não seja comprometido em sua execução ou que a senha PIN seja descoberta por atacantes. |
| [6] | *One enhanced secure access scheme for outsourced data* | Abordagem de computação confiável de *hardware* e *software*. Foi utilizado o TEE | Fornecer mais segurança a conexão de *smartphone*s a nuvem, aumentando o nível de confiança em *smartphone*s utilizando o TEE e o algoritmo CP-ABE. | Para o experimento foi utilizado um *hardware* equipado com um processador i3 e 3 GB de RAM com o sistema operacional OPEN-TEE. Para simular o servidor em nuvem foi utilizado a biblioteca PBC versão pbc-0.5.14. | A solução utilizando o CP-ABE otimizado pelo TEE demonstrou vantagem no overhead de armazenamento, comunicação e tempo, e apresentou resultados satisfatórios. |
| [7] | *Reversing and Fuzzing the Google Titan M Chip* | Abordagem de computação confiável de *hardware*. Foi utilizado o chip Titan M. | Avaliar a segurança e características do chip Titan M. | Um smartphone Pixel 3 com sistema operacional Android 11 foi utilizado para realizar os testes. Para obter acesso root, foram utilizadas as ferramentas Magisk, enquanto a engenharia reversa e o fuzzing foram realizados com Ghidra.". | Os testes encontraram uma vulnerabilidade de dia 0. Os autores citam o problema que poucas pesquisas exploram este componente, o que é uma desvantagem para seu aperfeiçoamento. |
| [8] | *Secure Authentication Key Sharing between Mobile Devices Based on Owner Identity* | Abordagem de computação confiável de *hardware* e *software*. Foi utilizado o TEE | Fornecer um serviço que aumente a usabilidade no gerenciamento de chaves criptográficas privadas de um dispositivo móvel para outro de forma segura e eficiente. | Foi utilizado o framework FIDO para a autenticação, foi projetado um aplicativo confiável para o TEE baseado em ARM *Trustzone*. O processo foi desenvolvido em um *smartphone* Samsung Galaxy S8. | Para realizar o processo de criptografar, enviar e decrepitar um total de 40 chaves, o processo ao todo levou menos de 1 segundo. Para obter esses resultados e manter a segurança foi utilizado o TEE para realizar as operações criptográficas e incluído um terceiro confiável no processo. |
| [9] | *TEEv: Virtualizing Trusted Execution Environments on Mobile Platform* | Abordagem de computação confiável de *software*. Foi utilizado uma adaptação do *hypervisor*. | Criar uma arquitetura que prove a virtualização de instâncias TEE isoladas, na qual cada TEE possui plena compatibilidade com o aplicativo confiável. Essa solução foi nomeada de TeeV. | Foi implementado o TEEv em uma placa de desenvolvimento Samsung Exynos 4412 e em um telefone móvel Mediatek equipado com um processador ARM Cortex-A9. Foram implementados o OP-TEE e o TrustKernel’s T6 e avaliado o desempenho da arquitetura. | Foram possíveis criar instancias TEE de diferentes fornecedores para executar seus próprios aplicativos confiáveis. As instâncias são isoladas uma das outras, caso haja mais de uma. O Teev foi implementado e executado em dispositivos reais, comprovando sua eficiência. |
| [10] | *The Case for a Virtualization-Based Trusted Execution Environment in Mobile Devices* | Abordagem de computação confiável de *software*. Foi utilizado uma adaptação do *hypervisor*. | Aumentar a eficiência do TEE utilizando um *hardware* de virtualização | Para o experimento foi utilizado uma placa de desenvolvimento HiKey equipada com um Kirin 620 SoC com CPU octa-core ARM Cortex-A53 de 64 bits com a capacidade de operar a 1,2GHz, 2 GB de RAM. Foi usado o hipervisor Xen 4.9 e como Sistema operacional utilizaram o CentOS com Kernel Linux versão 4.1. | Os resultados obtidos mostram que a arquitetura de virtualização suporta memória segura, entrada/saída de dados segura e poderia ser implementado acesso a memória segura também após algumas modificações no designe. |
| [11] | *Trusted Runtime Environment Building Technology for Mobile Terminals* | Abordagem de computação confiável como projeto de *software*. | Garantir a confiabilidade de terminais inteligentes em dispositivos móveis, por meio da validação do processo de segurança do sistema operacional Android. | Foi feito um mapeamento de técnicas de medições, detecção, validação, análise. O estudo também verificou o funcionamento de arquivos existentes no Android. | A solução para melhorar o *design* do ambiente de computação confiável, focando na detecção confiável em cada parte do aplicativo é promissor e eficiente.1 |
| [12] | *TrustyShare: A Sharing Scheme using ARM TrustZone* | Abordagem de computação confiável de *hardware* e *software*. Foi utilizado o TEE | Fornecer uma solução chamada *Trustyshare*, baseada em ARM *TrustZone*, para compartilhamento de serviços sem o suporte oficial desses serviços, evitando a divulgação das credenciais do usuário. | Foi utilizado um sistema P2P para conexão. Foi utilizado o protocolo Diffie-Hellman para auxiliar. Foi utilizado um Raspberry Pi 3B+ equipado com 1GB de RAM e CPU BCM2837B0 para arm64. O TEE utilizado foi o OP-TEE 3.9.0. | Foi testado dois cenários: colaboração por e-mail e compartilhamento de serviços web. Os resultados mostraram que a implementação possui desempenho aceitável para aplicações reais, na qual em *smartphone*s modernos podem obter resultados mais eficientes. |
| [13] | *Trusted mobile computing: An overview of existing solutions* | Abordagem de computação confiável de *hardware*, *software* e hibrida. | Apresentar as tecnologias eficientes para o armazenamento de chaves criptográficas em *smartphone*s e exemplo de aplicações. | Foi analisada a documentação e artigos referente a 4 soluções de computação confiável e de tecnologias emergentes. | Os autores afirmam que grande parte das soluções para computação confiável em dispositivos móveis utilizam a virtualização ou a computação em nuvem para solucionar limitações que componentes de *hardware* possuem. Sugerindo que soluções genéricas misturem ambiente de execução segura com tecnologias em nuvem. |
| [14] | *simTPM: User-centric TPM for Mobile Devices* | Abordagem de computação confiável de *hardware*. | Desenvolver uma solução de TPM em um SE SIM *card*. Garantindo que o TPM implementado em *smartphone*s tenha resultados semelhantes ao TPM 2.0 | Foi utilizado um cartão inteligente Gemalto MultiAPP Multos com um microprocessador da família Infineon SLE78CLX para o simTPM. O cartão foi testado em uma placa de referência Hikey960. Para os testes de desempenho, foram utilizados dois dispositivos: um Raspberry-PI e um laptop Lenovo com um TPM embutido. | O simTPM atende aos requisitos de segurança e desempenho. Porém o simTPM precisa de um TEE-proxy ou um protocolo de limite de distância para que sua segurança não seja violada. |