Classification

Approximate Computing Security

Security issues and countermeasures in a mixed software/hardware perspective.

Storage

Strictly separating storage and creating boundaries between approximate and precise data leads to a variety of **new security threats** and recent papers provide information on dangers like precision Flag Modifying or Misreading and theoretical attacks examples [1] [2]. The last paper especially provides the idea of "**blurring boundaries**" as a countermeasure, obfuscating when/how and what to approximate. **Memory Address Space Layout Randomization** is also considered a worthy countermeasure. [3]

Additionally, a potential attacker could compromise **DRAM refresh logic** and **memory allocator** [1] while the use of DRAM creates a whole other issue of **privacy deanonymization**. This threat is described in [4] alongside algorithms for creation, identification, comparison of DRAM "fingerprints" and creation of a mathematical model to **evaluate** the deanonymizing effects.

Compiler

There is only a little information on approximate computing dangers that focus on the compiler. One danger documented is the potential to change instructions to **falsely store data** [1]

Yet, many safe software creation ideas/frameworks have been developed. There is a **Java** language support [5] dedicated to safe approximate computing and a C/C++ Safe **Compiler Framework** "ACCEPT" [6]. Additionally, "**Parallelly**" is a software approach to the safety and accuracy of approximate parallel programs [7]. It provides a programming language and a system for the verification of approximations in parallel message-passing programs.

Data

Attacks on approximate data include **Error Injection** to Accurate Computing and **Modification of output,** while a common target is **misleading** of Accurate and Approximate Data [3].

Tactics like random noise injection have been used as a defense [3] but the recent bibliography focuses on the idea of **Information Hiding** in approximate data and approximate operations [8], meaning embedding information in the floating-point format.

Machine Learning

In Approximate **Artificial Neural Networks**, an attacker can **sabotage** weights and parameters for the trained neurons or even the arithmetic operations, as we see in [9]. The last paper also suggests a framework for defense. Most Significant Digit (MSD)-First arithmetic and information hiding techniques can also make Approximate Machine learning safer as suggested in [10].

When it comes to Approximation Computing's role in security, a defensive approximation technique for **CNNs** has been developed based on hardware-supported Approximation Computing (approximate floating-point multiplier) [11]. Moreover, there is a theoretical and algorithmic framework established for safety-critical learning using approximation techniques [12].

Approximate Query Engines/Sampling

The idea of sampling in databases to return approximate answers (Approximate Query Engine) is relatively old [13]. Using this logic for optimization of the **threat monitoring** processes [14] shows how approximation is tied to security issues in a variety of ways.

Sampling can also improve performance and **privacy** as seen by the "PrivApprox" implementation [15]. The paper introduces PrivApprox with design, implementation, and evaluation details. The main idea is the combination of approximation with adding explicit noise for privacy preservation.

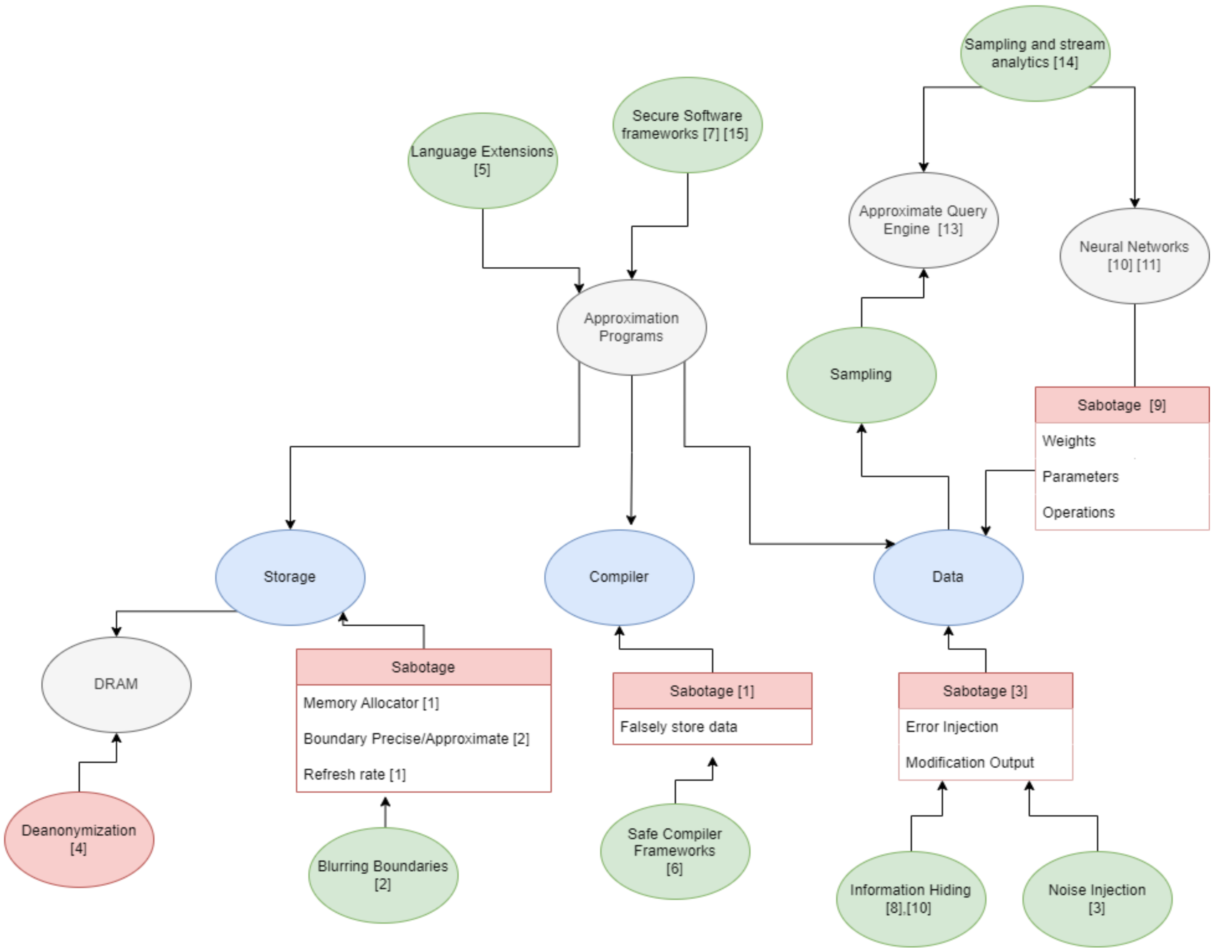

Figure 1: Map of AC threats, countermeasures, and contribution in security issues.

These notes focus on software-related issues so approximate **circuit** security threats are not fully examined.

Cryptography

Approximate computing can be used in a variety of cryptography-related issues [16].

Quantum Approximate Optimization Algorithm

Approximate quantum encryption has been defined since 2010 [17] with proofs of security, but **Quantum Approximate Optimization Algorithm** (QAOA), a variational hybrid quantum-classical algorithm, was introduced in 2014. This algorithm is implemented by a quantum circuit and produces approximate solutions for combinatorial optimization problems [18]. QAOA has been improved, widely discussed, and recently used for Secure Smart Logistics Systems [19].

Homomorphic encryption

The open-source implementation of **homomorphic encryption** for **approximate arithmetic** established a common ground between the two [20]. Since the creation of the HE library (HEAAN), more libraries have been created and used for secure machine learning and privacy-related issues. There is a variety of papers on the issue. The most recent one underlines the need for a stronger definition to evaluate the security of such schemes, providing a better homomorphic encryption **security evaluation** theoretical basis [21].

Approximate Hardware

With the usage of Approximate Modular-32 Adders approximation can be infused in cryptographic hash functions with description and evaluation of the effects [22]. Bibliography suggests that Approximate Computing techniques can be utilized for the design of area/power-efficient modular multiplier for R-LWE [23] [24].

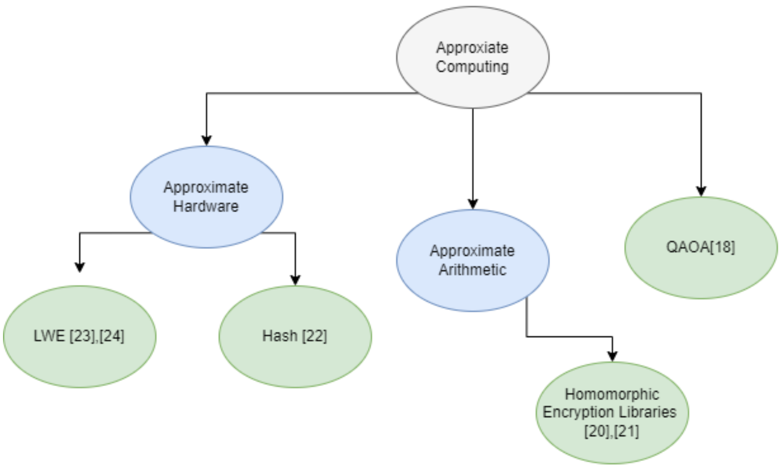

Figure 2: Mapping the relationship between AC and issues discussed

Bibliography

[1] “Security Threats in Approximate Computing Systems,” 2019.

[2] “Blurring Boundaries: A New Way to Secure Approximate Computing Systems,” 2015.

[3] “Is it Approximate Computing or Malicious Computing,” 2020.

[4] “Probable Cause: The Deanonymizing Effects of Approximate DRAM,” 2015.

[5] “FlexJava: language support for safe and modular approximate programming,” 2015.

[6] “ACCEPT: A Programmer-Guided Compiler Framework for Practical Approximate Computing”.

[7] “Verifying safety and accuracy of approximate parallel programs via canonical sequentialization,” 2019.

[8] “Information Hiding behind Approximate Computation,” 2019.

[9] “Security Threats and Countermeasures for Approximate Arithmetic Computing,” 2020.

[10] “Security Enhancements for Approximate Machine Learning,” 2021.

[11] “Defensive Approximate: Securing CNNs using Approximate Computing,” 2021.

[12] “Safe Pontryagin Differentiable Programming,” 2021.

[13] “Knowing when You’re Wrong: Building Fast and Reliable Approximate Query Processing Systems,” 2014.

[14] “Scalable Cyber-Security Analytics with a New Summary-based Approximate Query Engine and Approximate Computing,” 2017.

[15] “Privacy Preserving Stream Analytics: The Marriage of Randomized Response and Approximate Computing(PrivApprox),” 2017.

[16] “Securing in Approximate Computing and Approximate Computing for Security,” 2020.

[17] “Quantum entropic security and approximate,” 2010.

[18] “A Quantum Approximate Optimization Algorithm,” 2014.

[19] “A Quantum Approximate Optimization Algorithm Based on Blockchain Heuristic Approach for Scalable and Secure Smart Logistics Systems,” 2021.

[20] “Homomorphic encryption for arithmetic of approximate numbers,” 2016.

[21] “On the Security of Homomorphic Encryption on Approximate Numbers,” 2021.

[22] “ApproxHash: Delay, power and area optimized approximate hash functions for cryptography application,” 2017.

[23] “DWE: Decrypting Learning with Errors with Errors,” 2018.

[24] “AxMM: Area and Power Efficient Approximate Modular Multiplier for R-LWE Cryptosystem,” 2020.

[25] “DArL: Dynamic Parameter Adjustment for LWE-based Secure Inference,” 2019.

[26] “Security Threat Analyses and Attack Models for Approximate Computing From Hardware and Micro-architecture Perspectives”.

Στα πλαίσια του ειδικού θέματος: «Interplay between approximate computing and Security»

Διδάσκων Αντωνόπουλος Χρήστος, προπτυχιακή φοιτήτρια Ελένη Ξωχέλλη