Abstract

Symmetric key cryptography stands as a crucial pillar in ensuring security within the contemporary landscape of electronic communication. Cryptanalysis of classical symmetric key ciphers involves traditional methods and techniques for breaking or analyzing these cryptographic systems. For new ciphers, enforcing resistance against the linear and differential cryptanalysis is a common design criterion during the analysis of the cipher. The wide trail design technique for block ciphers leads to proving the security against linear and differential cryptanalysis. Finding the minimum number of active SBoxes for all the rounds of a cipher is a helpful way to assess the scheme's security against differential attacks. The propagation characteristics of a cryptographic component (like SBox) can be expressed using Boolean functions. Mixed Integer Linear Programming (MILP) is a useful technique for solving a boolean function. We generate a set of inequalities to model a boolean function to be solved by an MILP solver. Here, we must resolve two issues to determine an efficient solution model for any boolean function. Create a set of inequalities to solve the boolean function efficiently. If the first issue is resolved, select a minimal set of possible inequalities that perfectly model the boolean function. We propose algorithms for searching for the solution to the second issue and finding more optimized linear and non-linear components. We apply the approaches for modeling SBoxes (up to six bits) and EXOR operations with any number of inputs. We also introduce MILP approaches to search differential and impossible differential propagations in a cipher. The techniques are applied to five lightweight block ciphers: Lilliput, GIFT64, SKINNY64, Klein, and MIBS. Impossible differential attack is a powerful cryptanalysis technique against AES. Here, our primary focus is exploring the memory-efficient and parallel implementation techniques for impossible differential attacks targeting AES. We retrieve partial or full-key information for five to six rounds of AES. A partial sum attack is an extension square attack, a chosen plaintext attack. We suggest an efficient GPU-based implementation of the partial sum attack on five and six-round AES. Additionally, we outline a cluster-based approach for executing the partial sum attack, aiming at the full key recovery of six-round AES. Furthermore, we delve into the integration of

Deep Learning (DL) as a versatile tool for analyzing symmetric key ciphers. In doing so, we illustrate how conventional security frameworks, devoid of DL incorporation, tend to locate the vulnerabilities inherent in these ciphers. Notably, this study marks the pioneering use of DL in a generic capacity within this domain, expanding the scope of analysis and understanding. We use the tool to search for differential distinguishers for four ARX-based ciphers: HIGHT, LEA, SPARX, and SAND.

Keywords: Impossible Differential Cryptanalysis, AES, Differential Cryptanalysis, Mixed Integer Linear Programming, Partial Sum Attack, Non-linear Layer, Linear Layer, Deep Learning.