Algebraic Attacks on Clock-Controlled Cascade Ciphers

Kenneth Koon-Ho Wong[1], Bernard Colbert[2], Lynn Batten[2], and Sultan Al-Hinai[1]

[1] Information Security Institute (ISI), Queensland University of Technology (QUT), Brisbane, Australia
[2] Deakin University, Melbourne, Australia

Abstract. In this paper, we mount the first algebraic attacks against clock controlled cascade stream ciphers. We first show how to obtain relations between the internal state bits and the output bits of the Gollmann clock controlled cascade stream ciphers. We demonstrate that the initial states of the last two shift registers can be determined by the initial states of the others. An alternative attack on the Gollmann cascade is also described, which requires solving quadratic equations. We then present an algebraic analysis of Pomaranch, one of the phase two proposals to eSTREAM. A system of equations of maximum degree four that describes the full cipher is derived. We also present weaknesses in the filter functions of Pomaranch by successfully computing annihilators and low degree multiples of the functions.

1 Introduction

Algebraic attacks, in which the initial states are solved for as a system of multivariate polynomial equations derived from the target cipher, were introduced by Courtois and Meier in [10,12] as a new method of analyzing cipher output. This method of attack was first applied to block ciphers and public key cryptosystems by Courtois and Pieprzyk [9,15]. Many *regularly* clocked linear feedback shift register (LFSR) based stream ciphers have since then fallen under algebraic attacks, as demonstrated in [3,4,7,15,11], whereas *irregularly* clocked stream ciphers have been more resistant. There are, to our knowledge, only two papers in the literature dealing with algebraic attacks on irregularly clocked stream ciphers, first in [12] and then in [1], dealing with separate classes of clock controlled stream ciphers. Our interest in this paper is in extending algebraic attacks to a third class of clock controlled stream ciphers that has not yet been examined under algebraic attacks — the clock controlled LFSR-based *cascade* stream ciphers.

In an LFSR based clock control cascade cipher, the output of the first LFSR controls the clocking of a second LFSR, both outputs together control the clocking of a third LFSR, and so on. In this paper, we present algebraic analyses of two such ciphers, the Gollmann cascade generator [17] and Pomaranch, an eSTREAM project candidate [19,21].

The idea of cascading a set of LFSRs was due to Gollmann [17] and was further studied by Chambers and Gollmann in 1988 [18]. In the latter study, they

R. Barua and T. Lange (Eds.): INDOCRYPT 2006, LNCS 4329, pp. 32–47, 2006.

conclude that better security is achieved with a large number of short LFSRs instead of a small number of long ones. Park, Lee and Goh [24], having extended the attack of Menicocci on a 2-register cascade using statistical techniques [23], successfully broke 9-register cascades where each register has fixed length 100. They suggested also that 10-register cascades might be insecure. In 1994, Chambers [6] proposed a clock-controlled cascade cipher in which each 32-bit portion of the output sequence of each LFSR passes through an invertible s-box with the result being used to clock the next register. Several years later, the idea of cascade ciphers resurfaced in a proposal by Jansen, Helleseth and Kholosha [19] to the 2005 SKEW workshop, which became the eSTREAM candidate Pomaranch. Pomaranch can be viewed as a variant of the Gollmann cascade in which a number of bits from each register are filtered using a nonlinear function, and the result is used to control the clocking of the next register.

In the case of the Gollmann cascade, the key is the combined initial states of the registers, and the keystream output is the output of the final register. Pomaranch uses an initialization vector for key loading and its keystream output is the sum of certain bits taken from each of the registers. In subsequent sections of this paper, we present our algebraic attacks on the Gollmann cascade generator. This leads us into our algebraic analysis of Pomaranch, where the cipher construction is more complicated than the Gollmann cascade. Unless otherwise specified, additions and multiplications presented in this paper are defined over GF(2).

2 Clock-Controlled Gollmann Cascade Generator

The Gollmann cascade generator, introduced in [17], employs k LFSRs arranged serially such that each register except for the first one is clock-controlled by an input bit, which is the sum of the output bits of its predecessors. This structure is shown in Figure 1. Initially, all registers are filled independently with key bits. Let the input bit to the i-th register at time t be a_i^t, for $i \geq 2$. The i-th register is clocked if and only if $a_i^t = 1$. The output bit of the i-th register is then added to a_i^t, and the result becomes a_{i+1}^t. The keystream output of the generator is the output of the k-th LFSR.

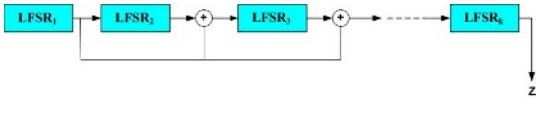

Fig. 1. The Gollmann Cascade Generator

Gollmann first proposed cascading k cyclic registers of the same prime length p with feedback polynomial $f(x) = x^p + 1$. This is known as the p-cycle. A variation of the Gollmann cascade, called an m-sequence cascade, has the cyclic registers replaced by maximum length LFSRs of the same length l. We will algebraically analyse this type of Gollmann cascade generator with registers of