

# A game on groups and information security.

Liljana Babinkostova and Marion Scheepers\*

## Abstract

We introduce a game on finite groups. This game encapsulates an adaptive chosen ciphertext attack against some cryptosystems, including El-Gamal style cryptosystems over any finite group and RSA style cryptosystems over any finite Abelian group. We give an initial analysis of the game and introduce two problems from computational group theory: the Coset Enumeration Problem (CEP) and the Isomorphism Class Problem (ICP). We give some simple complexity lowerbounds for CEP and ICP.<sup>1</sup>

Several commercial public key crypto-systems have a natural description in terms of groups. Considering crypto-systems in such generality helps to better distinguish which aspects of specific groups make them better suited to achieve a high level of security while maintaining acceptable computational efficiency and a consumer-friendly protocol.

Successful business practices necessitate that security requirements be balanced against customer satisfaction. For example: Account holders at a bank can access their accounts remotely using a computer to perform a variety of banking functions online - like checking on account balances or transferring money from a savings account to a checking account. Some banks permit only a limited number of successive failed login attempts before the online access to the account is disabled. Some clients may not like this because it makes them vulnerable to a simple denial of service attack: An attacker who correctly guesses the login name of an account holder can disable the online access to the account by intentionally making a number of failed login attempts. The inconvenience that could be caused in this way may convince the account holder to choose another bank where this denial of service attack is not possible. A second feature which clients may want in their online banking service is that when a failed login attempt occurs, the server responds with some user friendly message indicating the reason for the failure. It might be that the submitted username does not match any username in the bank's records, or the username matches, but the password not. And though this is not common, one may write the software to respond with more detailed messages when there is a password failure - such as that passwords are required to be only six characters long, or are

---

\*Supported by NSF grant DMS 99 - 71282

<sup>1</sup>**Key words:** adaptive chosen ciphertext attack, coset, coset enumeration problem, El-Gamal, game, group, isomorphism class problem, public key cryptography, RSA, winning strategy

required to be only lowercase letters, and so on. A third feature which account holders may want is that remote correspondence with the bank's computers be encrypted to protect their confidential information and financial activities on-line from possible eavesdroppers.

Some user-friendly features may actually compromise security unless great care is taken that the crypto-system can tolerate making public these specific diagnoses. In section 1 of this paper we give a general description of an attack which exploits such user friendly features. We formulate the attack as a game between two players, ONE and TWO. In section 2 we give an initial analysis of the game for finite, but not necessarily Abelian, groups. In section 3 we look more closely at the case of finite Abelian groups.

The attack belongs to the area of adaptive chosen ciphertext attacks (see page 285 of [4]), and has been considered recently in [1] for the classical RSA crypto-system. The term "classical RSA" in our paper means the system proposed in [5], specifically for the unitary groups  $\mathbb{U}_n$ . Recall that these groups are defined as follows: An integer  $n$  which is the product of two prime numbers  $p$  and  $q$  is given.  $U_n$  is the set  $\{x < n : gcd(x, n) = 1\}$ , and the operation on this set is multiplication modulo  $n$ , denoted  $*_n$ . Then  $\mathbb{U}_n$  denotes the group  $(U_n, *_n)$ . In section 4 we describe the RSA system based on arbitrary groups  $(G, *)$ , and for convenience remind the reader also of some of the details of the El-Gamal system based on an arbitrary group. Then we point out that the adaptive chosen ciphertext attack considered here is available for these generalizations of these two crypto-systems.

In section 5 we describe two problems from computational group theory: The coset enumeration problem (CEP), and the Isomorphism class problem (ICP). Each of these problems seems at least as hard as the factoring problem, and presumably each is much harder than the factoring problem.

Though the main context for our consideration here is finite groups, the game we describe can also be played on infinite groups, a case of independent mathematical interest. The game on an infinite group will be considered elsewhere.

## 1 A game on groups.

To begin, let  $(G, +)$  be a finite group, not necessarily Abelian. Consider the following simple game between two players ONE and TWO: The game has only one inning. In this inning ONE chooses a secret element  $x$  from  $G$ . Then TWO guesses an element  $y$  of  $G$ . TWO wins if  $y = x$ ; else, ONE wins. Since  $G$  is a finite set the probability that TWO wins a play is  $\frac{1}{|G|}$ .

Intuition suggests that if TWO may first ask some indirect questions from ONE about  $x$ , then TWO may learn information about  $x$  which may be used to improve TWO's chance of guessing the correct value of  $x$ . We study a specific version of this. Here, officially, is the game we study: Let  $\mathbf{0}$  be the identity element of  $(G, +)$  and let  $A$  be a subset of  $G$ . The following game, denoted  $G(G, A)$ , between players ONE and TWO, is played as follows: In the first

inning ONE first chooses a secret element  $x \in G$ . Then TWO chooses an element  $a_1$  of  $G$ , and asks ONE if  $a_1 + x$  is an element of  $A$ . ONE answers truthfully by  $\epsilon_1$ , where  $\epsilon_1 = 1$  indicates “yes”, while  $\epsilon_1 = 0$  indicates “no”. Then TWO chooses an  $a_2$  in  $G$  and asks ONE if  $a_2 + x$  is in  $A$ ; ONE answers truthfully by  $\epsilon_2$ , where  $\epsilon_2 = 1$  indicates “yes”, while  $\epsilon_2 = 0$  indicates “no”. The game continues like that. Thus ONE and TWO construct a sequence

$$x, a_1, \epsilon_1, a_2, \epsilon_2, \dots$$

and after a number of these moves TWO selects an element  $y \in G$ . TWO wins the play

$$x, a_1, \epsilon_1, a_2, \epsilon_2, \dots, y$$

if  $x = y$ . Else, ONE wins.

Let  ${}^{<\omega}2$  denote the set of finite binary sequences, and let  ${}^{<n}2$  denote the set of binary sequences of length less than  $n$ . A strategy for player TWO is a function  $\Psi : {}^{<\omega}2 \rightarrow G$ . A sequence

$$a_1, \epsilon_1, a_2, \epsilon_2, \dots, \epsilon_k, a_{k+1}$$

is a  $\Psi$ -history of the game if  $\Psi(\emptyset) = a_1$  and for all  $n \leq k$  we have  $\Psi(\epsilon_1, \dots, \epsilon_n) = a_{n+1}$ .

For a given subset  $A$  of  $G$  and for a fixed nonnegative integer  $n$  let  $p_{n,A}$  denote the probability that after TWO has asked ONE  $n$  questions, TWO guesses  $y$  correctly. Observe that for  $n \leq k$  we have  $p_{n,A} \leq p_{k,A}$ .

We are interested in whether TWO has a strategy  $\Psi$  which assures that for large enough  $n$  the value of  $p_{n,A}$  is something larger than  $\frac{1}{|G|}$ . Ultimately we are interested in whether TWO has a strategy which assures that for large enough  $n$  the value of  $p_{n,A}$  is 1.

A strategy  $\Psi$  for TWO is said to be a winning strategy if: There is an  $n$  such that for each secret  $x$  chosen by ONE, following  $\Psi$  TWO can with probability 1 guess the value of  $x$  correctly after  $n$  innings.

For some subsets  $A$  of  $G$  it is evident that for all  $n$ ,  $p_{n,A} = \frac{1}{|G|}$ . If  $A = \emptyset$  ONE always answers “0”. If  $A = G$  then ONE always answers “1”: In either case TWO is reduced to eventually making a lucky guess. For some subsets  $A$  of  $G$  TWO can, by careful questioning of ONE, always eventually declare the correct value of  $x$ : If  $A = \{a\}$  for some  $a \in G$ , or  $A = G \setminus \{a\}$  for some  $a \in G$  then after at most  $n = |G| - 1$  innings, TWO will declare the correct value of  $x$ . TWO proceeds as follows: Make a repetition free enumeration  $(g_1, g_2, \dots)$  of  $G$ . If  $A = \{a\}$ , define the strategy  $\Psi$  as follows:  $\Psi(\emptyset) = g_1$ , and for a finite sequence  $(0, \dots, 0)$  of length  $i$  of zeroes,  $\Psi(0, \dots, 0) = g_{i+1}$ . If for some  $i$ , ONE responds with  $\epsilon_i = 1$ , then TWO knows  $g_i + x = a$ , and TWO declares  $x = (-g_i) + a$ . If  $A = G \setminus \{a\}$ , then define the strategy  $\Gamma$  so that  $\Gamma(\emptyset) = g_1$ , and if  $(1, \dots, 1)$  is a sequence of  $i$  ones, then  $\Gamma(1, \dots, 1) = g_{i+1}$ . For some  $i$  ONE answers with  $\epsilon_i = 0$ . At this stage TWO knows  $g_i + x \notin A$ , and so  $g_i + x = a$ . Thus we have for  $n \geq |G| - 1$  that  $p_{n,A} = 1$ . Observe that when  $n < |G| - 1$ , then  $p_{n,A} \geq \frac{1}{|G|-n}$ .

When  $A = G$  or  $A = \emptyset$  the probability of guessing the correct value of  $x$  may be very low. When  $A = \{a\}$  or  $A = G \setminus \{a\}$  the probability is 1, but the number of questions TWO must ask ONE to arrive at the correct value of  $x$  may amount to an exhaustive search and may be exceedingly large. For the cryptographic applications of our game it is desirable that the probability of guessing  $x$  correctly is very low or the number of innings needed for TWO to arrive at the correct value of  $x$  is exceedingly large.

## 2 General finite groups.

For a subgroup  $H$  of  $G$  and an element  $a$  of  $G$  the set  $a + H (= \{a + h : h \in H\})$  is said to be a left coset of  $H$  in  $G$ . Distinct cosets of  $H$  in  $G$  are pairwise disjoint. The symbol  $G/H$  denotes the set of cosets of  $H$  in  $G$ . The number of cosets of  $H$  in  $G$ ,  $|G/H|$ , is  $|G|/|H|$ , and each coset of  $H$  has cardinality  $|H|$ . These facts can be gleaned from Chapter 7 of [2]. A subset  $S$  of  $G$  is a *system of distinct representatives* for subgroup  $H$  if for each left coset  $L$  of  $H$  in  $G$  we have  $|S \cap L| = 1$ .

**Proposition 1** *Let  $H$  be a subgroup of  $G$ . Suppose TWO knows a system of distinct representatives  $S$  for the subgroup  $H$  of  $G$ . Then TWO has a strategy  $\Psi$  such that  $p_{n,H} = \frac{1}{|H|}$  for  $n \geq |G/H|$ .*

**Proof :** Define a strategy  $\Psi$  for TWO as follows: Choose an enumeration of  $S$ , say  $(s_i : i \leq |G|/|H|)$ . For each  $i$  put  $a_i = -s_i$ . Define  $\Psi(\emptyset) = a_1$ , and for  $(0, \dots, 0)$  a sequence of  $i \leq |G|/|H|$  zeroes, define  $\Psi(0, \dots, 0) = a_{i+1}$ . For all other binary sequences, define  $\Psi$  arbitrarily.

Since  $G = \cup\{s_i + H : i \leq |G|/|H|\}$  there is for each secret choice of  $x$  by ONE some  $i \leq |G|/|H|$  with  $x \in s_i + H$ . For this  $i$ ,  $a_i + x \in H$ , and so within  $|G|/|H|$  innings TWO knows an  $i$  with  $x \in s_i + H$ .

Consider innings  $i$  and  $j$  where TWO has chosen an  $a_i$  such that  $a_i + x \in H$ , and an  $a_j$  with  $a_j + x \in H$ , respectively. Then TWO learns that  $x \in (-a_i + H) \cap (-a_j + H)$ , and thus we have  $(-a_i + H) = (-a_j + H)$  (see the coset properties listed on p. 133 of [2]).

Thus, the best information TWO can gather is that  $x$  is a member of the coset  $s_i + H$  discovered in the inning  $i \leq |G|/|H|$ . Thus, after inning  $|G|/|H|$  TWO is reduced to making a lucky guess as to which element of  $s_i + H$  the point  $x$  is. The probability of guessing correctly is  $\frac{1}{|H|}$ .

The number  $|G/H|$  of innings cannot be reduced for the strategy  $\Psi$  for TWO, because TWO does not know beforehand the  $x$  that ONE will choose.  $\diamond$

**Corollary 2** *Let  $A$  be a coset of a subgroup  $H$  of  $G$ . Assume that TWO knows a system of distinct representatives  $S$  for the subgroup  $H$ . Then TWO has a strategy  $\Psi$  such that  $p_{n,A} = \frac{1}{|A|}$  for  $n \geq |G|/|A|$ .*

**Proof** : Pick any element  $a$  of  $A$ . Then  $H = (-a) + A = \{-a + b : b \in A\}$ . Now simulate the game with  $H$  as a game with  $A$ .  $\diamond$

**Proposition 3** *Let  $A$  be a subset of  $G$  with  $A = H \cup \{a\}$  where  $H$  is a subgroup of  $G$  and  $a \notin H$ . Suppose TWO knows a set of distinct representatives  $S$  of  $H$  in  $G$ . Then TWO has a winning strategy in  $\mathbb{G}(G, A)$  which discovers ONE's secret in at most  $|G/H| + |H|$  innings.*

**Proof** : The game has, from the point of view of TWO, two stages. In the first stage TWO searches a point  $b$  such that  $b + x$  is in  $A$ . Since  $G$  is covered by the left cosets of  $H$  in  $G$ , and TWO beforehand knows a set of distinct representatives of the left cosets of  $H$ , TWO will find in at most  $|G/H|$  steps a  $b$  such that  $b + x \in A$ .

In the second stage TWO will use the point  $a$  to identify which element of  $A$  is  $b + x$ , as follows. Enumerate  $H$  bijectively as  $(h_1, \dots, h_k)$ . First TWO tests if  $b + x \in H$  by playing  $h_1 + b$ . If ONE responds that  $h_1 + b + x$  is not in  $A$ , then TWO knows that  $b + x = a$ . Else, TWO knows that  $b + x \in H$ . To determine which element of  $H$  this is, TWO now follows the following strategy: Take an  $h \in H$ . Ask if  $a - h + b + x$  is in  $A$ . If the answer is "yes", then TWO knows that  $x = (-b) + h$ . If ONE answers "no", then TWO knows that this choice of  $h \in H$  was the wrong one, and proceeds to another value of  $h$ . This stage takes at most  $|H|$  innings.

(Formally: Let  $(s_j : j \leq |G/H|)$  enumerate  $S$  and let  $(h_i : i \leq |H|)$  enumerate  $H$ . For each  $i$  put  $a_i = -s_i$ .  $\Psi(\emptyset) = a_1$ . For  $(0, \dots, 0)$  a sequence of  $i \leq |G/H|$  zeroes,  $\Psi(0, \dots, 0) = a_{i+1}$ . For some  $i \leq |G/H|$  ONE responds with an  $\epsilon_i = 1$ . At this stage TWO knows  $a_i + x \in A$ , and now the strategy  $\Psi$  calls on TWO to play  $\Psi(0, \dots, 0, 1) = h_1 + a_i$ . If ONE responds with 0, then TWO knows that  $a_i + x = a$ , and wins.

If ONE responds with a 1, then TWO knows that  $a_i + x \in H$ . Define  $\Psi(0, \dots, 0, 1) = a - h_1 + a_i$ . And for sequences of the form  $(0, \dots, 0, 1, 0, \dots, 0)$  where there are  $j$  zeroes following the 1, put  $\Psi(0, \dots, 0, 1, 0, \dots, 0) = a - h_{j+1} + a_i$ . When ONE responds with a 1 again, TWO knows that  $x = -a_i + h_j$ , and wins.)  $\diamond$

**Corollary 4** *Let  $K$  and  $H$  be subgroups of  $G$  such that  $K \cap H = \{\mathbf{0}\}$  and  $1 < |H|, |K|$  and put  $A = H \cup K$ . Suppose TWO has a set of distinct representatives for the left cosets of  $H$  and of  $K$  in  $G$ . Then TWO has a winning strategy in  $\mathbb{G}(G, A)$  which discovers ONE's secret  $x$  in  $|G|/\max\{|H|, |K|\} + \max\{|H|, |K|\}$  innings.*

**Proof** : We may suppose that  $|K| \geq |H|$ . By considering cosets of  $K$ , TWO discovers in at most  $|G|/|K|$  innings an  $a$  such that  $y = a + x$  is in  $A$ . To determine if  $y = \mathbf{0}$ , TWO first picks an  $h \in H \setminus K$  and a  $k \in K \setminus H$ . The TWO asks: Is  $h + y \in A$ ? If the answer is yes, then  $y \in H$ ; else,  $y \in K$ . If TWO determined that  $y \in H$ , then TWO asks:  $k + y \in A$ ? If the answer to this is

also yes, then TWO knows  $y = \mathbf{0}$ . In either case, TWO knows after these two questions if  $y \in H$  or if  $y \in K$ . If  $y \in H$ , then TWO uses  $k$  as in the proof of Proposition 3 to determine exactly which element of  $H$  is  $y$  - this takes at most  $|H|$  innings; if  $y \in K$  then TWO uses  $h$  as in the proof of Proposition 3 to determine exactly which element of  $K$  is  $y$ . This takes at most  $|K|$  innings.  $\diamond$

**Corollary 5** *Let  $B$  be a coset of a subgroup  $H$  of  $G$  and let  $A$  be  $B \cup \{a\}$  where  $a \notin B$ . Assume that TWO knows a system of distinct representatives  $S$  for the subgroup  $H$ . Then TWO has a winning strategy in  $\mathbb{G}(G, A)$  which discovers ONE's secret in at most  $|G|/|B| + |B|$  innings.*

**Proof :** Pick any element  $a$  of  $B$ . Then  $H = (-a) + B = \{-a + b : b \in B\}$ . Now simulate the game with  $H$  as a game with  $B$ .  $\diamond$

**Corollary 6** *Let  $B$  and  $C$  be cosets of the subgroups  $H$  and  $K$  of  $G$  where  $H \cap K = \{\mathbf{0}\}$ , and let  $A$  be  $B \cup C$ . Assume that TWO knows a system of distinct representatives for the subgroups  $H$  and  $K$ . Then TWO has a winning strategy in  $\mathbb{G}(G, A)$  which discovers ONE's secret in at most  $|G|/\max\{|C|, |B|\} + \max\{|C|, |B|\}$  innings.*

**Proof :** Pick any element  $b$  of  $B$ , and any element  $c$  of  $C$ . Then  $H = (-b) + B = \{-b + y : y \in B\}$  and  $K = (-c) + C$ . Now simulate the game with  $H$  and  $K$  as a game with  $B$  and  $C$ .  $\diamond$

In the proof of Proposition 3 the game had a second phase: During the first TWO merely attempted to get into  $A$ . In the second stage TWO was actually trying to determine which element of  $A$  is  $y$ , and thus which element of  $G$  is ONE's secret  $x$ . This amounted to an exhaustive search through  $A$ . While  $H$  might have very few cosets in  $G$ , for example 2,  $H$  may have an inordinate number of elements to search through to find the secret  $x$ . One might ask if there are subsets of  $G$  of the same cardinality as the subgroup  $H$ , but in which TWO has a more efficient winning strategy. We will now see that this could be the case, especially in the case of Abelian groups.

### 3 Finite Abelian Groups

We shall use the symbol  $Z_n$  to denote the set of possible remainders upon division by positive integer  $n$ , and endow it with the operation  $+$  of addition modulo  $n$ . Then  $(Z_n, +)$  is an Abelian group with identity element 0. We use the symbol  $\mathbb{Z}_n$  to denote this group. According to the Fundamental Theorem of Finite Abelian Groups, each finite Abelian group is isomorphic to a group of the form

$$\mathbb{Z}_{p_1^{n_1}} \oplus \cdots \oplus \mathbb{Z}_{p_k^{n_k}}$$

where  $p_1 \leq \dots \leq p_k$  are prime numbers – see Theorem 11.1 in [2]. We will now see that for special subsets  $A$  of  $\mathbb{Z}_{p_1^{n_1}} \oplus \dots \oplus \mathbb{Z}_{p_k^{n_k}}$  TWO has a winning strategy in  $\mathbf{G}(\mathbb{Z}_{p_1^{n_1}} \oplus \dots \oplus \mathbb{Z}_{p_k^{n_k}}, A)$ .

**Theorem 7** *For each positive integer  $n$  and for  $a < b \in \mathbb{Z}_n$  TWO has a winning strategy in the game  $\mathbf{G}(\mathbb{Z}_n, [a, b))$  which discovers ONE's secret  $x$  within  $\log_2(b-a) + \lceil \frac{n}{b-a} \rceil$  innings of the game.*

**Proof :** TWO's strategy will be executed in two phases. With  $x$  denoting the secret element chosen by ONE, TWO will in the first phase search for an  $r$  such that  $a \leq r + x \bmod n < b$ . Then, in the second stage TWO will use this information to identify  $x$ .

**Stage 1:** Put  $r_1 = n - a$ , and for each  $k$ , put  $r_{k+1} = r_k + (b-a) \bmod n$ . Observe that the set of translates  $r_1 + [a, b), \dots, r_j + [a, b), \dots$  has union  $\mathbb{Z}_n$ . TWO's strategy is to play successively  $t_j = n - r_j$ ,  $j = 1, 2, \dots$  until ONE responds that  $x + t_j \bmod n$  is in  $[a, b)$ , i.e. with a "1". Observe that such a  $j$  exists, since for some  $j$ ,  $x \in r_j + [a, b)$ , that is,  $(-r_j) + x = t_j + x \bmod n$  is in  $[a, b)$ . This occurs within  $\lceil \frac{n}{b-a} \rceil$  steps. Here is a formal definition of TWO's strategy  $\Psi$  on sequences  $(0, \dots, 0)$  of fewer than  $\lceil \frac{n}{b-a} \rceil$  zeroes:  $\Psi(\emptyset) = t_1$ , and with  $i$  zeroes,  $\Psi(0, \dots, 0) = t_{i+1}$ . Eventually ONE answers with a "1". Then stage 2 starts.

**Stage 2:** Now TWO knows that  $y = t_j + x \bmod n$  is in  $[a, b)$ , and remembers  $t_j$ , but does not know the value of  $y$ . In Stage 2 TWO will identify  $y$ . This is done by the bisection method. For each  $k$  put  $s_k = \lceil \frac{b-a}{2^k} \rceil$ . Also put  $a_0 = a$  and  $b_0 = b$ .

First TWO plays  $\Psi(0, \dots, 0, 1) = t_j + s_1 \bmod n$ . If ONE answers with  $\epsilon_1$  then TWO knows if  $y$  is in  $[a_{(\epsilon_1)}, b_{(\epsilon_1)})$ , that is in which part of  $[a, b)$  the point  $y$  is. Here we have:

$$a_{(\epsilon_1)} = \begin{cases} a_0 & \text{if } \epsilon_1 = 1 \\ b_0 - s_1 & \text{if } \epsilon_1 = 0 \end{cases}$$

and

$$b_{(\epsilon_1)} = \begin{cases} b_0 - s_1 & \text{if } \epsilon_1 = 1 \\ b_0 & \text{if } \epsilon_1 = 0 \end{cases}$$

Next TWO plays:  $\Psi(0, \dots, 0, 1, \epsilon_1) = t_j + \epsilon_1 * s_1 + s_2 \bmod n$ . If ONE now answers with  $\epsilon_2$ , then TWO knows in which fourth of  $[a, b)$  the point  $y$  is:

$$a_{(\epsilon_1, \epsilon_2)} = \begin{cases} a_{(\epsilon_1)} & \text{if } \epsilon_2 = 1 \\ b_{(\epsilon_1)} - s_2 & \text{if } \epsilon_2 = 0 \end{cases}$$

and

$$b_{(\epsilon_1, \epsilon_2)} = \begin{cases} b_{(\epsilon_1)} - s_2 & \text{if } \epsilon_2 = 1 \\ b_{(\epsilon_1)} & \text{if } \epsilon_2 = 0 \end{cases}$$

Let  $k \geq 1$  be given and assume that for all binary sequences  $\sigma$  of length  $\leq k+1$   $a_\sigma$  and  $b_\sigma$  have been defined such that  $\Psi(0, \dots, 0, 1, \sigma(1), \dots, \sigma(k)) + x \in [a_\sigma, b_\sigma)$ . Next TWO plays  $\Psi(0, \dots, 0, 1, \sigma(1), \dots, \sigma(k+1))$  which is

$$\Psi(0, \dots, 0, 1, \sigma(1), \dots, \sigma(k)) + \sigma(k+1) * s_{k+1} + s_{k+2}.$$

When ONE answers with  $\epsilon_{k+2}$  then TWO knows in which  $\frac{1}{2^{k+2}}$ -th segment of  $[a, b)$  the point  $y$  is, namely in  $[a_{\sigma \frown (\epsilon_{k+2})}, b_{\sigma \frown (\epsilon_{k+2})})$  where

$$a_{\sigma \frown (\epsilon_{k+2})} = \begin{cases} a_\sigma & \text{if } \epsilon_{k+2} = 1 \\ b_\sigma - s_{k+2} & \text{if } \epsilon_{k+2} = 0 \end{cases}$$

and

$$b_{\sigma \frown (\epsilon_{k+2})} = \begin{cases} b_\sigma - s_{k+2} & \text{if } \epsilon_{k+2} = 1 \\ b_\sigma & \text{if } \epsilon_{k+2} = 0 \end{cases}$$

Thus, it requires at most  $\log_2(b - a)$  steps to identify  $y$ , and so after a total of  $\log_2(b - a) + \lceil \frac{n}{b-a} \rceil$  innings, TWO identifies  $x$ .  $\diamond$

**Example:** Consider a factor  $k$  of  $n$  and choose  $a < b < n$  such that  $n = k * (b - a)$ . Let  $H$  be the subgroup  $\langle k \rangle$  of  $\mathbb{Z}_n$  generated by  $k$ . Then  $H$  is a subgroup of cardinality  $(b - a)$  of  $\mathbb{Z}_n$ , and  $[a, b)$  is a subset of cardinality  $(b - a)$ . According to Theorem 7 TWO has a winning strategy in  $\mathsf{G}(\mathbb{Z}_n, [a, b))$  which discovers ONE's secret  $x$  in  $k + \log_2(b - a)$  innings. According to Proposition 1 the best player TWO can do is to guess after  $k$  innings the value of the secret value  $x$  and with probability  $\frac{1}{b-a}$  will guess correctly. According to Proposition 3, if  $A = H \cup \{j\}$  for some  $j \in \mathbb{Z}_n$  not a multiple of  $k$ , then TWO can discover ONE's secret  $x$  in  $k + (b - a)$  innings.

**Theorem 8** *For positive integers  $n_1 \leq n_2 \leq \dots \leq n_k$ , and for  $0 \leq a_i < b_i < n_i$ ,  $i \leq k$  TWO has a strategy in the game  $\mathsf{G}(\mathbb{Z}_{n_1} \oplus \dots \oplus \mathbb{Z}_{n_k}, \prod_{j \leq k} [a_j, b_j))$  which discovers ONE's secret  $x$  within  $\log_2(\prod_{j \leq k} (b_j - a_j)) + \prod_{j \leq k} \lceil \frac{n_j}{b_j - a_j} \rceil$  innings of the game.*

**Proof :** Let ONE choose the secret  $x \in \mathbb{Z}_{n_1} \oplus \dots \oplus \mathbb{Z}_{n_k}$ . TWO's strategy is similar to the one before, executed in two stages. In the first stage TWO first search for an  $r$  such that  $r + x$  is in  $\prod_{j \leq k} [a_j, b_j)$ . Then TWO uses the bisection method to identify which element of  $\prod_{j \leq k} [a_j, b_j)$  the point  $r + x$  is. The rest of the argument is left to the reader.  $\diamond$

The work in [1] can be viewed as a heuristic analysis of the game  $\mathsf{G}(\mathbb{U}_n, [a, b))$  where  $[a, b)$  is a fairly large interval in  $\mathbb{U}_n$ .

## 4 An adaptive chosen ciphertext attack.

In an adaptive chosen ciphertext attack as described on page 285 of [4], the attacker chooses a ciphertext, and asks the victim to decrypt it, and then guided by the result, chooses another ciphertext which the victim should decrypt, and so on. In our consideration the attacker does not require to know the entire decryption of any chosen ciphertext, but merely to know whether the decryption belongs to a given set. In both cases the objective of the attacker is to determine the plaintext  $m$  of some ciphertext  $e$ , where  $e$  was known to the attacker before the attack started.

In the discussion below we assume that we have a group  $(G, *)$  and a one-to-one coding function  $C : M \rightarrow G$  from a sufficient message space  $M$  into the group  $G$ . We also assume that  $C$  has a “simple” inverse from its range to  $M$ . Here “simple” could mean that  $C^{-1}(g)$  is computable in polynomial time. We will assume that the user Alice will create encryption keys relative to this group, and publish her public key while keeping her private key secret. Bob will be the one encrypting a message to Alice, using Alice’s published keys. David is the eavesdropper, intercepting encrypted communication from Bob to Alice, and attempting to learn what was the original unencrypted message..

We consider this adaptive chosen ciphertext attack for two general classes of crypto-systems: The El-Gamal class of crypto systems and the RSA class of crypto-systems.

### **The class of El-Gamal crypto systems on general groups**

#### **Making a key:**

Alice, the key-maker, chooses a finite group  $(G, *)$ . Her keys relative to  $(G, *)$  are created as follows: She chooses an element  $g \in G$  with large order  $o(g)$ . Then she chooses a secret number  $p < o(g)$ , her private key. Using her private key she computes  $b = g^p$  in the group.

Public key:  $(G, *)$ ,  $g$ ,  $b$ .

Private key:  $p$ .

#### **Encrypting a message:**

If Bob wants to encrypt a message  $m$  to Alice he proceeds as follows:

1. Choose a random number  $r$ ;
2. Compute  $c = g^r$  and  $s = b^r$ ;
3. Compute  $e = C(m) * s$ ;
4. Send  $[c, e]$ .

#### **Decrypting a message:**

Upon receiving  $[c, e]$  Alice proceeds as follows to decrypt:

1. Compute  $c^p$  in  $G$ : The answer is  $s$ ;
2. Compute  $e * s^{-1}$  in  $G$ : The answer is  $C(m)$ ;
3. Compute  $m$ .

#### **Description of a chosen ciphertext attack:**

Now suppose that in fact Alice is a user-friendly server and Bob is attempting to log in. Part of Alice’s user friendliness is to inform Bob if the quantity  $m$  he encrypted as password has the correct format, that is, belongs to a specified subset  $A$  of  $G$ . Moreover the server Alice has been programmed to avoid the

denial of service attack described above, by not limiting the number of consecutive failed login attempts. Suppose David intercepted the encrypted message  $[c, e]$ . We may think of it that David wants to discover the secret group member  $C(m)$  in Alice's possession. He will attempt to do this by being player TWO in the game  $G(G, A)$ , considering Alice as player ONE in possession of  $C(m)$ .

David will use the crypto-system and Alice's user-friendly services as follows to indirectly ask from Alice if for an  $a$  of David's choice,  $a * C(m)$  is a member of  $A$ :

1. Choose a random number  $r_1$ ;
2. Compute  $c_1 = g^{r_1} * c$ ;
3. Compute  $s' = b^{r_1}$ ;
4. Compute  $e_1 = a * e * s'$ ;
5. Send  $[c_1, e_1]$  to Alice.

Alice processes  $[c_1, e_1]$  as follows:

1. Compute  $S = c_1^p$ , which is  $s * s'$ ;
2. Compute  $e_1 * S^{-1}$ , which is  $a * C(m)$ ;
3. Either give login access, or else send an error message that the submitted password does not have the correct format, or else send an error message that the submitted password is incorrect (but of correct format).

It is evident that David can use Alice's user-friendliness as well as the properties of El-Gamal encryption to simulate the game  $G(G, A)$  in such a way that David is player TWO. David wins when he discovers Bob's password  $C(m)$ .

### The class of RSA crypto systems on general groups

This is also possible with the class of RSA crypto-systems. The general construction of RSA crypto-systems is based on the following fact from elementary group theory (see Exercise 36 on page 144 of [2]):

**Theorem 9 (Folklore)** *Let  $(G, *)$  be a finite group and let  $n < |G|$  be a positive integer such that  $\gcd(n, |G|) = 1$ . Define the function  $f_n : G \rightarrow G$  as follows:  $f_n(a) = a^n$ . Then*

1.  $f_n$  is a one-to-one and onto function from  $G$  to  $G$ .
2. If  $(G, *)$  is an Abelian group then  $f_n$  is an automorphism of  $G$ .
3. With  $m = \frac{1}{n} \bmod |G|$ , the function  $f_m$  defined by  $f_m(a) = a^m$ ,  $a \in G$ , is the inverse of  $f_n$ .

**Making a key:**

To create an RSA crypto-system based on the group  $(G, *)$  requires that Alice knows  $|G|$ , the order of the group  $G$ . We will also assume that the group  $(G, *)$  is commutative - i.e., Abelian. Given this, Alice proceeds to define keys as follows:

Alice's keys relative to the group  $(G, *)$  are created as follows: She chooses a positive integer  $s$  such that  $\gcd(s, |G|) = 1$  and she computes  $t = \frac{1}{s} \text{mod} |G|$ .

Public key:  $(G, *)$ ,  $s$ .

Private key:  $t$ .

**Encrypting a message:**

If Bob wants to encrypt a message  $m$  to Alice he proceeds as follows:

1. Compute  $e = C(m)^s$  in  $G$ ;
2. Send  $e$ .

**Decrypting a message:**

Upon receiving  $e$  Alice proceeds as follows to decrypt:

1. Compute  $e^t$  in  $G$ : The answer is  $C(m)$ ;
2. Compute  $m$ .

**Description of a chosen ciphertext attack:**

Again suppose that in fact Alice is a user-friendly server and Bob is attempting to log in, as above. Suppose David intercepted the encrypted message  $e$  from a successful login attempt by Bob and wants to discover the secret group member  $C(m)$  in Alice's possession. Here's how David will attempt to do this by being player TWO in the game  $G(G, A)$ , considering Alice as player ONE in possession of  $C(m)$ .

David will use the crypto-system and Alice's user-friendly services to indirectly ask from Alice if for an  $a$  of David's choice,  $a * C(m)$  is a member of  $A$ :

1. Compute  $e_1 = a^s * e$ ;
2. Send  $e_1$  to Alice.

Alice processes  $e_1$  as follows:

1. Compute  $e_1^t$ , which is  $a * C(m)$  (since the group is Abelian);
2. Either give login access, or else send an error message that the submitted password does not have the correct format, or else send an error message that the submitted password is incorrect (but of correct format).

Evidently David can use Alice's user-friendliness as well as the properties of RSA encryption to simulate the game  $G(G, A)$  in which David is player TWO. David wins when he discovers Bob's password  $C(m)$ .

## 5 Final remarks: Two problems in computational group theory.

In the results in Section 2 we used the hypothesis that TWO has an enumeration of a system of distinct representatives for the left cosets of a subgroup of a group.

**The coset enumeration problem (CEP):** Let  $G$  be a finite group and let  $H$  be a subgroup of  $G$ . Give a list of elements from  $G$ , one from each left coset of  $H$  in  $G$ .

It is easy to describe an algorithm that would solve this problem, but it appears that such algorithms must necessarily be of high computational complexity. Constructing for a given subgroup of a group a system of distinct representatives for its cosets can be computationally as hard as factoring an integer which is a product of two primes. To see this, consider the unitary group  $\mathbb{U}_n$  where  $n = p * q$  and  $p$  and  $q$  are distinct prime numbers. Let  $H$  be any finite group, and consider the group  $G = H \times \mathbb{U}_n$ . Then  $H$  is a subgroup of  $G$  and has  $|G|/|H| = |\mathbb{U}_n|$  left cosets in  $G$ . If  $S$  is a system of distinct representatives of  $H$  in  $G$ , then  $|S| = |\mathbb{U}_n| = \phi(n)$ , and so the ability to construct or at least count the number of cosets of  $H$  in  $G$  amounts to the ability to compute  $\phi(n)$  for a number  $n$  which is a product of two distinct primes. It is well known that  $n = p * q$  can be factored in polynomial time from knowledge of the numerical values of  $n$  and  $\phi(n)$  (see Proposition I.3.4 in [3]). This at least gives some lowerbound of the complexity of this problem in terms of the quantity  $|G|/|H|$ . Presumably the lowerbound to the complexity of the coset enumeration problem is higher than the complexity of factoring integers.

This raises also the general problem of what is the true computational complexity of winning strategies for TWO in the scenario described in Proposition 3, even in the case of Abelian groups.

By the Fundamental Theorem for finite Abelian groups Theorem 8 describes for arbitrary finite Abelian groups some subsets for which TWO has a winning strategy in our game. But for this application one must first find for an arbitrary finite Abelian group a representation as in the Fundamental Theorem.

**The Isomorphism Class Problem (ICP):** Let  $(G, +)$  be a finite Abelian group. Determine the prime powers  $p_i^{n_i}$ ,  $i \leq k$  such that  $p_1 \leq \dots \leq p_k$  and  $(G, +)$  is isomorphic to  $\mathbb{Z}_{p_1^{n_1}} \oplus \dots \oplus \mathbb{Z}_{p_k^{n_k}}$ .

Again there are algorithms that would solve this problem, but it appears that also such algorithms must be of high computational complexity. Consider again the unitary group  $\mathbb{U}_n$  where  $n$  is a product of two prime numbers. If one has an efficient algorithm to determine the isomorphism class of  $\mathbb{U}_n$ , then one would have an efficient algorithm for computing the order,  $\phi(n)$ , of  $\mathbb{U}_n$ , and thus one would have an efficient algorithm for factoring  $n$ . Thus the factoring problem gives a lowerbound on the computational complexity of the Isomorphism Class Problem. Presumably this estimate is much lower than the true complexity of ICP.

This raises also the problem of what is the true computational complexity of a winning strategy for TWO in the case of finite Abelian groups.

Proposition 1 seems to suggest that one way to circumvent this adaptive chosen ciphertext attack would be to find a representation of data in the group  $G$  in such a way that the data form a reasonably large subgroup  $H$  of  $G$ . If this can be done, and if such a representation of data would not open the cryptosystem to another security compromise, then one might be able to implement the system with a larger degree of “customer friendliness”.

## References

- [1] D. Bleichenbacher, *Chosen ciphertext attacks against protocols based on the RSA encryption standard PKCS# 1*, in **Proceedings of Crypto '98** (1998), 1 – 12.
- [2] J.A. Gallian, *Contemporary Abstract Algebra* 4-th edition, **Houghton Mifflin Company**, 1998.
- [3] N. Koblitz, *A course in number theory and cryptography* (2 nd. Edition), **Springer Graduate Texts in Mathematics** 114, 1994.
- [4] A.J. Menezes, P.C. van Oorschot and S.A. Vanstone, *Handbook of Applied Cryptography*, **CRC Press**, 1997.
- [5] R.L. Rivest, A. Shamir and L.M. Adleman, *A method for obtaining digital signatures and public-key cryptosystems*, **Communications of the Association for Computing Machinery** 21 (1978), 120 – 126.

L. Babinkostova  
Department of Mathematics  
Boise State University  
Boise, Idaho 83725  
e-mail: liljanab@math.boisestate.edu

M. Scheepers  
Department of Mathematics  
Boise State University  
Boise, Idaho 83725  
e-mail: marion@math.boisestate.edu