Tatra Mt. Math. Publ. **41** (2008), 47–64

TATRA MOUNTAINS
Mathematical Publications

KLEPTOGRAPHIC ATTAKS ON E-AUCTION SCHEMES

Marcin Gogolewski*—Marcin Gomułkiewicz—
—Jarosław Grząślewicz—Przemysław Kubiak—
—Mirosław Kutyłowski—Anna Lauks

ABSTRACT. In this paper we examine resistance of electronic auctions protocols to kleptographic attacks. It turns out that these protocols are vulnerable to threats posed by kleptography and practical consequences might be profound. A party controlling production of software or hardware used in an auction may get access to complete data on user's decisions and construct his profile through *passive* observation only. At the same time, no other party can retrieve such data (even *after* reverse engineering the devices).

On technical side we show that an adversary using kleptography might deploy a single elliptic curve over a prime field to RSA key generation. This approach gives a shorter key than in the case of a twisted pair of elliptic curves over a binary field. This improves the attacks presented by Young and Yung. We also show that in case of tamper resistant devices the goals of kleptography might be achieved without public key algorithms, but through usage of hash functions.

1. Introduction

Electronic auctions fall into two main categories: In a *sealed bid auction* bidders simultaneously submit their bids so that the bids remain hidden until the bidding period is closed. In an *open auction* the bids are known to all bidders yet in the bidding period. For a more detailed taxonomy we refer the reader to the Appendix, and for a variety of schemes to the websites [14], [27].

There are many security requirements for auction protocols (cf. [3], [29]), we focus our attention on non-profilability and secrecy of sealed bids. Creating a profile of a bidder based on previous auctions could help to predict the bidder's decisions, thus good protocols require non-profilability [26]. Mind that

2000 Mathematics Subject Classification: 94A60.
Keywords: electronic auction, non-profilability, bid secrecy, kleptography, elliptic curves.
Partially supported by KBN grant 3T11C 011 26 in year 2006.
*Author is a holder of a scholarship received from Measure 2.6 of Integrated Regional Operational Programme.

non-profilability is a stronger requirement than anonymity, since the real name is unnecessary to create a profile, the same pseudonym in several open auctions and ability to link bidders with their bids is enough.

Secrecy of sealed bids is crucial for example in procurement schemes.

In this paper we examine resistance of electronic auctions protocols to kleptographic attacks. It turns out that these protocols are vulnerable to threats posed by kleptography and practical consequences might be profound. A party controlling production of software or hardware used in an auction may get access to complete data on user's decisions and construct his profile through *passive* observation only. At the same time, no other party can retrieve such data (even *after* reverse engineering the devices).

Kleptography was introduced in 1996 by Young and Yung [33], [34] (for more recent results see [35], [36] and references given there). It is a technique of embedding a trapdoor in a black box cryptosystem by the cryptosystem's manufacturer. The system works exactly according to its specification, but nevertheless it leaks private keys of the users. This covert channel is encrypted with a manufacturer's public key, hence even if later the implementation is reversely engineered and the trapdoor is revealed, the covert channel still remains inaccessible to others. Mind that thorough reverse engineering of hardware or software might be very costly, hence it is performed in very exceptional cases.

Recently, it has been pointed out that most e-voting protocols are vulnerable to kleptographic attacks [10], despite that these protocols should be particularly trustworthy. This is due to the fact that the electronic voting protocols do not incorporate any mechanism verifying pseudorandomness on the protocol level.

In this paper we show that for several e-auction protocols the situation is not any better. So attacks of this kind can be exploited against already deployed electronic procurement systems. The attacks presented in subsequent sections fall into two main categories:

1. Attacks well suited to software implementations: the device/software does not contain any individual secret shared with its manufacturer (also referred as Mallet), but only requiring access to Mallet's public key (for example discrete log related key (y_M, g, p) for a prime field, or elliptic curve one). The key might depend on the public key of the auction house, by sharing the same g, p, or if the attacker is able to eavesdrop regularly the communication line between the device and an auction house, it might be independent.
2. Hardware oriented attacks, requiring some unique secret key K, shared by the device and its manufacturer (like unique ID in attacks from [35], [36]). K is used to hide the attack by making statistical analysis of data on device's output intractable, but simultaneously helps Mallet to identify the

device. Moreover, key K evolves in the way that even after reverse engineering the device the past leakage remains unavailable to other parties.

1.1. Notation

In the paper we will use the symbols $\mathcal{R}$, H, $\tilde{H}$ to denote: some *cryptographically secure pseudorandom bit generator* (cf. [16], Def. 5.8) and two distinct *collision resistant hash functions* (cf. [16], Subsect. 9.2.2). Note that $\mathcal{R}$ is a deterministic one-way function.

In the subsequent sections elliptic curves are deployed extensively. This is due to compact representation of Diffie-Hellman key exchange. We shall use a method described in [18] and exploited also in [37]. We start with a more general setting. Let $E_{a,b}$ denote an elliptic curve defined over a binary field $\mathbb{F}_{2^\ell}$, given by equation

$$y^2 + xy = x^3 + ax^2 + b. \tag{1}$$

For any $E_{a,b}$ there are points $P_{a,1}, P_{a,2} \in E_{a,b}$ such that

$$E_{a,b} = \langle P_{a,1}\rangle \times \langle P_{a,2}\rangle \qquad \text{and} \qquad \operatorname{ord} P_{a,2} \,|\, \operatorname{ord} P_{a,1}.$$

If $\operatorname{ord} P_{a,1}$ has a single large prime factor q_a such that $q_a^2 \not| \operatorname{ord} P_{a,1}$, then some point G_a of order q_a can be determined, as well as some $P'_{a,1}$ such that

$$\langle P_{a,1}\rangle = \langle G_a\rangle \times \langle P'_{a,1}\rangle.$$

From now on we assume that both $\operatorname{ord} P'_{a,1}$, $\operatorname{ord} P_{a,2}$ are small. Under these assumptions Algorithm 2 from [17], which finds points $P_{a,1}$, $P_{a,2}$, is fast (it must factorize $\#E_{a,b}$, the number of points of $E_{a,b}$).

The output produced by a corrupted device/software should have no statistically distinctive features, indicating that the attack has been implemented. In order to achieve this Mallet will use a pair of twisted curves $E_{0,b}$, $E_{1,b}$ over $\mathbb{F}_{2^\ell}$, where ℓ is prime (in [18], $\ell = 163$). Clearly, point $\left(0, \sqrt{b}\right)$ belongs to both curves. At the same time, for each nonzero x there are two *different* points ((x, y) and $(x, x + y) = -(x, y)$) on *exactly one* of the curves $E_{0,b}$, $E_{1,b}$: from (1) we get

$$\left(\frac{y}{x}\right)^2 + \frac{y}{x} = x + a + \frac{b}{x^2},$$

and applying the trace function $\mathrm{Tr}_{\mathbb{F}_{2^\ell}/\mathbb{F}_2}()$ to both sides of the last equality we get

$$0 = \mathrm{Tr}_{\mathbb{F}_{2^\ell}/\mathbb{F}_2}\left(x + a + \frac{b}{x^2}\right), \quad \text{i.e.,} \quad \mathrm{Tr}_{\mathbb{F}_{2^\ell}/\mathbb{F}_2}(a) = \mathrm{Tr}_{\mathbb{F}_{2^\ell}/\mathbb{F}_2}\left(x + \frac{b}{x^2}\right),$$

so the right hand side of the last equation indicates the right curve $E_{a,b}$ for $a \in \{0, 1\}$. To sum up, each $x \in \mathbb{F}_{2^\ell}$ occurs twice in $E_{0,b} \cup E_{1,b}$. On the other hand, apart from points (x, y) satisfying (1) there is also a point $\mathcal{O}_{a,b} \in E_{a,b}$, called

"point at infinity", which is the neutral element of the group $E_{a,b}$. Consequently, the number $2 \cdot 2^{\ell}$ corresponds to the number of points in the set

$$\big(E_{0,b} \setminus \{\mathcal{O}_{0,b}\}\big) \cup \big(E_{1,b} \setminus \{\mathcal{O}_{1,b}\}\big).$$

Let $[k]B$ stand for point B multiplied by scalar k. Thus after setting $E_{0,b}$, $E_{1,b}$ and points G_0, $P'_{0,1}$, $P_{0,2}$, G_1, $P'_{1,1}$, $P_{1,2}$, any point $Q_a \in E_{a,b}$, for $a \in \{0,1\}$, might be expressed as

$$Q_a := [u]G_a + [v_1]P'_{a,1} + [v_2]P_{a,2} \tag{2}$$

for $0 \leq u < \operatorname{ord} G_a$, $0 \leq v_1 < \operatorname{ord} P'_{a,1}$, $0 \leq v_2 < \operatorname{ord} P_{a,2}$. Note that according to the choice of $E_{a,b}$ described above, $\operatorname{ord} P_{a,2} | \operatorname{ord} P'_{a,1}$.

The contaminated device/software will secretly send the x-coordinate $x(Q_a)$ of some point Q_a. Having received a nonzero $x(Q_a)$, Mallet may determine $a \in \{0,1\}$ ($a = \operatorname{Tr}_{\mathbb{F}_{2^\ell}/\mathbb{F}_2}\big(x(Q_a) + b \cdot (x(Q_a))^{-2}\big)$ for odd ℓ and $a \in \{0,1\}$). Then he may calculate any of the two y-coordinates for $x(Q_a)$. Next he multiplies the resulting point $\pm Q_a$ by a scalar k such that $k = 1 \bmod \operatorname{ord} G_a$ and $k = 0 \bmod \operatorname{ord} P'_{a,1}$ (note that such $k = 0 \bmod \operatorname{ord} P_{a,2}$). The outcome is one of the points $\pm[u]G_a$. If Mallet has received $x(Q_a) = 0$, then he knows that this is the x-coordinate of the point $Q_a = \big(0, \sqrt{b}\big)$ of both curves, and that the order of Q_a is equal to 2 (this is because $\big(0, \sqrt{b}\big) = -\big(0, \sqrt{b}\big)$). Hence u in $[u]G_a$ must be equal to 0.

Following [18], Mallet chooses curves $E_{0,b}$, $E_{1,b}$ with orders $4q_0$, $2q_1$, respectively, where q_0 and q_1 are primes (so the conditions imposed are quite sharp). Consequently,

$$|E_{0,b} \setminus \{\mathcal{O}_{0,b}\}| = 4q_0 - 1,$$
$$|E_{1,b} \setminus \{\mathcal{O}_{1,b}\}| = 2q_1 - 1.$$

The point of order 2 on $E_{1,b}$ is obviously the point $P'_{1,1} = \big(0, \sqrt{b}\big)$. Point $P'_{0,1} = \big(\sqrt[4]{b}, \sqrt{b}\big)$ is a point of order 4 on $E_{0,b}$. It follows that points $P_{a,2}$ disappear from (2). Next, Mallet finds points G_0, G_1 such that $G_a \in E_{a,b}$ and $\operatorname{ord} G_a = q_a$ for $a = 0, 1$. To finalize his public key generation Mallet chooses $x_a \in \{2, \ldots, q_a - 1\}$ uniformly at random and assigns $Y_a = [x_a]G_a$ for $a = 0, 1$. Finally, a description of $\mathbb{F}_{2^\ell}$, value b, and (G_0, G_1, Y_0, Y_1) or their x-coordinates, stand for his public key (refer to [24], [13] for binary elliptic curve's point compression methods).

To encode a message, the contaminated software sets $a = 0$ with probability $\frac{4q_0 - 1}{2^{\ell+1}}$, and $a = 1$ with probability $\frac{2q_1 - 1}{2^{\ell+1}}$ (neutral elements $O_{a,b}$ will not be generated, and as we could see above $|E_{0,b} \setminus \{\mathcal{O}_{0,b}\}| + |E_{1,b} \setminus \{\mathcal{O}_{1,b}\}| = 2^{\ell+1}$). Then it selects $u \in \{0, 1, \ldots, q_a - 1\}$ uniformly at random, calculates $[u]G_a$, $[u]Y_a$, chooses $v \in \{0, \ldots, 2^{2-a} - 1\}$ uniformly at random and assigns

$$Q_a := [u]G_a + [v]P'_{a,1}. \tag{3}$$

If $u = v = 0$, the selection is repeated from the beginning. *The crucial point is that this procedure ensures uniform distribution of values* $x(Q_a)$ *in the set* $\{0, \ldots, 2^\ell - 1\}$. The 163 bits of the x-coordinate $x(Q_a)$ will be transmitted, and $\mathcal{R}\big(H(x([u]Y_a))\big)$ will be used by the device as a source of random data.

To decode the secret, Mallet firstly gathers the bits of $x(Q_a)$, and then reconstructs $\pm[u]G_a$. Using his private key x_a he determines $\pm[u]Y_a$, and since the x-coordinate is independent of the sign of the point, he obtains $x\big([u]Y_a\big)$.

Note that the above encoding is suitable for software implementations as well, since device specific data (like ID or a key K) were not exploited.

Except the symbols introduced above, we follow the original notation from the protocols attacked.

2. Harkavy, Tygar and Kikuchi's scheme

The well known paper [11] describes a sealed bid auction protocol that uses secure distributed computations and preserves privacy of submitted bids. In this section we describe two kleptographic attacks aimed to break secrecy of sealed bids.

Let us briefly summarize the scheme from [11]. Let p be a sufficiently large prime (64 to 128 bit long), let $A_1, \ldots, A_m$ denote the auctioneers and $B_1, \ldots, B_n$ denote the bidders. Let b_j be a bid of the bidder B_j. According to [11] the value of each bid b_j is bounded by some V, i.e., $b_j \in \{0, 1, \ldots, V-1\}$. The bids are represented in some fixed base c, that is $b_j = b_{j1} \ldots b_{jd}$, for digits $b_{jk} \in \{0, 1, \ldots, c-1\}$ and $d = \lfloor \log_c(V-1) \rfloor + 1$. Each digit b_{jk} is encoded by an ordered set of $(c-1)$ *random* polynomials $s_{jk\ell} \in \mathbb{F}_p[X]$ of fixed degree t, where $\ell \in \{1, 2 \ldots, c-1\}$. Let

$$s_{jk\ell}(X) = \sum_{\xi=0}^{t} a_\xi^{(jk\ell)} X^\xi, \tag{4}$$

and let $z \in \{0, \ldots, c-1\}$ be the value of digit b_{jk}. Then the first z polynomials $s_{jk\ell}$ are chosen so that $a_0^{(jk\ell)}$ is chosen uniformly at random in $\{1, \ldots, p-1\}$, and for the remaining $(c-1) - z$ polynomials we have $a_0^{(jk\ell)} = 0$. Let $\alpha_i = \omega^i$, where $\omega \in \mathbb{F}_p$ is a primitive mth root of unity. Then the sequence

$$M_{ij} := s_{j11}(\alpha_i), \ldots, s_{jk\ell}(\alpha_i), \ldots, s_{jd(c-1)}(\alpha_i)$$

of $d \cdot (c-1)$ values of $d \cdot (c-1)$ polynomials $s_{jk\ell}$, where $k \in \{1, 2, \ldots, d\}$, $\ell \in \{1, 2 \ldots, c-1\}$, is a share for auctioneer A_i. The submission has the form

$$B_j \to A_i \colon E_{A_i}[M_{i,j}], D_{B_j}\big[h(M_{ij})\big], \tag{5}$$

where public key encryption E and signature scheme D are unspecified in [11].

The first attack. Since $\deg s_{jk\ell}$ is bounded by t, we have $t+1$ degrees of freedom in choosing polynomial $s_{jk\ell}$ with a nonzero free coefficient (probability that $a_0^{(jk\ell)}=0$ is $\frac{1}{p}$, hence it is negligible for a random polynomial). Assume that manufacturer Mallet can eavesdrop some $t+1$ communication lines $B_j \to A_i$ from bidder B_j. Since bidder's signature is given in (5) and B_j gets a receipt after submitting a bid, Mallet will know both i and j (if B_j would be mobile and his submission would be anonymous, the attack described subsequently might be applied). The choice of $t+1$ indexes $\{i_1, i_2, \ldots, i_{t+1}\}$ of communication lines was done arbitrarily by the manufacturer during the device's preparation. The number of eavesdropped lines might be chosen smaller, then at the expense of kleptographic channel's capacity the demand on listening watch would be smaller, too.

The device will transmit the value $x(Q_a)$ from (3), while $x\big([u]Y_a\big)$ for u, a chosen by the contaminated device/software shall serve as a seed for pseudorandom number generator $\mathcal{R}$.

Below, in (8) we need a deterministic scheme. If both E and D from (5) are probabilistic schemes, then the first output value of $\mathcal{R}$ is always taken for E, making it deterministic for Mallet. Next, if exactly one of the schemes is deterministic, then it will be used in (8) — it might be D instead of E, and the other scheme might be a source of an additional leakage. Below we assume that both schemes are deterministic.

A leak can be made by determining the polynomials $s_{jk\ell}$, or more exactly, by appropriate tuning their values at points α_i. Namely, from $\mathcal{R}$ the malicious device takes their consecutive nonzero coefficients $a_\xi^{(jk\ell)}$ for consecutive polynomials $s_{jk\ell}$. The only exception is the first polynomial of the polynomials encoding the first nonzero digit $b_{jk'}$ in the bid b_j ($b_{jk'} \geq 1$, hence $a_0^{(jk'1)} \neq 0$ in the polynomial $s_{jk'1}$). For $s_{jk'1}$, the output of $\mathcal{R}$ is used to set consecutive values in the vector

$$\big[s_{jk'1}(\alpha_{i_1}), s_{jk'1}(\alpha_{i_2}), \ldots, s_{jk'1}(\alpha_{i_{t+1}})\big] \tag{6}$$

and not the coefficients from formula (4)! The values chosen can be used to construct a polynomial. Indeed, ω is a primitive mth root of unity, so α_i are all distinct, and any square submatrix of the corresponding Vandermonde matrix is invertible. Consequently, if we compose a system of linear equations based on substituting X by $\alpha_{i_1}, \ldots, \alpha_{i_{t+1}}$ in (4), and take the right hand side values from (6), then we get a unique solution for

$$\Big[a_t^{(jk'1)}, a_{t-1}^{(jk'1)}, \ldots, a_0^{(jk'1)}\Big], \tag{7}$$

where $a_0^{(jk'1)} \neq 0$ with probability $1-\frac{1}{p}$. Then for $i \notin \{i_1, i_2, \ldots, i_{t+1}\}$ the values $s_{jk'1}(\alpha_i)$ are set by (7). Note that each coordinate of (6) appears in a different $M_{i'j}$ for $i' \in \{i_1, i_2, \ldots, i_{t+1}\}$, i.e., in bid submission (5) to a different auctioneer.

At each consecutive auction, in $t+1$ submissions (5) the device will send a parcel of say $4(t+1)$ consecutive bits of the value $x(Q_a)$ introduced in Subsection 1.1. So the whole number will be sent in $\lceil \frac{\ell}{4(t+1)} \rceil$ auctions.

In order to encode the parcel, the device will separately increment each of the coordinates of (6) until the following conditions are satisfied: the vector

$$\Big[\big(E_{A_{i_1}}[M_{i_1,j}]\big) \bmod 2^4, \ldots, \big(E_{A_{i_{t+1}}}[M_{i_{t+1},j}]\big) \bmod 2^4\Big] \tag{8}$$

agrees with bits of the parcel, and $a_0^{(jk'1)} \neq 0$. This tuning will take $(t+1) \cdot 2^4$ trials on average.

After receiving the whole value $x(Q_a)$ Mallet determines appropriate $x\big([u]Y_a\big)$ in a way described in Section 1. Then having the seed and one ciphertext $E_{A_{i'}}[M_{i',j}]$, he is able to find the proper bid value b_j in a number of trials proportional to $b_j \cdot 2^4$: for a candidate for b_j he performs $2^4 \cdot T$ trials, where T is some constant, and if all of them fail, i.e., the resulting ciphertexts are all different than the captured one, he takes the next b_j. The constant T is chosen according to Chebyshev's Inequality. Obviously, the attacker can also check b_j against other

$$E_{A_{i''}}[M_{i'',j}] \qquad \text{for} \quad i'' \in \{i_1, i_2, \ldots, i_{t+1}\} \setminus \{i'\}.$$

As we can see, the scheme E used in (8) must be deterministic for Mallet.

After transferring $x(Q_a)$, the device performs a deterministic update of the seed. From the output of $\mathcal{R}\big(\tilde{H}(x([u]Y_a))\big)$, it deterministically obtains a pair of numbers (u', S), where

$$u' \in \{0, \ldots, (\operatorname{ord} G_0 \cdot \operatorname{ord} G_1) - 1\}, \quad S \in \{0, \ldots, 2^{\ell+1} - 1\},$$

and this pseudorandom numbers generator gives uniform probability distributions for both u' and S. Next, the device substitutes $a' := 0$ if $S < 4q_0 - 1$ and $a' := 1$ otherwise. Then it changes the seed to $x\big([u']Y_{a'}\big)$ and calculates the corresponding value $x(Q_{a'})$ to be transferred now. Having received $x(Q_a)$, Mallet might also perform this and each of the subsequent deterministic updates.

Note however that if the scheme E used in (8) is time consuming, the tuning performed in (8) might cause large variations of execution time of the corrupted bidder's application. Thus the backdoor should be carefully implemented.

The second attack. Mallet will eavesdrop the channels from a bidder to auctioneers only occasionally. Additionally, suppose that the bidder is mobile and uneasy to track. Hence the following requirements need to be fulfilled: the bidder's device should be identifiable by its output, the past leak of bids values will remain unavailable to others even after reverse engineering, and the device must not be tamper-evident, since a bidder will not certainly use a tampered device. Accordingly, we assume that the device contains a unique key K, say a 128 bit random number known only to Mallet. Then for each new submission,

as a source of random data it uses $\mathcal{R}\big(H(K)\big)$ instead of $\mathcal{R}\big(H([u]Y_a)\big)$ mentioned above. To allow Mallet to identify the device by pointing K the coordinates (6) are incremented until

$$H\Big(E_{A_{i'}}\big[M_{i',j}\big] \;||\; K\Big) \bmod 2^4 = 0 \tag{9}$$

for all $i' \in \{i_1, i_2, \ldots, i_{t+1}\}$, where $||$ denotes concatenation. This gives $4(t+1)$ bits of identification of K for a single bid. The key K shall be changed immediately after transferring the bid, $K := \tilde{H}(K)$. This guarantees that after reverse engineering the device all transferred bid values remain hidden. Such a substitution requires non-volatile rewritable memory in the device like in [35].

Obviously, starting with the initial values for K stored in a database, the manufacturer Mallet performs the same substitutions to all those K, and after some time finds the right key by checking (9) for all $E_{A_{i'}}[M_{i',j}]$ from the captured bid. Having found K he proceeds like in the previous attack.

Both presented attacks work also for the schemes from [19], [12]. In these protocols a bidder chooses fresh large random number $\tilde{x}$ for each bit from her bid and sends a ciphertext of $\tilde{x}$ to the auctioneer (which is a front-end server). In these schemes number $\tilde{x}$ is a key allowing the auctioneer to decrypt one of two ciphertexts sent by a back-end server in 1-out-of-2 oblivious transfer protocol. Since each bid has at least a few bits, the situation is similar to the one in (8).

3. Omote and Miyaji's scheme

One-time registration (preceding participating in a series of English auctions) is a tempting idea, as it saves expenses and allows to split Registration Manager (RM) and Auction Manager (AM) entities. Such a scheme is presented in papers [21], [22] by K. Omote and A. Miyaji.

During initialization phase RM chooses and fixes some $g \in \mathbb{F}_p^*$ having large prime order. Each bidder $\mathcal{B}_i$ picks up her private key x_i, and submits to RM the corresponding public key $y_i = g^{x_i}$, which shall be published on RM's bulletin board. AM has its own key pair, $Y_{AM} = g^{\rho}$, where Y_{AM} is published.

Kleptographic attack on AM. Each successive auction has its own initialization phase. In scheme [21] during the auction setup fresh random numbers r_i for $i \in \{1, 2, 3, \ldots, I\}$ are generated, one for each bidder $\mathcal{B}_i$. Then pairs $(g^{r_i}, y_i^{r_i})$ are calculated and published on AM's bulletin board.

Assume that device or software used by AM has been contaminated with a kleptocode. As in the previous section, let (G_0, G_1, Y_0, Y_1) be points of twisted elliptic curves being Mallet's public key.

Suppose that the contaminated software takes $H\big(x([u]Y_a)\big)$ as a seed for the pseudorandom number generator $\mathcal{R}$ that produces the sequence of exponents r_α

for a given auction. Assume that ℓ-bit number $x(Q_a)$ is represented in a b-radix system for minimal b such that the b-ary length $\ell_b = \lfloor \log_b(2^\ell - 1) \rfloor + 1$ of the maximal ℓ-bit number is not greater than I. Let us consider the case of b not being the power of 2. By $\tilde{\ell}_b$ denote the binary length $\lfloor \log_2(b^{\ell_b} - 1) \rfloor + 1$ of the maximal b-radix ℓ_b digit number. Accordingly, the device extends the string $x(Q_a)$ with $(\tilde{\ell}_b - 1) - \ell$ random bits. Since strings $x(Q_a)$ are uniformly distributed in $\{0, \ldots, 2^\ell - 1\}$, this procedure gives strings $\tilde{x}(Q_a)$ uniformly distributed in $\{0, \ldots, 2^{\tilde{\ell}_b - 1} - 1\}$. To obtain the uniform distribution in $\{0, \ldots, b^{\ell_b} - 1\}$, a variant from [36] of the *Probabilistic Bias Removal Method* (PBRM) introduced in [34], is used. Assumptions of the method are fulfilled:

$$\frac{b^{\ell_b}}{2} < 2^{\tilde{\ell}_b - 1} < b^{\ell_b}.$$

PBRM randomly discards some of the $\tilde{x}(Q_a)$, but the resulting additional effort is negligible. For b being a power of two the PBRM is not necessary.

Let I_b be some number greater than I and let $(g^{r_\alpha})_{\alpha=1}^{I_b}$ be a sequence with exponents generated from $\mathcal{R}\big(H(x([u]Y_a))\big)$. The number $I_b \approx I \cdot b \cdot \ln c$ is determined by the probability that in the sequence of $b \cdot \ln c$ random numbers there would be no element in a row giving modulo b a fixed value. Easy calculations show that this probability is less than $\frac{1}{c}$.

Then the software sorts elements g^{r_α} and chooses some of them so that the ith value g^{r_α} on the list of chosen elements has the property that $g^{r_\alpha} \bmod b$ is equal to the ith b-radix digit of $\tilde{x}(Q_a)$.

For each r_α chosen the device also produces a value $y_j^{r_\alpha}$ for some bidder j, according to the protocol.

As a result Mallet can take the sequence of pairs $(g^{r_i}, y_i^{r_i})$ from AM's bulletin board, sort them according to the first coordinate, and after some calculations to undo the PBRM he reads $x(Q_a)$. Then, on the basis on his private key, Mallet can find the right $x\big([u]Y_a\big)$ and therefore he can get the whole sequence of initial exponents used by the device. Now he might link $y_i^{r_i}$ from the bulletin board with y_i. Thus in consecutive auctions Mallet will be able to make profiles about all registered bidders.

Note that the device has taken $I_b > I$, since during computing $(g^{r_\alpha})_{\alpha=1}^{I_b}$ the order of I final entries is unknown. The situation is much easier, when the order is predetermined. This happens for the auction scheme from [4], where each bidder publishes a *sequence* of ElGamal ciphertexts $(\mu y^r, g^r)$. Then the exponents r might be generated from $\mathcal{R}\big(x([u]Y_a)\big)$ and appropriately incremented, i.e., techniques from the first attack in Section 2 could be applied.

In the second scheme by Omote and Miyaji [22] the situation is completely different from the one in [21]. Each y_i, as well as g, is raised to the same exponent $r \cdot s$: first these numbers are raised to power r by RM, then the results

are raised to power s by AM. Such a procedure prevents the attack described above.

Kleptographic attack on bidder(s). The attack below works for both [21], and [22] (in the latter case exponent rs instead of r_i would be used). In an auction, the bidder $\mathcal{B}_i$ uses g^{r_i} and $y_i^{r_i}$ provided by AM. To make a bid m_i, the bidder $\mathcal{B}_i$ must show a signature of knowledge of his/her secret exponent x_i (see [5]). The signature is a pair (c, s) such that:

$$c = h\big(m_i||y_i^{r_i}||g^{r_i}||(g^{r_i})^s \cdot (y_i^{r_i})^c\big), \tag{10}$$

where h is a hash function. To determine (c, s), a bidder, which knows x_i such that $y_i = g^{x_i}$, chooses at random some R and sets:

$$\begin{cases} c &= h\big(m_i||y_i^{r_i}||g^{r_i}||(g^{r_i})^R\big), \\ s &= R - cx_i. \end{cases} \tag{11}$$

According to the protocol "anybody can check the validity" of the signature, using equality (10). Since the signature is publicly verifiable, anyone might obtain

$$(g^{r_i})^R = (g^{r_i})^s \cdot (y_i^{r_i})^c.$$

Assume, like in the second attack in Section 2, that bidder's device contains some unique key K set by the manufacturer. Let the exponent R be obtained from $\mathcal{R}\big(H(K)\big)$, and after making the bid let the key is immediately changed by substituting

$$K := \tilde{H}(K) \qquad \text{for} \quad \tilde{H} \neq H.$$

Consequently, having a database of initial values of keys K manufacturer Mallet might perform a series of substitutions $K := \tilde{H}(K)$ on these keys. After some time he will find the key yielding exponent R for $(g^{r_i})^R$. Then, having c, s from bidder's signature, Mallet would get secret exponent x_i from the second equation in (11).

Note that in the above attack the device might transmit some additional message to Mallet. Let $\Delta \in \{0, \ldots, 2^k - 1\}$, for fixed $k \in \{10, \ldots, 20\}$, be such a message to be transferred in a single bid. Note that all bids of a given bidder have the same g^{r_i} within a single auction. Hence for each bidder whose private key is not known yet, Mallet prepares an array of entries

$$\big((g^{r_i})^\alpha, \alpha\big), \qquad \alpha = 1, \ldots, 2^k - 1,$$

and sorts it according to the first coordinate. Let a contaminated device substitute

$$R := \mathcal{R}\big(H(K)\big) + \Delta \bmod \operatorname{ord} g.$$

Then Mallet, instead of checking whether $(g^{r_i})^R$ from signature's of knowledge verification is equal to $(g^{r_i})^{R'}$, where

$$R' = \mathcal{R}\big(H(K)\big) \bmod \operatorname{ord} g$$

for some K delivered from his database, checks whether

$$(g^{r_i})^R \cdot \left((g^{r_i})^{R'}\right)^{-1} \bmod p$$

is in the array. If yes, then he gets Δ being the second coordinate of this entry. Having correspondence between the private key of the bidder and the initial key K of the bidder's device, Mallet finds the next messages much faster. This procedure works, since observed $(g^{r_i})^R$ are extremely sparsely distributed in $\langle g \rangle$.

4. Continuous double auctions

An auction protocol [32] and its improved version from [30], both propose auction schemes suitable for so called *Continuous Double Action* (cf. Appendix). In the schemes, participants are bidders (called *traders* in [32]) and two authorities: one for registration and one for leading auctions.

In [32] the registration phase is interesting from a kleptographic point of view. Each trader T generates his pair of temporary signing keys (TS_T, TP_T), and submits the public key TP_T to the Registration Manager. These keys are used in the bidding's process. The signature scheme used by traders is unspecified in the protocol. If it is probabilistic, information might leak kleptographically through signatures (cf. [35]). If RSA scheme is used, then an attack against generation of RSA modulus n might be applied. A recent kleptographic attack [37] exploits a pair of twisted elliptic curves over a binary field. It reveals to Mallet about $\frac{\log_2 n}{4}$ most significant bits of one of the prime components, hence it suffices for Mallet to use Coppersmith's algorithm (cf. [6], [7]) to effectively factor n.

To minimize the size of Mallet's "public" keys stored in a device we shall modify the solution from [37], and use a single elliptic curve $E_{a,b}(\mathbb{F}_p)$

$$y^2 = x^3 + ax + b \tag{12}$$

defined over large prime field $\mathbb{F}_p$. Let $G \in E_{a,b}(\mathbb{F}_p)$ be a base point of a large prime order. It has been proved (see [15]) that for prime fields the x-coordinates of points from the group $\langle G \rangle$ are uniformly distributed in the set $\{0, 1, \ldots, p-1\}$.

From Hasse Theorem we know that

$$\left(\sqrt{p} - 1\right)^2 < \#E_{a,b}(\mathbb{F}_p) < \left(\sqrt{p} + 1\right)^2.$$

According to Conjecture A from [9], in Hasse interval there is a fraction of $\approx \frac{c}{\ln p}$ curves having prime order, where c is a constant lying between about 0.44 and 0.62. Hence it is quite easy to generate such an elliptic curve, or simply take one from the NIST list [20]. Let

$$Y = [s]G \qquad \text{for} \quad s \in_U \{2, 3, \ldots, \operatorname{ord} G - 1\},$$

where $\in_U$ stands for a choice with uniform probability distribution. From Hasse Theorem and from $E_{a,b}(\mathbb{F}_p)$ choice of prime order we get $\frac{p}{\operatorname{ord} G} \approx 1$.

Mallet's public key is then composed of p, $E_{a,b}$ (NIST curves have very short a), and x-coordinates of the points G, Y (for any $P \in E_{a,b}$ the same x-coordinate has only the point $-P$). The key is stored in user's device. The device will calculate

$$Q = [u]G \qquad \text{for} \quad u \in_U \{1, \ldots, \operatorname{ord} G - 1\}$$

and shall use $x\big([u]Y\big)$ as a private value s_{priv} for RSA modulus generation (see [37]). Number $x(Q)$ will be transferred (since $\frac{p}{\operatorname{ord} G} \approx 1$, it takes one of about $\frac{p}{2}$ possible values).

Note that after opening another contaminated device with the same Mallet's public key (in particular, with the same equation (12), one can determine that the value $x(Q)$ transferred gives a quadratic residue on the right hand side of (12). It means that the value is the x-coordinate of some point of that elliptic curve. This property is strongly undesirable from the kleptographic point of view, therefore Mallet will "camouflage" the transfer.

Let

$$s_{publ} := x(Q) + \big[H(x([u]Y)) \bmod \Delta\big] \bmod p$$

for some fixed Δ (say $\Delta = 2^{16}$). Recall that $x(Q)$ are uniformly distributed, and note that values $\big[H(x([u]Y)) \bmod \Delta\big]$ should appear random, uncorrelated with $x(Q)$ (unlike for example $[-x(Q) \bmod \Delta]$ are correlated with $x(Q)$). Thus Mallet might expect that s_{publ} are uniformly distributed in $\{0, 1, \ldots, p-1\}$. Moreover, due to pseudorandom jump " $+\big[H(x([u]Y)) \bmod \Delta\big]$ " s_{publ} can be a quadratic nonresidue. Clearly, Δ of a moderate size is necessary: for *every* s_{publ} the distance between s_{publ} and the "greatest" quadratic residue not "greater" than s_{publ} would be smaller than Δ, whereas with probability very close to $2^{-(\Delta-1)}$ should equal at least Δ (cf. [23], p is large). For small Δ this deviation would be statistically noticeable.

Now it remains to *hide* the property "$s_{publ} < p$". Like in Section 3 the PBRM shall be used. Although this method randomly discards some of the s_{publ}, the value s_{publ} is set outside the main loop of the attack [37], so the additional computations are easy. Then such value s_{publ} might be used instead of concatenation of $x(Q_a)$ with one bit of $y(Q_a)$ exploited in [37].

To decode the secret, at first Mallet undoes the PBRM from s_{publ}. Then for all X from $\{s_{publ} - (\Delta - 1), \ldots, s_{publ}\}$ giving quadratic residues on the right hand side of (12) checks the condition $f(X) = s_{publ}$, where

$$f(X) = X + \Big[H\big(x([s]P_X)\big) \bmod \Delta\Big] \bmod p$$

and P_X is a point with the x-coordinate equal to X. Since f is a pseudorandom function and s_{publ} is set, we suppose that even for $\Delta = 2^{16}$ there are only a few candidates X for $x(Q)$, say δ candidates. For each one Mallet retrieves s_{publ} and

repeats the factoring procedure from [37], thus his average computational effort grows only by a factor of δ, but key-size in the device has decreased.

Auction scheme presented in [30] is based on the idea similar to [32], but the protocol is in fact designed from scratch. Group signatures from [1] are used and subsequent bids are unlinkable at least in case of honest implementation of software/hardware components.

The public key in the group signature scheme [1] is $(n, a, a_0, y = g^x, g, h)$, where n is an RSA modulus, and a, a_0, g, h are quadratic residues (i.e., numbers being squares) randomly selected from $\mathbb{Z}_n^*$. Each bidder b_i possesses a membership certificate $[B_i, e_i]$ such that $a^{x_i} a_0 = B_i^{e_i} \bmod n$ for some exponent x_i, which is kept secret by the b_i. When a bid m is being submitted, the bidder (or more precisely, a hardware or software on his behalf) chooses w, r_1, r_2, r_3, r_4 at random and computes his group signature for m:

1. $T_1 = B_i y^w \bmod n,\ T_2 = g^w \bmod n,\ T_3 = g^{e_i} h^w \bmod n,$
2. $d_1 = T_1/(a^{r_2} y^{r_3}) \bmod n,\ d_2 = T_2^{r_1}/(g^{r_3}) \bmod n,$
   $d_3 = g^{r_4} \bmod n,\ d_4 = g^{r_1} h^{r_4} \bmod n,$
3. $c = \mathcal{H}\big(g||h||y||a_0||a||T_1||T_2||T_3||d_1||d_2||d_3||d_4||m\big),$
4. $s_1 = r_1 - c(e_i - 2^{\xi_1}),\ s_2 = r_1 - c(x_i - 2^{\lambda_1}),$
   $s_3 = r_3 - c e_i w,\ s_4 = r_4 - cw,$

where ξ_1, λ_1 are some constants. Note that s_1, s_2, s_3, s_4 are calculated in $\mathbb{Z}$. The bid m and the signature

$$(c, s_1, s_2, s_3, s_4, T_1, T_2, T_3) \tag{13}$$

are sent to the auctioneer for verification, and next to the bulletin board. The verification makes use of congruences:

$$\begin{aligned} &d_1 = \Big(a_0^c T_1^{(s_1 - c2^{\xi_1})}\Big) \Big/ \Big(a^{s_2 - c2^{\lambda_1}} y^{s_3}\Big) \bmod n, \quad d_2 = \Big(T_2^{s_1 - c2^{\xi_1}}\Big) \Big/ (g^{s_3}) \bmod n, \\ &d_3 = T_2^c g^{s_4} \bmod n, \qquad\qquad d_4 = T_3^c g^{s_1 - c2^{\xi_1}} h^{s_4} \bmod n, \end{aligned} \tag{14}$$

and might be performed by anyone.

The scheme concerned is perfect for an attack on software implementation. Let

$$y_M = g^{x_M} \bmod n$$

be the Mallet's public key. Note that to set up the key Mallet does not need to know the factorization of n. Let contaminated bidder's software choose w for T_2 uniformly at random. Let

$$r'_j, \quad j = 1, 2, 3, 4,$$

be exponents produced by pseudorandom number generator $\mathcal{R}$ on the seed $y_M^w \bmod n$. The software adds some k-bit values Δ_j (for fixed $k \in \{30, \ldots, 40\}$) to the exponents, i.e.,

$$r_j := r'_j + \Delta_j,$$

and then uses these r_j in points 4 and 4 of signature generation.

Mallet takes signature (13) from the bulletin board, and calculates d_1, d_2, d_3, d_4 by (14). Next he calculates r'_j from $\mathcal{R}\big((T_2)^{x_M} \bmod n\big)$, and gets

$$g^{\Delta_4} = d_3/g^{r'_4} \bmod n$$

at first. Using the Shank's Baby Step Giant Step algorithm [25] he calculates Δ_4 in a number of multiplications bounded by $2^{\frac{k}{2}+1}$. Hence he gets r_4. In a similar way he calculates Δ_1, r_1 from d_4, then Δ_3, r_3 from d_2, and finally Δ_2, r_2 from d_1. Having r_j he immediately obtains w from s_4, next B_i from T_1 and e_i, x_i from s_3, s_2 correspondingly. As a result, Mallet gets all necessary data to forge bidder's group's member signature. Moreover, all bids of bidder b_i might be linked now, hence a profile of b_i might be made. On top of that, in $(\Delta_1||\Delta_2||\Delta_3||\Delta_4)$ the bidder's software might transfer at least 120 bits of any interesting data, for example bidder's real ID. Notice that all this might be done with a single bid.

In fact, the above procedure is an attack on group signatures from [1].

The same method of deploying Shank's algorithm might be applied against a sealed bid auction scheme from [8], where public values of so called $V\Sigma S$ *signatures* are broadcasted by a bidder to auction servers.

5. Conclusions

To eliminate any possibility of a covert channel, the output should be entirely verifiable for device's owner, while simultaneously completely unpredictable for others, including the manufacturer. For a method meeting these expectations we refer the reader to [2, Sect. V].

REFERENCES

[1] ATENIESE, G.—CAMENISCH, J.—JOYE, M.—TSUDIK, G.: *A practical and provably secure coalition-resistant group signature scheme*, in: Advances in Cryptology——CRYPTO '00 (M. Bellare, ed.), Lecture Notes in Comput. Sci., Vol. 1880, Springer--Verlag, Berlin, 2000, pp. 255–270,
http://www.zurich.ibm.com/~jca/papers/group2000.pdf.

[2] BORZĘCKI, P.—KABAROWSKI, J.—KUBIAK, P.—KUTYŁOWSKI, M.—ZAGÓRSKI, F.: *Kleptographic weaknesses in Benaloh-Tuinstra protocol*, in: 2nd International Conference on Systems and Networks Communications—ICSNC '06, IEEE Comput. Soc., Washington, DC, USA 2006, pp. 26–31.

[3] BOYD, C.—MAO, W.: *Security issues for electronic auctions*, Technical Report HPL-2000-90, Trusted E-Services Laboratory, HP Laboratories Bristol, 2000,
http://www.hpl.hp.com/techreports/2000/HPL-2000-90.html.

[4] BRANDT, F.—SANDHOLM, T.: *Efficient privacy-preserving protocols for multi-unit auctions*, in: 9th International Conference—FC '05 (A. S. Patrick, M. Yung, eds.), Lecture Notes in Comput. Sci., Vol. 3570, Springer-Verlag, Berlin, 2005, pp. 298–312, http://www.cs.cmu.edu/~sandholm/privacy_preserving_multi-unit_auctions.lncs05.pdf.

[5] CAMENISCH, J.—STADLER, M.: *Efficient group signature schemes for large groups (extended abstract)*, in: Advances in Cryptology—CRYPTO '97 (B. S. Kaliski, Jr., ed.), Lecture Notes in Comput. Sci., Vol. 1294, Springer-Verlag, Berlin, 1997, pp. 410–424, http://citeseer.ist.psu.edu/14628.html.

[6] COPPERSMITH, D.: *Finding a small root of a bivariate integer equation; factoring with high bits known*, in: Advances in Cryptology—EUROCRYPT '96 (U. M. Maurer, ed.), Lecture Notes in Comput. Sci., Vol. 1070, Springer-Verlag, Berlin, 1996, pp. 178–189, http://dsns.csie.nctu.edu.tw/research/crypto/HTML/PDF/E96/178.PDF.

[7] CORON, J.-S.: *Finding small roots of bivariate integer polynomial equations revisited*, in: Advances in Cryptology—EUROCRYPT '04 (Ch. Cachin, J. Camenisch, eds.), Lecture Notes in Comput. Sci., Vol. 3027, Springer-Verlag, Berlin, 2004, pp. 492–505, http://www.eleves.ens.fr/home/coron/publications/bivariate.pdf.

[8] FRANKLIN, M. K.—REITER, M. K.: *The design and implementation of a secure auction service*, IEEE Trans. Software Eng. **22** (1996), 302–312, http://citeseer.ist.psu.edu/franklin95design.html.

[9] GALBRAITH, S.—MCKEE, J.: *The probability that the number of points on an elliptic curve over a finite field is prime*, J. London Math. Soc. **62** (2000), 671–684, http://www.isg.rhul.ac.uk/~sdg/cm.ps.gz.

[10] GOGOLEWSKI, M.—KLONOWSKI, M.—KUBIAK, P.—KUTYŁOWSKI, M.—LAUKS, A.—ZAGÓRSKI, F.: *Kleptographic attacks on e-voting schemes*, in: Emerging Trends in Information and Communication Security—ETRICS '06 (G. Müller, ed.), Lecture Notes in Comput. Sci., Vol. 3995, Springer-Verlag, Berlin, 2006, pp. 494–508.

[11] HARKAVY, M.—TYGAR, J. D.—KIKUCHI, H.: *Electronic auctions with private bids*, in: Proc. of the 3rd USENIX Workshop on Electronic Commerce, Vol. 3, USENIX Association, Berkeley, CA, USA, 1998, pp. 61–74, http://www.usenix.org/publications/library/proceedings/ec98/full_papers/harkavy/harkavy.pdf.

[12] JUELS, A.—SZYDLO, M.: *A two-server, sealed-bid auction protocol*, in: Financial Cryptography—FC '02 (M. Blaze, ed.), Lecture Notes in Comput. Sci., Vol. 2357, Springer-Verlag, Berlin, 2002, pp. 72–86, http://www.rsasecurity.com/rsalabs/staff/bios/ajuels/publications/two-server/two-server-auction.pdf.

[13] KING, B.: *A point compression method for elliptic curves defined over GF(2^n)*, in: Public Key Cryptography—PKC '04 (F. Bao et al., eds.), Lecture Notes in Comput. Sci., Vol. 2947, Springer-Verlag, Berlin, 2004, pp. 333–345.

[14] LIPMAA, H.: *Cryptology pointers: Cryptographic auctions* [online], http://www.adastral.ucl.ac.uk/~helger/crypto/link/protocols/auctions.php.

[15] EL MAHASSNI, E.—SHPARLINSKI, I.: *On the uniformity of distribution of the elliptic curve ElGamal signature*, Finite Fields Appl. **8** (2002), 589–596, http://www.comp.mq.edu.au/~igor/EC-ElGamal-Distr.ps.

[16] MENEZES, A. J.—VAN OORSCHOT, P. C.—VANSTONE, S. A: *Handbook of Applied Cryptography*. CRC Press, Boca Raton, Florida, 1996, http://www.cacr.math.uwaterloo.ca/hac/.

[17] MILLER, V. S.: *The Weil pairing, and its efficient calculation*, J. Cryptology **17** (2004), 235–261, http://dx.doi.org/10.1007/s00145-004-0315-8.

[18] MÖLLER, B.: *A public-key encryption scheme with pseudo-random ciphertexts*, in: Computer Security—ESORICS '04 (P. Samarati et. al., eds.), Lecture Notes in Comput. Sci.,

Vol. 3193, Springer-Verlag, Berlin, 2004, pp. 335–351,
http://www.bmoeller.de/pdf/pke-pseudo-esorics2004.pdf.

[19] NAOR, M.—PINKAS, B.—SUMNER, R.: *Privacy preserving auctions and mechanism design*, in: Proc. of the 1st ACM Conference on Electronic Commerce—EC '99, ACM, New York, NY, USA, 1999, pp. 129–139,
http://citeseer.ist.psu.edu/naor99privacy.html.

[20] NIST, *Recommended elliptic curves for federal government use*, 1999,
http://csrc.nist.gov/CryptoToolkit/dss/ecdsa/NISTReCur.pdf.

[21] OMOTE, K.—MIYAJI, A.: *A practical English auction with one-time registration*, in: 6th Australasian Conference—ACISP '01 (V. Varadharajan, Y. Mu, eds.), Lecture Notes in Comput. Sci., Vol. 2119, Springer-Verlag, Berlin, 2001, pp. 221–234,
http://citeseer.ist.psu.edu/447131.html.

[22] OMOTE, K.—MIYAJI, A.: *A practical English auction with simple revocation*, IEICE Trans. Fundamentals **E85-A** (2002), 1054–1061,
http://citeseer.ist.psu.edu/omote02practical.html.

[23] PERALTA, R.: *On the distribution of quadratic residues and nonresidues modulo a prime number*, Math. Comp. **58** (1992), 433–440,
http://citeseer.ist.psu.edu/peralta92distribution.html.

[24] SEROUSSI, G.: *Compact representations of elliptic curve points over GF(2^n)*, Tech. Report HPL-98-94R1, HP Laboratories, September 2000,
http://www.hpl.hp.com/techreports/98/HPL-98-94R1.html.

[25] SHANKS, D.: *Class number, a theory of factorization, and genera*, in: Proc. Sympos. Pure Math., Vol. 20, American Mathematical Society, Providence, R. I., 1971, pp. 415–440.

[26] STUBBLEBINE, S. G.—SYVERSON, P. F.—GOLDSCHLAG, D. M.: *Unlinkable serial transactions: protocols and applications*, ACM Trans. Inf. Syst. Secur. **2** (1999), 354–389,
http://www.stubblebine.com/99tissec-ust.pdf.

[27] TREVATHAN, J.: *Electronic auction research,*
http://www.cs.jcu.edu.au/~jarrod/auctionResearch.htm.

[28] TREVATHAN, J.: *Electronic auctions — literature review*, June 2005,
http://www.cs.jcu.edu.au/~jarrod/lit.ps.

[29] TREVATHAN, J.—GHODOSI, H.—READ, W.: *Design issues for electronic auctions*, 2004, http://www.cs.jcu.edu.au/~jarrod/publications/design.pdf.

[30] TREVATHAN, J.—GHODOSI, H.—READ, W.: *An anonymous and secure continuous double auction scheme*, in: Proc. of the 39th Annual Hawaii International Conference on System Sciences—HICSS '06, IEEE Comput. Soc., Washington, DC, USA, 2006,
http://citeseer.ist.psu.edu/trevathan05anonymous.html.

[31] VICKREY, W.: *Counterspeculation, auctions, and competitive sealed tenders*, Journal of Finance **16** (1961), 8–37,
http://libeccio.dia.unisa.it/EC04/Biblio/e-documents/Papers/vickrey61.pdf.

[32] WANG, CH.—LEUNG, H.-F.: *Anonymity and security in continuous double auctions for internet retails market*, in: Proc. of the 39th Annual Hawaii International Conference on System Sciences—HICSS '04, Vol. 7, IEEE Comput. Soc., Washington, DC, USA, 2004,
http://csdl2.computer.org/comp/proceedings/hicss/2004/2056/07/205670180b.pdf.

[33] YOUNG, A.—YUNG, M.: *The dark side of "black-box" cryptography, or: Should we trust capstone?* in: Advanced in Cryptology—CRYPTO '96 (N. Koblitz, ed.), Lecture Notes in Comput. Sci., Vol. 1109, Springer-Verlag, Berlin, 1996, pp. 89–103,
http://citeseer.ist.psu.edu/young96dark.html.

[34] YOUNG, A.—YUNG, M.: *Kleptography: Using cryptography against cryptography*, in: Advanced in Cryptology—EUROCRYPT '97 (W. Fumy, ed.), Lecture Notes in Comput. Sci., Vol. 1233, Springer-Verlag, Berlin, 1997, pp. 62–74,
http://dsns.csie.nctu.edu.tw/research/crypto/HTML/PDF/E97/62.PDF.

[35] YOUNG, A.—YUNG, M.: *Bandwidth-optimal kleptographic attacks*, in: Cryptographic Hardware and Embedded Systems—CHES '01 (Ç. Kaya Koç et al. eds.), Lecture Notes in Comput. Sci., Vol. 2162, Springer-Verlag, Berlin, 2001, pp. 235–250.

[36] YOUNG, A.—YUNG, M.: *Malicious cryptography: Kleptographic aspects*, in: Topics in Cryptology—CT-RSA '05 CT-RSA (A. Menezes, ed.), Lecture Notes in Comput. Sci., Vol. 3376, Springer-Verlag, Berlin, 2005, pp. 7–18, `http://www.cs.utsa.edu/~shxu/CS6973-Fall2005/papers/cv-4.pdf`.

[37] YOUNG, A.—YUNG, M.: *A space efficient backdoor in RSA and its applications*, in: Selected Areas in Cryptography—SAC '05 (B. Preneel, S. E. Tavares, eds.), Lecture Notes in Comput. Sci., Vol. 3897, Springer-Verlag, Berlin, 2005, pp. 128–143.

APPENDIX — Types of auctions

In general, a few main types of electronic auctions might be distinguished, below we follow the description from [28].

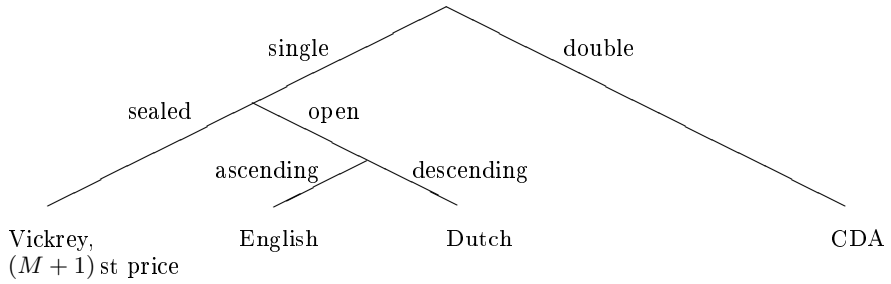

FIGURE 1. Classification of electronic auction schemes.

Usually in an auction protocol we have three roles: a bidder, a seller, and an auctioneer. In a single auction there is one seller and many buyers, whereas in double auction there are many sellers. In a sealed bid auction bidders simultaneously submit their bids so that the bids remain hidden until the bidding period is closed. In an open auction the bids are known to all bidders yet in the bidding period. In ascending auctions the price starts low and competing bidders are pushing up the price until nobody is ready to bid higher or until the bidding period is closed. In descending auction the price starts high and is reduced successively.

A Vickrey auction [31] is a sealed bid auction in which a bidder offering the highest price wins and pays the second highest bid. An $(M+1)$st price auction is a generalization of the previous one: there are M units for sale, hence M bidders offering the highest bids win and pay the $(M+1)$st highest bid (thus a Vickrey auction is exactly an $(M+1)$st price auction with $M=1$). An English auction

is an open bid, first price auction. Due to open bids the first price usually does not differ much from the second one. In a Dutch auction the auctioneer starts at a high price and announces successively lower prices. The first bidder to bid wins and pays the current price at the time he bids. A *Continuous Double Auction* (CDA) is an open bid auction where both sellers and buyers submit bids for the sale and purchase of a single commodity.

Received September 28, 2007

Marcin Gogolewski
Faculty of Mathematics and Computer Science
Adam Mickiewicz University
ul. Umultowska 87
PL-61–614 Poznań
POLAND
E-mail: marcin.gogolewski@amu.edu.pl

Marcin Gomułkiewicz
Jarosław Grząślewicz
Przemysław Kubiak
Mirosław Kutyłowski
Anna Lauks
Institute of Mathematics and Computer Science
Wrocław University of Technology
Janiszewskiego 14
PL-50–370 Wrocław
POLAND
E-mail: marcin.gomulkiewicz@gmail.com
jaroslaw.grzaslewicz@gmail.com
przemyslaw.kubiak@pwr.wroc.pl
miroslaw.kutylowski@pwr.wroc.pl
anna.lauks@pwr.wroc.pl