CHEATING AT MENTAL POKER

Don Coppersmith
IBM Research
Yorktown Heights, NY 10598

We review the "mental poker" scheme described by Shamir, Rivest and Adleman [SRA]. We present two possible means of cheating, depending on careless implementation of the SRA scheme. One will work if the prime p is such that $p-1$ has a small prime divisor. In the other scheme, the names of the cards "TWO OF CLUBS" have been extended by random-looking bits, chosen by the cheater.

Background

In 1979 Shamir, Rivest and Adleman [SRA] proposed a scheme for playing "mental poker," i.e. play a fair poker game over the telephone between two mutually suspicious players. As a corollary, their paper gave a practical method for exchanging secret information over a public channel. (This method of exchanging information is still viable, and nothing in this paper affects its usefulness.)

In their scheme, players A and B agree on a large prime p. They create a deck of cards c_i, $i=1,2,\ldots,52$, where, for example, c_1 might be the EBCDIC coding of the characters "TWO OF CLUBS". Player A creates two secret numbers $a, \bar{a}$, such that $a\bar{a} \equiv 1(\bmod p)$; Player B similarly creates secret numbers $b, \bar{b}$. Player A shuffles the deck, encodes each card by raising to the a power $(\bmod p)$, and sends the deck to Player B. (At this point, B sees $c_{\pi(i)}^{a}(\bmod p)$, where π denotes the permutation or shuffle applied by A.) Player B selects five cards for A, say $c_{A1}^{a}(\bmod p), \ldots, c_{A5}^{a}(\bmod p)$, and returns them to A, who decodes them by raising to the $\bar{a}$ power $(\bmod p)$. B also selects five cards for himself, and adds his own encryption by raising to the b power $(\bmod p)$. He sends the resulting cards, $c_{B1}^{ab}(\bmod p), \ldots, c_{B5}^{ab}(\bmod p)$, to A. In turn, A raises B's cards to the $\bar{a}$ power, obtaining $c_{Bi}^{ab\bar{a}} \equiv c_{Bi}^{b}(\bmod p)$, and returns them to B. Finally B raises these cards to the $\bar{b}$ power $(\bmod p)$ to obtain $c_{Bi}^{b\bar{b}} \equiv c_{Bi}(\bmod p)$, his own hand in the clear.

Thus is the hand dealt. Betting proceeds as usual. At the end of the game, the secret keys are revealed, so that the hands are made known to both sides.

Method 1: when p-1 has a small factor.

The first method of cheating is a generalization of the "quadratic residue" trick, due to Lipton [DDDHL].

Suppose that $p-1$ is divisible by a small integer q, say $30 < q < 10^{12}$.

The multiplicative group of integers $(\bmod p)$ is denoted by Z_p^*. It is isomorphic to the additive group of integers $(\bmod p-1)$, Z_{p-1}. (There are several isomorphisms available, and we can select one by selecting a generator g of the multiplicative group.) For each integer q dividing $p-1$ there is a projection from Z_{p-1} onto Z_q. Composing these two maps, to each $x \not\equiv 0(\bmod p)$ we can associate an element $(\bmod q)$, which we will call $\log x(\bmod q)$, suppressing

H.C. Williams (Ed.): Advances in Cryptology - CRYPTO '85, LNCS 218, pp. 104-107, 1986.

the dependence on g. The Pohlig-Hellman technique ([PH], *attributed by them to* Roland Silver) enables us to compute $\log x(\bmod q)$ for the price of $O(\log p + \sqrt{q})$ multiplications $(\bmod p)$. For q in the range given, this is a feasible amount of computation.

Suppose Player B sees the cards before they are encrypted. Then he can determine $\{ \log c_i(\bmod q), 1 \leq i \leq 52\}$. Now he receives the shuffled and encrypted deck from A. Again, he determines $\{ \log(c^a_{\pi(i)})(\bmod q) \equiv a \log c_{\pi(i)}(\bmod q)\}$. By comparing the distributions of the logarithms, before and after encryption, B can usually determine the value of $a(\bmod q)$. Thus he can recover $\{ \log c_{\pi(i)}(\bmod q)\}$. This gives him some information about the permutation π: he can tell which cards are which, up to ambiguities caused when two logarithms are the same: $\log c_i(\bmod q) = \log c_j(\bmod q)$. The expected number of uniquely determined cards is about $52(e^{-51/q})$; for $q > 30$ one expects to have at least nine cards uniquely determined.

Finally, if we choose our prime p uniformly at random, we will have some prime q, $30 < q < 10^{12}$, dividing $p - 1$ about eighty-seven percent of the time.

Conclusion 1: If you don't want cheating, choose your primes p to be of the form $p = 2q + 1$, q prime, so that the cheater can only tell the difference between quadratic residues and non-residues. Also, append bits so that all the cards are quadratic residues, to block even that information from the cheater.

Method 2: when the cards are padded by random bits.

The string "THREE OF DIAMONDS" in EBCDIC is very short: only seventeen characters or 136 bits. Our prime p cannot be this short, because efficient techniques exist for finding logarithms modulo primes this small [WM], [Adl], [COS]. So suppose the EBCDIC strings are padded out with random bits, in accordance with good cryptographic practice. (Note: the original paper [SRA] did not suggest such padding.) Suppose these bits occupy half the description of the cards.* Suppose also that Player B is allowed to select these "random bits". Then he can cheat.

Let the i^{th} card be given by $c_i = s_i + r_i < p$, where s_i is the EBCDIC coding of the card's name in English, left-adjusted in the representation of the integer, and r_i is the "random" portion, constricted by $0 \leq r_i \leq \sqrt{p}$.

Player B fixes the representation of c_1 as "TWO OF CLUBS" padded with truly random bits. Now for each i, $2 \leq i \leq 26$, B tries to select r_{2i-1}, r_{2i} so that the resulting integers c_{2i-1}, c_{2i} satisfy

$$c_1^i c_{2i-1} \equiv c_{2i}(\bmod p)$$
$$(c_1^i s_{2i-1}(\bmod p)) + (c_1^i(\bmod p))r_{2i-1} = s_{2i} + r_{2i} + tp,$$

where r_{2i-1}, r_{2i} and t are unknown integers less than $\sqrt{p}.$ This is just a linear diophantine equation, easily solved, for example, by a basis reduction algorithm; see [Lag], [LLL] for the techniques involved.

* An interesting problem remains: what if the "random bits" occupy only 1/3 or 1/4 of the description? Can a similar scheme be implemented?

Now A shuffles and encrypts the deck, and sends the entire deck to B. Recall that B sees $c^a_{\pi(i)}(\bmod p)$. Notice that since $c_1^i c_{2i-1} \equiv c_{2i}(\bmod p)$, then the same relation holds among the encrypted cards: $(c_1^a)^i(c^a_{2i-1}) \equiv (c^a_{2i})(\bmod p)$. So B tries each of $52 \times 51 = 2652$ ordered pairs of cards in the encrypted shuffled deck, computing $(c^a_{\pi(j)})^2(c^a_{\pi(k)})(\bmod p)$ and comparing the results to the remaining 50 cards. On finding a match, $(c^a_{\pi(j)})^2(c^a_{\pi(k)}) \equiv (c^a_{\pi(\ell)})(\bmod p)$, Player B has probably identified three cards: $\pi(j) = 1, \pi(k) = 3, \pi(\ell) = 4$. Now for $3 \leq i \leq 26$, $1 \leq m \leq 52$, $m \neq j,k,\ell$, compute $(c^a_{\pi(j)})^i(c^a_{\pi(m)})(\bmod p)$ and compare to the remaining cards; each match $(c^a_{\pi(j)})^i(c^a_{\pi(m)}) \equiv (c^a_{\pi(n)})(\bmod p)$, gives two more cards $\pi(m) = 2i - 1, \pi(n) = 2i$. At the cost of a few thousand multiplications $(\bmod p)$, B has recovered the permutation π, and can now select both hands quite maliciously.

Conclusion 2: If you're going to have "random padding," make sure your opponent doesn't select the random numbers.

Conclusion 3: The protocol is fairly fragile in the sense that seemingly innocuous changes (selection of p, padding with seemingly random bits) can allow for cheating. If you don't trust a man enough to play cards with him, don't play mental cards with him either.

Note: Goldwasser and Micali [GM] have proposed an alternate, more complicated protocol for mental poker, which is evidently more secure.

References

[Adl] L.M. Adleman, "A subexponential algorithm for the discrete logarithm problem with applications to cryptography," *Proc. 20th IEEE Found. Comp. Sci. Symp.* (1979), 55-60.

[COS] D. Coppersmith, A.M. Odlyzko and R. Schroeppel, "Discrete Logarithms in GF(p)," Research Report RC 10985, IBM T.J. Watson Research Center, Yorktown Heights, N.Y., 10598, February 14, 1985.

[DDDHL] R.A. DeMillo, G.I. Davida, D.P. Dobkin, M.A. Harrison and R.J. Lipton, *Applied Cryptology, Cryptographic Protocols, and Computer Security Models*, vol. 29, Proceedings of Symposia in Applied Mathematics, American Mathematical Society, 1983. Chapter 4.11, "Compromising Protocols."

[GM] S. Goldwasser and S. Micali, "Probabilistic Encryption & How To Play Mental Poker Keeping Secret All Partial Information," *Proc. 14th ACM Symposium on Theory of Computing* (1982), 365-377.

[Lag] J.C. Lagarias, "Knapsack Public Key Cryptosystems and Diophantine Approximation (Extended Abstract)," *Advances in Cryptology, Proceedings of Crypto 83*, (Ed.: D. Chaum), Plenum Press, New York, 1983, 289-301.

[LLL] A.K. Lenstra, H.W. Lenstra, Jr. and L. Lovasz, "Factoring Polynomials with Rational Coefficients," *Math. Annalen. 261* (1982), 515-534.

[PH] S.C. Pohlig and M. Hellman, "An improved algorithm for computing logarithms over *GF(p)* and its cryptographic significance," *IEEE Trans. Inform. Theory IT-24* (1978), 106-110.

[SRA] A. Shamir, R.L. Rivest and L.M. Adleman, "Mental Poker," MIT/LCS/TM-125, Laboratory for Computer Science, Massachusetts Institute of Technology, 545 Technology Square, Cambridge, MA 02139, February 1979.

[WM] A.E. Western and J.C.P. Miller, *Tables of Indices and Primitive Roots,* Royal Society Mathematical Tables, vol. 9, Cambridge Univ. Press, 1968.