Problem Set 4-2 Solutions

Problem 4-1. Vote all you want

We'll post a solution shortly.

Problem 4-2. Weak Randomness and ElGamal

You were asked to recover the secret key a from ElGamal signatures when k was not random. We used the notation from Menezes.

Below is an explanation based on the solution from John Giffin, Rachel Greenstadt, Peter Litwack, and Richard Tibbetts.

(a) Reusing k

Explain how to recover the secret key a when the signer reuses the same k for two different messages. Write an equation for a. That is, extract the private key from the public key and two (message, signature) pairs. State your assumptions.

From the definition of ElGamal, we have:

$$R = \alpha^k \pmod{p}$$
$$S = k^{-1}\{h(m) - a * R\} \pmod{p-1}$$

So:

$$k * S = h(m) - a * R \pmod{p-1}$$
$$k = \frac{h(m) - a * R}{S} \pmod{p-1}$$

Substituting for m_1 and m_2, we get two equations with two unknowns:

$$k = \frac{h(m_1) - a * R}{S_1} \pmod{p-1}$$
$$k = \frac{h(m_2) - a * R}{S_2} \pmod{p-1}$$

Solving:

$$\frac{h(m_1) - a * R}{S_1} = \frac{h(m_2) - a * R}{S_2} \pmod{p-1}$$
$$S_2 * h(m_1) - a * R * S_2 = S_1 * h(m_2) - a * R * S_1 \pmod{p-1}$$
$$S_2 * h(m_1) - S_1 * h(m_2) = a * (R * S_2 - R * S_1) \pmod{p-1}$$

$$a = \frac{S_2 * h(m_1) - S_1 * h(m_2)}{R * (S_2 - S_1)} \pmod{p-1}$$

This solution assumes that R and $S_2 - S_1$ are invertible. Note it is possible to solve the system of equations slightly differently so as not to divide by S_1 or S_2.

(b) Generating k with a linear congruential number generator

Louis Reasoner was unable to afford the advanced Lava Lamp random number generator – nor would it fit in his wallet-sized smartcard. Instead, he seeds a linear congruential number generator with a truly random number, k_0. During the signature of the ith message in Louis' scheme, $k =$

1. If $i = 0$, return a truly random number k_0 where $3 \leq k_0 \leq p - 2$ and $gcd(k_0,\ p-1) = 1$.
2. Otherwise let $k = k_{i-1}$.
3. Do $k \leftarrow k + z \pmod{p-1}$ until $gcd(k,\ p-1) = 1$.
4. Remember $k_i = k$ and return k.

z is a secret, truly random even number that is the same for every signature generation. This LCNG remembers its output for its next execution. Assume that p is a safe Sophie-Germain prime in that $p = 2q + 1$ where q is prime.

Explain how an adversary can break the system after seeing many sequential (message, signature) pairs. Write an equation for a. Note that your attack need not work 100% of the time, but it should work most of the time assuming that certain values are invertible $\pmod{p-1}$. State your assumptions about which values must be invertible and any other conditions for your attack to work. Discuss how many (message, signature) pairs you will need to break the scheme.

This problem was very tricky. Note that the loop in step 3 hardly ever executes more than once. It only loops when $gcd(k,\ p-1) \neq 1$. This is true only when k is even (never) or equal to q. For all practical purposes, you then can assume the loop executes exactly once.

A number of students made overly broad assumptions about invertibility. In fact, when too many variables are invertible, there is provably no solution with 3 consecutive messages. The final solution should be equivalent to:

$$a = \frac{h(m_2)s_1s_0 - 2s_0s_2h(m_1) + s_1s_2h(m_0)}{s_1s_2r_0 - 2s_0s_2r_1 + s_0s_1r_2} \bmod p-1$$

If you assume that the r_i and s_i values are invertible mod $p - 1 = 2q$, then the denominator above is **not** invertible. If a value mod $2q$ is invertible, then it must be odd. The denominator then becomes:

$$\begin{aligned}
&= \text{ODD} * \text{ODD} * \text{ODD} - 2\text{ODD} * \text{ODD} * \text{ODD} + \text{ODD} * \text{ODD} * \text{ODD} \bmod \text{EVEN} \\
&= \text{ODD} - \text{EVEN} + \text{ODD} \bmod \text{EVEN} \\
&= \text{EVEN} \bmod \text{EVEN} \\
&= \text{EVEN}
\end{aligned}$$

But we know that an even number cannot be invertible mod $2q$! Hence, solving this particular set of simultaneous equations requires that some values are not invertible. You can avoid this problem by using non-consecutive equations. Note that the multiplier 2 in the denominator comes directly from the 2 in $k_2 = k_0 + 2z$. If you instead solve a system of equations for messages 0, 1, and 3, then the multiplier becomes 3 which is odd. Then all the r_i and s_i can be invertible without forcing the denominator to become non-invertible mod $2q$.

Below is an explanation based on the solution from John Giffin, Rachel Greenstadt, Peter Litwack, and Richard Tibbetts. Find three consecutive messages whose signatures contain s values that are invertible $\pmod{p-1}$.

Since s is assumed to be invertible $\pmod{p-1}$, we can get the following equation for the value of k for the ith message:

$$s_i = k_i^{-1}\{h(m_i) - ar_i\} \pmod{p-1}$$
$$k_i = s_i^{-1}\{h(m_i) - ar_i\} \pmod{p-1}$$

Now look at the k generation algorithm. It takes the old k and adds z as many times as necessary to get k such that $gcd(k, p-1) = 1$. However, almost all the time, it will only need to add z once. Since p is a safe prime, $gcd(k, p-1)$ must be either 1, 2, q or $p-1$. We know that $gcd(k, p-1) \neq 2$ because k_0 was odd and z is even. Therefore, all k_i will be odd. We know that $gcd(k, p-1) \neq p-1$ because k is taken $\pmod{p-1}$. Thus, $gcd(k, p-1)$ must be either 1 or q. The only way that it can be q is if $k = q$. The odds of this are astronomically low. Thus, we can safely assume that $k_{i+1} = k_i + z$.

Since we know that $k_{i+1} = k_i + z$ nearly all the time, the following equations almost certainly hold for our three messages:

$$k_1 = s_1^{-1}\{h(m_1) - ar_1\} \pmod{p-1}$$
$$k_1 + z = s_2^{-1}\{h(m_2) - ar_2\} \pmod{p-1}$$
$$k_1 + 2z = s_3^{-1}\{h(m_3) - ar_3\} \pmod{p-1}$$

With a little algebra, we find:

$$z = s_2^{-1}\{h(m_2) - ar_2\} - s_1^{-1}\{h(m_1) - ar_1\} \pmod{p-1}$$

By substituting this expression for z and the previously derived expression for k_1 into the third equation, we get the following equation:

$$s_1^{-1}\{h(m_1) - ar_1\} + 2(s_2^{-1}\{h(m_2) - ar_2\} - s_1^{-1}\{h(m_1) - ar_1\}) = s_3^{-1}\{h(m_3) - ar_3\} \pmod{p-1}$$

Amazingly enough, this equation has just one unknown – the secret key a! With a few more turns of the algebraic crank, we get a closed form expression for a.

$$s_1^{-1}\{h(m_1) - ar_1\} + 2(s_2^{-1}\{h(m_2) - ar_2\} - s_1^{-1}\{h(m_1) - ar_1\}) = s_3^{-1}\{h(m_3) - ar_3\} \pmod{p-1}$$
$$2(s_2^{-1}\{h(m_2) - ar_2\}) - s_1^{-1}\{h(m_1) - ar_1\} = s_3^{-1}\{h(m_3) - ar_3\} \pmod{p-1}$$
$$2s_2^{-1}h(m_2) - 2s_2^{-1}r_2a - s_1^{-1}h(m_1) + s_1^{-1}r_1a = s_3^{-1}h(m_3) - s_3^{-1}r_3a \pmod{p-1}$$
$$2s_2^{-1}h(m_2) - s_3^{-1}h(m_3) - s_1^{-1}h(m_1) = 2s_2^{-1}r_2a - s_1^{-1}r_1a - s_3^{-1}r_3a \pmod{p-1}$$
$$2s_2^{-1}h(m_2) - s_3^{-1}h(m_3) - s_1^{-1}h(m_1) = (2s_2^{-1}r_2 - s_1^{-1}r_1 - s_3^{-1}r_3)a \pmod{p-1}$$
$$\frac{2s_2^{-1}h(m_2) - s_3^{-1}h(m_3) - s_1^{-1}h(m_1)}{2s_2^{-1}r_2 - s_1^{-1}r_1 - s_3^{-1}r_3} = a \pmod{p-1}$$

It follows that

$$a = \frac{h(m_3)s_2s_1 - 2s_1s_3h(m_2) + s_2s_3h(m_1)}{s_2s_3r_1 - 2s_1s_3r_2 + s_1s_2r_3} \bmod p-1$$

Of course, for this to work, $s_2s_3r_1 - 2s_1s_3r_2 + s_1s_2r_3$ must be invertible $\bmod p-1$. As we mentioned above, $gcd(n, p-1)$ will almost always be either 1 or 2. Thus, the statement n is invertible $\bmod p-1$ is for all practical purposes equivalent to being $n = 1 \bmod 2$. Thus, we need $s_2s_3r_1 - 2s_1s_3r_2 + s_1s_2r_3 = 1 \bmod 2$. This will be true if either all three terms of the sum are odd or if exactly one is odd. The second term is clearly even. Thus, the expression is invertible if exactly one of the first and third terms is odd. The first term is odd only when r_1, s_2, s_3 are odd and the third is odd just when r_3, s_1, s_2 are odd. Assuming the r_i, s_i values are chosen (pseudo) randomly, exactly one of the first and third terms will be odd with a probability of $2 * (\frac{1}{2}^3 * \frac{3}{4}) = \frac{3}{16}$ since the probability of the first term is odd is $\frac{1}{2}^3$. Then the probability the third term is even is $\frac{3}{4}$ because the s_2 term must be odd. There are 3 ways out of 4 for the product of r_3, s_1 to be even. If the denominator is odd, then there is only a $\frac{1}{q}$ chance that it is not invertible (that is, equal to q). So the probability of a given set of three message/signature pairs resulting in a solution is $\frac{3}{16} - \frac{1}{q}$. Note that $\frac{1}{q}$ is very small. $q \approx 10^{300}$ if $p \approx 1024$ bits. So we can essentially drop this term. The assumption of finding three consecutive messages with an invertible denominator is reasonable because all numbers that affect the values of s_i, r_i are either random or the output of a cryptographic hash (which should be close to random). Thus, s_i, r_i will be more or less random and should be odd with probability $\approx 50\%$.

It is likely that this attack will be possible with as few as 18 messages. Any group of 3 consecutive messages has a three in 16 chance of being suitable. If a given set does not work, simply ignore the first message and recalculate using the next message. There are also attacks using non-consecutive message/signature pairs.

Problem 4-3. Holiday

In honor of the 4-day weekend, there is no problem 3. Instead, we offer you a couple marginal jokes. For your protection, we have triple encrypted the answers with a special cipher. :-)

Why do cryptographers like to watch TV at night?
Orpnhfr bgurejvfr gurl jbhyq zvff gur cevzr gvzr arjf.
`Because otherwise they would miss the prime time news.`

What do you call it when an automaton sounds like it computes?
Nhgbzngncbrvn.
`Automatapoeia.`

This problem was a joke and was not supposed to be turned in. But a couple people turned it in for fun. Our encryption algorithm is known as "rot13." Each letter is rotated 13 letters. Hence, the encryption functions is its own inverse for 26-character alphabets. Rot13 is often used in USENET postings to hide controversial jokes from people who may be offended. One must hit Ctrl-X in the `trn` news reader to rot13 a message. In emacs, one can execute `M-x rot13-other-window`.