The University of Newcastle
School of Electrical Engineering and Computer Science

COMP3260/COMP6360 Data Security
Week 9 Workshop – 9 & 10 May 2019

Solutions

1. In 1985, T. Elgamal announced a public-key scheme based on discrete logarithms, closely related to the Diffie-Hellman key exchange technique introduced in 1976.
The global elements of ElGamal scheme are a q and α, where q is prime, and α is a primitive root of q. A user A selects a private key X_A and calculates a public key $Y_A = \alpha^{X_A} \bmod q$.

User A encrypts a plaintext M < q intended for user B as follows.
1. Choose a random integer k such that $1 \le k \le q-1$.
2. Compute $K = (Y_B)^k \bmod q$.
3. Encrypt M as the pair of integers (C_1, C_2) where $C_1 = \alpha^k \bmod q$ and $C_2 = K \cdot M \bmod q$.

User B receives the ciphertext (C_1, C_2) and recovers the plaintext as follows:
1. Compute $K = (C_1)^{X_B} \bmod q$. (i.e. use C_1 to recover K)
2. Compute $M = (C_2 \cdot K^{-1}) \bmod q$. (i.e. use K and C_2 to recover M)

Show that the system works. (i.e. show that the decryption process recovers the plaintext)

Solution:

We only need to show that $K = (C_1)^{X_B} \text{ mod } q$ and $M = (C_2 \cdot K - 1) mod\ q$.

$$\begin{aligned} K &= (C_1)^{X_B} \text{ mod } q \\ &= (\alpha^k \text{ mod } q)^{X_B} \text{ mod } q \\ &= \alpha^{kX_B} \text{ mod } q \\ &= (\alpha^{X_B} \text{ mod } q)^k \text{ mod } q \\ &= (Y_B)^k \text{ mod } q \\ &= K \end{aligned}$$

$$\begin{aligned} M &= (C_2 \cdot K^{-1}) \text{ mod } q \\ &= (K \cdot M \text{ mod } q) K^{-1} \text{ mod } q \\ &= K \cdot K^{-1} \cdot M \text{ mod } q \\ &= M \end{aligned}$$

2. In the RSA public-key encryption scheme, each user has a public key e and a private key d. Suppose Bob leaks his private key. Rather than generating a new modulus, he decides to generate a new public and a new private key. Is this safe?

Solution:
No, it is not safe.

An attacker will know the original e, d, n, as well as the new e' that Bob generates. The things an attacker does not know are the new private key d' generated by Bob, and the factors of n, p and q.

If we know d, then we also know $(ed - 1)$, which is a multiple of $\phi(n)$. That is because $(ed) \bmod (\phi(n)) = 1 \Rightarrow ed = k\phi(n) + 1$. There is a probabilistic algorithm (Las Vegas) that runs in expected polynomial time and yields the factorization $n = pq$ if $k\phi(n)$ is known. (If $\phi(n)$ itself is known, then we can find p and q using the quadratic formula, as shown in the additional thoughts below this solution)

To find p and q, knowing n, e, d, we start by finding x, where $x^2 \bmod n = 1$.

Recall that if we have $n = pq$ then $a \, mod \, n = 1$ implies $a \, mod \, p = 1$ and $a \, mod \, q = 1$. (Look back at the Chinese Remainder Theorem notes for more discussion on this)

This means that $x^2 \bmod n = 1$ implies that $x^2 \bmod p = 1$ and $x^2 \bmod q = 1$.
This can only be the case if and only if $x \bmod p = \pm 1$ and $x \bmod q = \pm 1$.

Solutions $x \bmod p = x \bmod q = x \bmod n = \pm 1$ are trivial.
If we could find one of the other two solutions

$x \bmod p = 1$, $x \bmod q = -1$ or
$x \bmod p = -1$, $x \bmod q = 1$

(note that in these two cases $x \bmod n \neq \pm 1$)

then we would have

$\gcd(x+1,n) = p$ or q and
$\gcd(x-1,n) = q$ or p

and it would be straightforward to find p and q (Euclid's algorithm for gcd).

The following is a probabilistic algorithm for finding x.
We pick a random number w such that $1 < w < n$. If $\gcd(w,n) > 1$, we have either p or q. If $\gcd(w,n) = 1$ then

$w^{ed-1} \bmod n = w^{k\phi(n)} \bmod n = 1$

We can write $(ed - 1) \bmod n$ as $2^s r$ where r is odd. Then we have

$w^{2^s r} \bmod n = 1$

We now need to find t, $0 < t \leq s$, such that $v^2 = w^{2^{t} r} \bmod n = 1$ and $v \neq \pm 1$.
We can use brute force to find t.

If there is no such t, we need to randomly generate a new w and start all over again. The probability that there will be such t for any given w is > ¾.
Thus on average we will need to generate < 4/3 random numbers w.

Additional thoughts:

If an attacker knew $\phi(n)$ itself, then the attacker could simply find d using $(ed) \bmod \big(\phi(n)\big) = 1$.
An attacker that knows $\phi(n)$ and n can also quickly determine p and q, as shown below:

$$\begin{aligned}\phi(n) &= (p-1)(q-1)\\ &= pq - p - q + 1\end{aligned}$$

Substituting $n = pq$:

$$\begin{aligned}\phi(n) &= n - p - q + 1\\ q &= n - \phi(n) - p + 1\end{aligned}$$

If we substitute this back into $n = pq$, we can express the equation as a quadratic in terms of p:

$$\begin{aligned}n &= pq\\ &= p(n - \phi(n) - p + 1)\\ &= pn - p\phi(n) - p^2 + p\\ p^2 - p(n - \phi(n) + 1) + n &= 0\end{aligned}$$

This can be solved using the quadratic formula

$$p = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Where

$$\begin{aligned}a &= 1\\ b &= -(n - \phi(n) + 1)\\ c &= n\end{aligned}$$

Since we only know $\phi(n) = \frac{(ed-1)}{k}$, we still need to use the probabilistic approach described in the initial solution.

3. In an RSA system, the public key of one user is (31, 3599). What is the user's private key?

Solution:

$n = 59 \times 61$
$\phi(n) = 58\times60 = 3480$

$e\times d \bmod \phi(n) = 1$
$31\times d \bmod 3480 = 1$
$3480 = 2^3\times3\times5\times29$, thus $\phi(3480) = 2^2\times2\times4\times28 = 896$
Using Euler's theorem we get
$d = 31^{895} \bmod 3480 = 31\times31^{894} \bmod 3480 = 31\times (31\times31)^{447} \bmod 3480=$
$= 31\times961\times (961\times961)^{223} \bmod 3480 = 31\times961\times1321\times (1321\times1321)^{111} \bmod 3480$
$= 31\times961\times1321\times1561\times (1561\times1561)^{55} \bmod 3480 =$
$= 31\times961\times1321\times1561\times721\times (721\times721)^{27} \bmod 3480 =$
$= 31\times961\times1321\times1561\times721\times1321\times (1321\times1321)^{13} \bmod 3480 =$
$= 31\times961\times1321\times1561\times721\times1321\times1561\times (1561\times1561)^{6} \bmod 3480 =$
$= 31\times961\times1321\times1561\times721\times1321\times1561\times (721\times721)^{3} \bmod 3480 =$
$= 31\times961\times1321\times1561\times721\times1321\times1561\times1321\times1321^{2} \bmod 3480 =$
$= 31\times961\times1321\times1561\times721\times1321\times1561\times1321\times1561 \bmod 3480 =$
$= 31\times961\times1321\times1561\times721\times1321\times721\times1321 \bmod 3480 =$
$= 31\times961\times1321\times1561\times721\times1561\times721 \bmod 3480 =$
$= 31\times961\times1321\times1561\times1321\times1561 \bmod 3480 =$
$= 31\times961\times1321\times721\times1321 \bmod 3480 = 31\times961\times1561\times721 \bmod 3480 =$
$= 31\times961\times1561\times721 \bmod 3480 = 3031$

Checking:
$31\times3031 \bmod 3480 = 93961 \bmod 3480 = 1$

4. Prove that RSA public system works correctly even when gcd(M, n) ≠ 1.

Solution idea:
If gcd(M, n) ≠ 1, then M is either a multiple of p or a multiple of q. Prove $M^{k\Phi(n)+1}$ mod p = M mod p separately for gcd(M,p) = 1 and gcd(M, p) ≠ 1. Do the same for mod q, and from these two show that $M^{k\Phi(n)+1}$ mod n = M mod n for all n.

Recall that for the RSA algorithm, the public key is a pair of numbers (e, n) and the and the private key is a pair of numbers (d, n) where $n = p \cdot q$ for some distinct prime numbers p and q. A message M, is an integer between 0 and $n - 1$. We will start by taking the following as true (as shown in lectures):

$$\begin{aligned}\phi(n) &= (p-1)(q-1)\\ \gcd\big(d, \phi(n)\big) &= 1\\ (e \cdot d) \bmod \big(\phi(n)\big) &= 1\\ E(M) &= M^e \bmod n\\ D(M) &= M^d \bmod n\\ E\big(D(M)\big) &= M^{e \cdot d} \bmod n\\ D(E(M)) &= M^{e \cdot d} \bmod n\end{aligned}$$

From this point, to prove that the RSA system works correctly, we need to prove that $E\big(D(M)\big) = M$ and $D\big(E(M)\big) = M$. This is equivalent to showing $M^{e \cdot d} \bmod n = M$.

We will approach this by first showing

$$\begin{aligned}M^{e \cdot d} \bmod p &= M \bmod p\\ M^{e \cdot d} \bmod q &= M \bmod q\end{aligned}$$

And since $\gcd(p, q) = 1$, then by the Chinese Remainder Theorem we know

$$\begin{aligned}M^{e \cdot d} \bmod (p \cdot q) &= M \bmod (p.q)\\ M^{e \cdot d} \bmod (n) &= M \bmod (n)\\ &= M\end{aligned}$$

Since M is an integer between 0 and $n - 1$.

To prove $M^{e \cdot d} \bmod p = M \bmod p$, we need to work though two cases. Case 1 is when $\gcd(M, p) = 1$, and Case 2 is when Case 2 is when $\gcd(M, p) \neq 1$.

For Case 1: If $\gcd(M, p) = 1$, then $M^{\phi(p)} \bmod p = 1$ by Euler's generalisation of Fermat's little theorem. Observe that $(e \cdot d) \bmod \big(\phi(n)\big) = 1 \Rightarrow (e \cdot d) = k\phi(n) + 1$. Then

$$\begin{aligned}M^{e \cdot d} \bmod p &= M^{k\phi(n)+1} \bmod p\\ &= M^{k\left((p-1)(q-1)\right)+1} \bmod p\\ &= \left(M \cdot M^{k(p-1)(q-1)}\right) \bmod p\\ &= \left(M \cdot \left(M^{(p-1)}\right)^{k(q-1)}\right) \bmod p\end{aligned}$$

$$
\begin{aligned}
&= \left(M \cdot \left(M^{(p-1)} \bmod p\right)^{k(q-1)}\right) \bmod p \\
&= \left(M \cdot \left(M^{\phi(p)} \bmod p\right)^{k(q-1)}\right) \bmod p \\
&= \left(M \cdot (1)^{k(q-1)}\right) \bmod p \\
&= (M) \bmod p
\end{aligned}
$$

Which is what we wanted to show.

For Case 2: If $\gcd(M, p) \neq 1$, then $M = (k \cdot p)$ (i.e. M is a multiple of p, since p is prime). Thus, we know that $M \bmod p = 0$, and

$$
\begin{aligned}
M^{e \cdot d} \bmod p &= (M \bmod p)^{e \cdot d} \bmod p \\
&= (0)^{e \cdot d} \bmod p \\
&= 0 \\
&= M \bmod p \text{ (because } M \bmod p{=}0)
\end{aligned}
$$

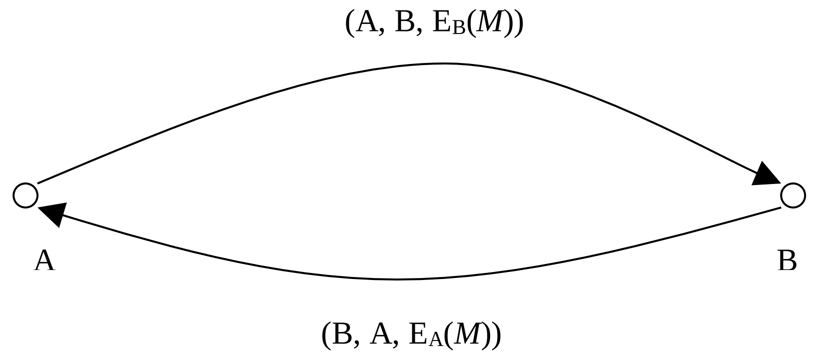
Which is what we wanted to show.

This shows that $M^{e \cdot d} \bmod p = M \bmod p$. To show $M^{e \cdot d} \bmod q = M \bmod q$, we apply the same argument, but replace p with q. We can then apply the Chinese Remainder Theorem as described above to show $M^{e \cdot d} \bmod n = M$, and thus $E\big(D(M)\big) = M$ and $D\big(E(M)\big) = M$, proving that RSA works correctly.

5. Show how an active wiretapper could break the following scheme to determine M. Users Alice and Bob exchange a message M using the following public-system protocol:

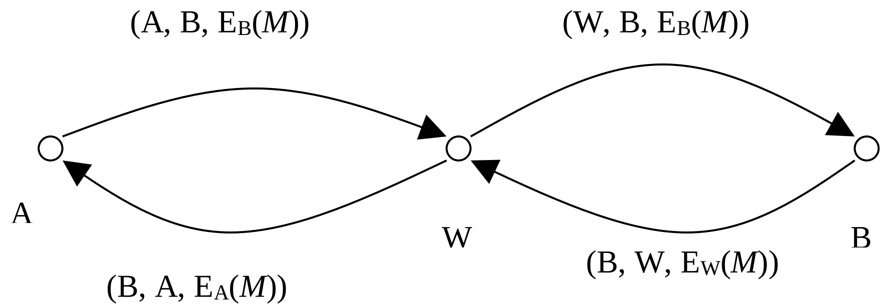
a. Alice encrypts M using Bob's public key and sends the encrypted message $E_B(M)$ together plaintext stating both Alice's and Bob's identity, i.e., $(A, B, E_B(M))$

b. Bob deciphers the ciphertext and replies to Alice with $(B, A, E_A(M))$.



Solution:

An active wiretapper Will can intercept the message $(A, B, E_B(M))$ and replace it with $(W, B, E_B(M))$; Bob will reply with $(B, W, E_W(M))$, and Will can find M by decrypting $E_W(M)$.



6. Suppose users Alice and Bob exchange a message *M* in a conventional system using a trusted third party S and the protocol given below. Show how an active wiretapper could break the scheme to determine *M* by replaying $E_A(R)$.

a. Alice generates a random number R and sends to S her identity A, destination B and $E_A(R)$.
b. S responds by sending $E_B(R)$ to Alice.
c. Alice sends $(E_R(M), E_B(R))$ to Bob.
d. Bob decrypts $E_B(R)$ and uses R to decrypt $E_R(M)$ and get M.

Solution:

An active wiretapper Will can pretend to be A, and send [A, W and $E_A(R)$] to S - S will respond with $E_W(R)$. Will can then decrypt $E_W(R)$ and use R to decrypt $E_R(M)$ and get M.