Exam Security

20 January 2015, 9:30–11:30

You can score a maximum of 100 points. Each question indicates how many points it is worth. You are NOT allowed to books or notes, or a (smart) phone. You may answer in Dutch or in English. Please write clearly, and don't forget to put your name and student number on each page that you hand in. You can keep this exam yourself.

1. (**12 points**) (understanding public keys and hashes)
   The creator of PGP is Phil Zimmerman. Of course, he has his own RSA key pair for use in PGP. Online you can find:

Phil's key fingerprint is:	9E 94 45 13 39 83 5F 70 7B E7 D8 ED C4 BE 5A A6
Phil's Key Id is:	C7A966DD

   The fingerprint is the hash $h(n, e)$ of the public key (n, e). The example contains 32 hexadecimals; it is thus a 128 bit hash value. The Key Id consists of 8 hexadecimals, corresponding to 32 bits. It is a part of the public key itself.

   (a) (**6 points**) Some people put their Key ID on their personal webpage, or on their business cards. Other, more security aware people, insist on putting their key fingerprint instead. Why is this more secure? Explain briefly what could go wrong if you publish only your Key ID.

   Begin Secret Info: .....

   The Key ID is much shorter than the fingerprint, so there is a much bigger chance that different people (with different public keys), share the same Key ID.
   In that case, if you look up the public key using the Key ID you may find the key for the wrong person, and thus encrypt a secret message for the wrong person and send it to him/her. The secret message can then be read by the wrong person.

Grading (total 6):	
aspect:	*points*
increased chance of collision	*3*
encrypt to the wrong person	*3*

   End Secret Info .....

   (b) (**6 points**) On an implementation level an encrypted (PGP) message $\{m\}_{(n,e)}$ is of the form

   $$keyid \mid \cdots$$

   where $\mid$ is concatenation. Thus it has the Key ID as a prefix. Is it more secure in this case to use the key fingerprint instead of the Key ID as prefix? Answer with Yes/No, and explain briefly.

   Begin Secret Info: .....

   No! Let A and B have different keypairs, but with the same key id. If the message is encrypted with A's public key, but ends up with B, then B can still not decrypt it, because he has the wrong key.

Grading (total 6):	
aspect:	*points*
"No"	*2*
correct explanation	*4*

End Secret Info .....

2. (**21 points**) (protocols, authentication, secret key, hash)
Consider smart cards using symmetric encryption. Each card C has its own "diversified key". We write id_C for some unique identifier of card C, known by the card itself. Card readers do not have the keys of all cards. (This takes too much memory, and is too risky and inflexible). Instead, card readers should be able to somehow compute the (symmetric) key K_C of card C from the card's identity id_C. Let's write this computation as $K_C = f(\mathrm{id}_C)$ for some function f. What f precisely is, does not matter for the moment.

   (a) (**7 points**) Write a short protocol with which the card can authenticate itself towards the reader, using this function f. Write this protocol in standard protocol notation, starting with the message $C \to R : id_C$ where C = card and R = reader.

Begin Secret Info: .....

$$
\begin{aligned}
C &\longrightarrow R : id_C \\
&\qquad\quad R \textit{ computes } K_C = f(\mathrm{id}_K) \\
R &\longrightarrow C : N \\
C &\longrightarrow R : K_C\{N\}
\end{aligned}
$$

Or:

$$
\begin{aligned}
C &\longrightarrow R : id_C \\
&\qquad\quad R \textit{ computes } K_C = f(\mathrm{id}_K) \\
R &\longrightarrow C : K_C\{N\} \\
C &\longrightarrow R : N
\end{aligned}
$$

Grading (total 7):	
aspect:	*points*
idea that reader used K_C for challenge-response authentication	*2*
correct protocol	*5*

End Secret Info .....

Below we list three possible versions of the function f, where h is some hash function, and M is a secret master key that is known to the "scheme manager" that operates these cards. Each card reader knows M.

(i) $K_C = M\{id_C\}$
(ii) $K_C = h(id_C)$
(iii) $K_C = h(id_C \mid M)$.

(b) (**7 points**) One of these choices of f is insecure, in the sense that card-authentication is not achieved. Which choice? Briefly explain an exploit.

Begin Secret Info: .....

The second one: $K_C = h(id_C)$. Since the first message is $C \to R : id_C$, an attacker can learn the card identity id_C by eavesdropping one protocol run. Then he can compute $K_C = h(id_C)$ himself and impersonate this card C.

Grading (total 7):	
aspect:	*points*
right outcome (ii)	*3*
proper explanation	*4*

End Secret Info .....

(c) (**7 points**) Concentrate on the above third option $K_C = h(id_C \mid M)$. Adapt, if needed, your protocol in (a), without changing the number of lines/messages, in such a way that the card knows that the reader knows M. Does this authenticate the particular reader that the card is communicating with? Answer Yes/No, and briefly explain your answer.

Begin Secret Info: .....

$$
\begin{aligned}
C &\longrightarrow R : id_C \\
&\qquad\qquad R \text{ computes } K_C = h(id_C \mid M) \\
R &\longrightarrow C : K_C\{N \mid \mathrm{id}_K\} \\
C &\longrightarrow R : N
\end{aligned}
$$

The identity id_K, or some other fixed name, should be included in the second challenge message from the reader, so that the card can see that this is a sensible message, encrypted with K_C. This shows to the card that the terminal knows M in order to compute K_C.
*This does **not** authenticate the particular reader (individually), but only as member of the group of readers.*

Grading (total 7):	
aspect:	*points*
proper adaption of the protocol	*4*
right answer "No"	*1*
right explanation	*2*

End Secret Info .....

3. (**24 points**) (modes of operation, CTR)
The counter (or CTR for short) mode of operation is increasingly popular. It can be used with many encryption functions; here we make use of AES.

Alice and Bob share a 128-bit AES key K. When Alice wants to encrypt a message to Bob, she generates a fresh 80-bit-long nonce N, and sets the counter CTR to 0 (represented with appropriate bit length). She also divides the message in blocks of length 128 bits:

$m_0, m_1, \ldots, m_k$. Finally, the ciphertext is computed block by block as described below and sent to Bob along with the plaintext nonce N. As usual, concatenation is denoted by | and bitwise XOR by $\oplus$.

$$c_0 = K\{N \mid CTR\} \oplus m_0,\ c_1 = K\{N \mid CTR+1\} \oplus m_1, \ldots,\ c_k = K\{N \mid CTR+k\} \oplus m_k.$$

In this notation the nonce N concatenated with the values $CTR + i$ fill one block. The resulting ciphertext is:

$$K\{m\} = (N \mid c_0 \mid c_1 \mid \ldots \mid c_k).$$

(a) (**6 points**) Using one nonce N, what is the maximum number of message blocks m_i for Alice's message that she can securely encrypt? Explain your answer briefly.

Begin Secret Info: .....

Since the block length is 128 and the length of the nonce is 80 bits, there are 2^{48} different CTR. Thus, there are 2^{48} different pad blocks that can be securely xored with the message blocks.
If you use more than 2^{48} blocks, different message blocks m_i will be xored with the same key-block, which is a classical mistake: the key can then be removed by xoring the two ciphertext blocks.

Grading (total 6):	
aspect:	*points*
outcome 2^{48}	*4*
same key-block explanation	*2*

End Secret Info .....

(b) (**6 points**) Describe how Bob can decrypt the received ciphertext $K\{m\}$. Which property of $\oplus$ is used?

Begin Secret Info: .....

$$m_0 = K\{n|CTR\} \oplus c_0,\ m_1 = K\{n|CTR+1\} \oplus c_1, \ldots,\ m_k = K\{n|CTR+k\} \oplus c_k.$$

Here one uses $k \oplus (k \oplus m) = m$.

Grading (total 6):	
aspect:	*points*
correct formula	*4*
correct property	*2*

End Secret Info .....

(c) (**6 points**) Write down at least two favorable properties of the counter mode.
Begin Secret Info: .....
The counter mode can easily be parallelised, which makes it very efficient on multi-core processor implementations.
There is basically no error propagation: If a bit of the ciphertext is flipped during transmission, only that block of the message is affected.
The encryption and decryption are identical, which is also beneficial from an implementation point of view; in particular, on embedded (lightweight) devices.

Grading (total 6):	
aspect:	*points*
per property	*3*

End Secret Info .....

(d) (**6 points**) Assume that the bit length of the message is not a multiple of 128, that is, the last message block m_k is less than 128 bits. How can Alice still encrypt this message to make sure that Bob receives it correctly? Explain briefly why your construction is secure.

Begin Secret Info: .....
The counter mode does not require any special padding (this can be added as an extra advantage for the previous problem). Alice can simply truncate the last block $K\{n|CTR+k\}$ to the same length as m_k.
Since any number of bits of $K\{n|CTR+k\}$ can be considered random, it does not reveal any additional information either about the pad or about the message.

Grading (total 6):	
aspect:	*points*
any padding answer	*2*
correct explanation	*4*

End Secret Info .....

4. (**28 points**) (modular computation, RSA public-key encryption and randomised signature) Consider the RSA cryptosystem. Assume that Alice's public key is $(n_A = 55, e_A = 3)$ and Bob's public-private key pair is $(n_B = 55, e_B = 17, d_B = 33)$.

(a) (**5 points**) Assume that you want to send an encrypted message $m_1 = 3$ to Bob. What is the ciphertext?

Begin Secret Info: .....
$\{m_1\}_{(n_B,e_B)} = 3^{17} \mod 55 \equiv 53$, *computed as follows.*

- $3^{17} \equiv (3^4)^4 \cdot 3 \equiv 81^4 \cdot 3 \equiv (26^2)^2 \cdot 3 \equiv \ldots$
- *intermezzo:* $26^2 = 676 = 12 \cdot 55 + 16 \equiv 16$
- $\ldots \equiv 16^2 \cdot 3 \equiv 256 \cdot 3 \equiv \ldots$
- *intermezzo:* $256 = 4 \cdot 55 + 36 \equiv 36$
- $\ldots \equiv 36 \cdot 3 \equiv 108 \equiv -2 \equiv 53$

Grading (total 5):	
aspect:	*points*
right formula	*2*
right outcome	*1*
computation	*2*

End Secret Info .....

(b) (**5 points**) Verify that $d_B = 33$ is indeed the private key of Bob.

Begin Secret Info: .....
Since $55 = 5 \cdot 11$, $\varphi(55) = (5-1)(11-1) = 40$. *One has to confirm that* $e_B \cdot d_B \equiv 1 \mod \varphi(n)$. *This is indeed the case as* $17 \cdot 33 = 561 = 14 \cdot 40 + 1 \equiv 1 \mod 40$.

Grading (total 5):	
aspect:	*points*
right factorisation $55 = 5 \cdot 11$	*1*
right φ	*1*
right condition $e \cdot d \equiv 1 \mod \varphi$	*2*
right computation	*1*

End Secret Info .....

(c) (**5 points**) What is Alice's secret key? Make all computations explicit.

Begin Secret Info: .....

From the previous problem: $\varphi(55) = 40$. *We have to compute* $d_A = (e_A)^{-1} \mod \varphi(n) = 3^{-1} \mod 40$. *By the extended Euclidean algorithm one can compute that:*

$$-2 \cdot 40 + 27 \cdot 3 = -80 + 81 = 1.$$

Hence the inverse of 3 is 27. *Thus, her private key* $(n_A = 55, d_A = 27)$. *Indeed,* $3 \cdot 27 = 81 = 2 \cdot 40 + 1 \equiv 1 \pmod{40}$.

Grading (total 5):	
aspect:	*points*
right formula $d_A = (e_A)^{-1} \mod \varphi(n)$	*2*
right egcd equation $-2 \cdot 40 + 27 \cdot 3 = 1$	*2*
right final outcome $d_A = 27$, *or* $(n_A = 55, d_A = 27)$	*1*

End Secret Info .....

(d) (**8 points**) We will use the blind RSA signature. Have Bob sign the message $m_2 = 13$ for you using $r = 3$ as the random value.

Begin Secret Info: .....
There are three steps: blinding of m_2, *signing of this blinding, and undoing the blinding on the signed message.*

i. *We can use the outcome* $3^{17} \equiv 53 \mod 55$ *from (a) to compute the blinded version* m_2' *of* m_2, *as:*

$$\begin{aligned} m_2' &= (r^e) \cdot m_2 \mod n \\ &= 3^{17} \cdot 13 \mod 55 \\ &\equiv 53 \cdot 13 \\ &= 689 \\ &= 660 + 29 \\ &= 12 \cdot 55 + 29 \\ &\equiv 29. \end{aligned}$$

ii. Next Bob signs this blinded message as:

$$\begin{aligned} k &= \{m_2'\}_{(n_B,d_B)} \\ &= (m_2')^{d_B} \\ &= 29^{33} \mod 55 \\ &= 24. \end{aligned}$$

For this last step we can use some earlier results:

- $29^{33} \equiv 29 \cdot (29^2)^{16} \equiv \ldots$
- *intermezzo:* $29^2 = 841 = 15 \cdot 55 + 16$
- $\ldots \equiv 29 \cdot (16^2)^8 \equiv 29 \cdot 36^8 \equiv 29 \cdot (36^2)^8 \equiv 29 \cdot 1296^8 \equiv \ldots$
- *intermezzo:* $1296 = 23 \cdot 55 + 31$
- $\ldots \equiv 29 \cdot (31^2)^4 \equiv 29 \cdot 961^2 \equiv \ldots$
- *intermezzo:* $961 = 17 \cdot 55 + 26$
- $\ldots \equiv 29 \cdot 26^2 \equiv 29 \cdot 16 \equiv 464 \equiv 8 \cdot 55 + 24 \equiv 24.$

iii. The final blind signature is:

$$\{m_2\}_{(n_B,d_B)} = \frac{k}{r} = \frac{24}{3} = 8.$$

(Properly, we can also get this by computing first the inverse of 3. This can be computed by the extended Euclidean algorithm, or making the following observation. Since $111 \equiv 1 \mod 55$ *and 111 is divisible by 3,* $111/3 = 37$ *is the inverse of 3 modulo 55.* $\{m_2\}_{(n_B,d_B)} = \frac{k}{r} = k \cdot r^{-1} = 24 \cdot 37 \mod 55 = 8$*.)*

Grading (total 8):	
aspect:	*points*
step 1, right formula	*1*
step 1, right computation	*1*
step 1, right outcome	*1*
step 2, right formula	*1*
step 2, right computation	*1*
step 2, right outcome	*1*
step 3, right formula	*1*
step 3, right outcome	*1*

End Secret Info .....

(e) (**5 points**) Verify that the signature in the previous point is valid.

Begin Secret Info: .....

$8^{17} \stackrel{?}{\equiv} 13$. *This is indeed true:* $8^{17} = 8 \cdot 8^{16} = 8 \cdot (((8^2)^2)^2)^2 \equiv 8 \cdot (26^2)^2 \equiv 8 \cdot 36 \equiv 13 \mod 55$

With intermediate steps, using some earlier results:

- $8^{17} \equiv 8 \cdot (8^2)^8 \equiv 8 \cdot 64^8 \equiv 8 \cdot 9^8 \equiv 8 \cdot (9^2)^4 \equiv 8 \cdot 81^4 \equiv 8 \cdot 26^4 \equiv 8 \cdot 36 \equiv 288 \equiv 5 \cdot 55 + 13 \equiv 13.$

Grading (total 5):	
aspect:	*points*
right formula to check	*2*
right computation	*2*
right outcome	*1*

End Secret Info .....

5. (**15 points**) (computing modular inverses, understanding RSA, computing factors)
In the previous exercise Alice and Bob use the same modulus, namely $n = 55$. In general, this is bad. In practice users don't have access to the factorisation $n = p \cdot q$ of their own modulus, but they do know $\varphi(n)$. We will use this below.

   Assume You and Bob both use $n = 1591$ as modulus, and thus both know $\varphi(n) = 1512$.

   (a) (**5 points**) Let the public key of Bob be $(1591, 5)$. Now you can compute Bob's private key. Do it!

   Begin Secret Info: .....
   Compute $d = e^{-1} \mod \varphi(n) = 5^{-1} \mod 1512$.
   Use extended gcd to find x, y *with* $5 \cdot x + 1512 \cdot y = 1$. *This can be done easily by hand. Take* $y = -2$, *so that* $1512 \cdot (-2) = -3024$. *Take* $d = \frac{3025}{5} = 605$.

Grading (total 5):	
aspect:	*points*
right $d = e^{-1} \mod \varphi(n)$	*2*
right egcd equation $5 \cdot 605 + 1512 \cdot (-2) = 1$	*2*
right outcome $d = 605$	*1*

   End Secret Info .....

   (b) (**10 points**) You can also factorise n. Make this explicit, step-by-step, and compute the factors.

   Begin Secret Info: .....
   We have $pq = 1591$ *and* $1512 = \varphi(n) = (p-1)(q-1) = pq - p - q + 1 = 1592 - p - q$. *Thus* $p + q = 1592 - 1512 = 80$. *There are two possible substutitutions.*

   - *Substituting* $p = 80 - q$ *in* $pq = 1591$ *yields* $80q - q^2 = 1591$, *so* $q^2 - 80q + 1591 = 0$. *This second order equation* $aq^2 + bq + c = 0$ *has determinant:*

$$b^2 - 4ac = 80^2 - 4 \cdot 1591 = 6400 - 6364 = 36 = 6^2$$

     Hence the solutions are:

$$\frac{80-6}{2} = \frac{74}{2} = 37 \qquad \frac{80+6}{2} = \frac{86}{2} = 43$$

     Indeed, $37 \cdot 43 = 1591$ *is the factorisation of* n.
   - *Substituting* $q = 80 - p$ *in* $pq = 1591$ *yields* $80p - p^2 = 1591$, *so* $p^2 - 80p + 1591 = 0$. *This yields the same equation, but with a different variable.*

Grading (total 10):	
aspect:	*points*
right equations for $p+q$ and pq	*3*
right quadratic equation	*3*
right solution method	*2*
right solutions	*2*

End Secret Info .....

Begin Secret Info: .....
Finally we have a ***toetsmatrix****, relating the original goals of the course to the exercises:*

goal	***exercise***
recognise security goals	*1,2*
hash functions	*1,2*
public key crypto	*1,4,5*
secret key crypto	*2,3*
protocols	*2*

End Secret Info .....