Cryptography and Architectures for Computer Security

Exam Code: 095947 (old 090959), A.Y. 2017–2018, Semester: 2

Prof. G. Pelosi

September 13th, 2018 – Exam Session

Name:.................................... Surname:..

Student ID:............................... Signature: ....

Time: 2h:30'. Use of textbooks, notes, or Internet connected devices (including smartphones) is not allowed. The usage of simple calculators is allowed. Prior to turn in your paper, write your name on any additional sheet and sign it.

Question1 [3 pts]
Consider the TLS protocol answer the following questions:

(a) Can an *active* attacker break the confidentiality property of a TLS connection if the ciphersuites accepted by both the client and the server include insecure choices? How?

(b) Consider the TLS ciphersuites `TLS_DHE_ECDSA_WITH_AES_256_SHA256`, using an ECDSA keypair relying on the P-521 curve and `TLS_DHE_ECDSA_WITH_AES_192_SHA384` using an ECDSA keypair relying on the P-384 curve. Which one provides the most homogeneous security margin across all the primitives?

Solution:

(a) An active attacker may tamper with the TLS connection establishment forging the **ServerHello** message, which is sent in cleartext, and contains the ciphersuite choice. Tampering with it allows him to downgrade the connection to a `TLS_*_WITH_NULL_*` ciphersuite eliminating the symmetric encryption altogether, or moving to an *export-grade* ciphersuite which is breakable computationally.

(b) The second one. Indeed, all the involved parameter choices require around 2^{192} operations to be broken, while obtaining a collision on SHA-2-256 requires "only" 2^{128} operations, versus the 2^{256} required to compromise AES-256 or ECDSA on P-521.

Question 2 [5 pts]
Consider a password hashing scheme composed as follows: given a 7 printable ASCII (also known as 7-bit ASCII) character password and a single byte salt, the concatenation of the password and the salt is hashed 1024 times with SHA-2-512. The result is stored, together with the salt, as the password hash.

(a) Considering a CPU able to compute 2^{23} SHA-2-512 hashes per second, and the availability of 128GiB of RAM, compute the chain length of a rainbow table aimed at breaking the aforementioned password hashing scheme minimizing the computation time for a worst case lookup. Calculate the time required to compute the rainbow table, not taking into account the memory access time. Do not take into account the space required for the index or other data structure required to access the table in $O(1)$.

(b) Patch the password hashing scheme adding enough salt so that the aforementioned attack requires at least 2^{128} SHA-2-512 computations.

(c) Is it possible to find whether a user's password is "password" followed by his date of birth when employing your patched scheme?

Solution:

(a) Considering that a chain in the table is stored only as its beginning and end, that we have 2^{37}B available, and that a SHA-2-512 digest is 64B long, we have that we can store at most $\frac{2^{37}}{64\times 2} = 2^{30}$ chains. The password space to be swept is $2^{(7\times 7)+8} = 2^{57}$. We thus have that a chain will be $\frac{2^{57}}{2^{30}} = 2^{27}$ hashes long. Computing the entire table will require $2^{57} \times 2^{10}$ SHA-2-512 computations: at a rate of 2^{23} SHA-2-512 per second this will take 2^{34} seconds or around 200k core hours.

(b) Since computing the rainbow table requires a computational effort in the same order of magnitude as an single exhaustive search, and the password space is 2^{49} elements wide, a 79 bit salt is enough to force the attacker to compute the required number of SHA-2-512.

(c) Yes. Considering the computing power of the single CPU above (2^{13} password hashes per second) and the fact that reasonably valid birth dates are 365×150 we have that it is possible to test exhaustively all the said passwords in $\frac{365\times 150}{2^{13}} \approx \frac{2^{16}}{2^{13}} = 8$ seconds.

Question 3 [3 pts]
Describe the design principles of a stream cipher, mentioning its most common structures.

Solution:
see lectures...

Question 4 [5 pts]
Consider the finite field $\mathbb{F}_{2^4}$.

(a) Compute the number of irreducible and primitive polynomials that can be employed to represent the elements of the field.

(b) Exhibit the value of all primitive polynomials, justifying your answer.

Solution:

(a) $2^4 = 4 \cdot N_4(2) + 2 \cdot N_2(2) + 1 \cdot N_1(2)$
$N_1(2) = 2;$
$N_2(2) = \frac{2^2-2}{2} = 1;$
$N_4(2) = 16 - 2 \cdot 1 - 1 \cdot 24 = 3$

$\varphi(2^4 - 1) = 2 \cdot 4 = 8;$
$M_4(2) = \frac{8}{4} = 2$

(b) The irreducible polynomials that can be employed to represent the elements of the field are: $f_0(x) = x^4 + x^3 + x^2 + x + 1$, $f_1(x) = x^4 + x^3 + 1$, $f_2(x) = x^4 + x^2 + 1$, $f_3(x) = x^4 + x + 1$.

If α is a generator of $\mathbb{F}^*_{2^4}$, and $f(x) \in \mathbb{F}_2[x]$, with $\deg(f(x)) = 4$, is irreducible and and $f(\alpha) = 0$ then $f(x)$ is primitive. Given $n = |\mathbb{F}^*_{2^4}| = 15 = 3 \cdot 5$, if $f(x)$ is primitive it must hold that $\alpha^3 \not\equiv 1$ (trivially true as $deg(f(x)) = 4$), $\alpha^5 \not\equiv 1, \alpha^{15} \equiv 1$.

Considering $f(x) = f_0(x) = x^4 + x^3 + x^2 + x + 1$,
$\alpha^5 = \alpha \cdot (x^3 + x^2 + x + 1) = \ldots = \alpha$, therefore we can conclude that $f_0(x)$ is NOT primitive (it could be irreducible but the question of the exercise does not ask us to be more precise)

Considering $f(x) = f_1(x) = x^4 + x^3 + 1$,
$\alpha^5 = \alpha \cdot \alpha^4 \equiv \alpha^3 + \alpha + 1 \not\equiv 1$ ok!;
$\alpha^{15} = (\alpha^4)^3 \cdot \alpha^3 = (\alpha^3 + 1)^3 \cdot \alpha^3 = (\alpha^9 + \alpha^6 + \alpha^3 + 1) \cdot \alpha^3 = \alpha^{12} + \alpha^9 + \alpha^6 + \alpha^3 = (\alpha^3 + 1)^3 + \alpha^9 + \alpha^6 + \alpha^3 = \alpha^9 + \alpha^6 + \alpha^3 + 1 + \alpha^9 + \alpha^6 + \alpha^3 = 1$ ok!

Considering $f(x) = f_2(x) = x^4 + x^2 + 1$,
$\alpha^5 = \alpha \cdot \alpha^4 \equiv \alpha^3 + \alpha \not\equiv 1$ ok!;
$\alpha^{15} = (\alpha^4)^3 \cdot \alpha^3 = (\alpha^2 + 1)^3 \cdot \alpha^3 = (\alpha^6 + \alpha^4 + \alpha^2 + 1) \cdot \alpha^3 = (\alpha^4 + \alpha^2 + \alpha^4 + \alpha^2 + 1) \cdot \alpha^3 = \alpha^3$ **NO**! This means that $f_2(x) = x^4 + x^2 + 1$ is NOT primitive!!!

Therefore, primitive polynomials for the field $\mathbb{F}_{2^4}$ are:
$f_1(x) = x^4 + x^3 + 1$, and $f_3(x) = x^4 + x + 1$.

We can note that as $f_2(x) = x^4 + x^2 + 1 = (x^2 + x + 1)^2$, it is clearly NOT irreducible too, therefore the irreducible polynomials are:
$f_0(x) = x^4 + x^3 + x^2 + x + 1$, $f_1(x) = x^4 + x^3 + 1$, and $f_3(x) = x^4 + x + 1$.

Question 5 [5 pts]

(a) Consider an instance of the Diffie-Hellmann protocol over the multiplicative group $G = (\mathbb{F}^*_{11^2}, \cdot)$. Considering the primitive polynomial $f(x) = x^2 + 1 \in \mathbb{F}_{11}[x]$, the elements of the field are represented as first degree polynomials of the form $\theta_1\alpha + \theta_0$ with $\theta_0, \theta_1 \in \mathbb{F}_{11}$, and $\alpha \in \mathbb{F}_{11^2} \backslash \mathbb{F}_{11}$ such that $\alpha^2 \equiv -1$.

State if the following discrete logarithms exists and, in the positive case, compute their values.

$$x_1 = \log^{\mathsf{D}}_{\alpha}(-1),\ x_2 = \log^{\mathsf{D}}_{1/\alpha}(-1)$$

(b) Many implementations of the Diffie-Hellman protocol assume to work in a subgroup G of the multiplicative group $(\mathbb{Z}_p, \cdot)$; in order to simply the implementation is it a good idea to employ a subgroup of the $(\mathbb{Z}_p, +)$, with the same prime p?. How is security affected?

Solution:

(a) The discrete logarithm problems can be rewritten as the problems of finding the exponents $x_1, -x_2 \in \{0, 1, 2, \ldots, |G| - 1\}$, $|G| = 120$ such that $\alpha^{x_1} \equiv -1$, and $\alpha^{-x_2} \equiv -1$

Both logarithms can be rewritten keeping the generator $\alpha \in G$ as radix. Thus, both logarithms exist!

It is quite immediate to look at the primitive polynomial employed to represent the elements of the multiplicative group of the polynomial field $\mathbb{F}_{11^2}$ and conclude that:
$x_1 = 2 \bmod 120$
$-x_2 = 2 \bmod 120 \Leftrightarrow x_2 = -2 \bmod 120 = 118 \bmod 120$.

(b) see lectures ...

Question 6 [14 pts]

(a) Apply the Pollard's ρ method to factorize the RSA modulus $n = p \cdot q = 551_{\text{dec}}$.
Assume $f(x) = x^2 + 1 \bmod n$ as the "random-walking" function. Show every step of the computation. (As a backup alternative, apply a "trivial division" strategy).

(b) Describe the Pollard's $P - 1$ factoring algorithms, pointing out its computational complexity.

(c) Given the values $e_1 = 9$, $e_2 = 11$ state which of them is an admissible public exponent, motivating your answer. Show the value of the corresponding RSA public and private keys $k_{\text{pub}}{=}(e, n)$, $k_{\text{priv}}{=}(p, q, \varphi(n), d)$.

(d) Sign the message $m{=}256_{\texttt{dec}}{\in}\mathbb{Z}_n$ (without employing any padding scheme) through applying the CRT. Describe each step of the procedure.

(e) Consider a schoolbook RSA cryptosystem employing the Montgomery multiplication as building block. Write down the pseudo code of both the encryption and the CRT-based decryption transformations, pointing out their asymptotic computational complexities.

(f) Consider the modulus $N = 29_{\text{dec}}$ and the value $A = 6_{\text{dec}}$. Assuming a binary encoding of the operands and the smallest possible value for the Montgomery radix R

- compute the value $\widetilde{A}$ that is the result of mapping A into the Montgomery domain;
- apply the Montgomery multiplication algorithm to compute $\widetilde{A}^2$.

Show every step of the computations.

Solution:

(a) $p = 19$, $q = 29, n = p \cdot q = 551$.

(b) see lectures ...

(c) $\varphi(n) = 504 = 2^3 \cdot 3^2 \cdot 7$, $e_2 = 11_{\text{dec}} \in \mathbb{Z}^*_{\varphi(n)}$, $d \equiv_{\varphi(n)} {e_2}^{-1} \equiv_{504} 11^{-1} \equiv_{504} 11^{143} \equiv_{504}$ 275_{dec};

(d) $m_p \equiv_{19} 256^{275 \bmod 18} \equiv_{19} 9^5 \equiv_{19} 16$,
$m_q \equiv_{29} 256^{275 \bmod 28} \equiv_{29} -5^{23} \equiv_{29} -4 \equiv_{29} 25$,
$s = q \cdot (q^{-1} \bmod p) \cdot m_p \equiv_{551} 29 \cdot 2 \cdot 16 + 19 \cdot 26 \cdot 25 \equiv_{551} 928 + 12350 \equiv_{551} 13278 \equiv_{551}$ 54dec.

(e) see lectures ...

(f) $N = 29$, $R = 32$, $RR' - NN' = 1$,
$R' = R^{-1} \bmod N = 3^{27} \bmod 29 = 10$,
$N' = -N^{-1} \bmod R = -29^{\varphi(32)-1} \bmod 32 = -29^{15} \bmod 32 = -21 \bmod 32 = 11$.
$\widetilde{A} = \texttt{MMul}(A, R^2) = A \cdot R^2 \cdot R^{-1} \bmod N = 6 \cdot 32 \bmod 29 = 18$,
$\widetilde{A}^2 = \texttt{MMul}(\widetilde{A}, \widetilde{A}) = 18 \cdot 18 \cdot 10 \bmod 29 = 21$,
see lectures for the execution trace of the Montgomery multiplication with binary encoded operands.