Cryptography and Architectures for Computer Security

Exam Code: 095947 (old 090959), A.Y. 2018–2019, Semester: 2

Prof. G. Pelosi

January 20th, 2020

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

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

Time: 2h, 15min. 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.

Question 1 [5 pts]
Suppose you are in charge of security for a major web site, and you are considering what would happen if an attacker stole your database of `username`s and `password`s. You have already implemented a basic defense: instead of storing the plaintext passwords, you store their SHA-256 hashes. Your threat model assumes that the attacker can carry out 4 million SHA-256 hashes per second. His goal is to recover as many plaintext passwords as possible from the information in the stolen database. Valid passwords for your site may contain only characters `a`– `z`, `A`–`Z`, and `0`–`9`, and are exactly 8 characters long.

(a) Given the hash of a single password, how many hours would it take for the attacker to crack a single password by brute force, on average?

(b) Describe an attack based on the use of rainbow tables, specifying what they are and the computation-memory tradeoffs that improve on the brute force approach.

(c) Describe possible improvements to the basic approach of storing `username`s and `hash`es (where `hash` = SHA-256(`password`) for the users of the website at hand, with the aim to provide strong protection agaist attacks based on the use of rainbow tables.

Solution:

(a) There are $n = (26+26+10)^8 \approx 2^{48}$ possible passwords. Since we assume passwords are uniformly random selected, an unknown password has an equal probability of having any of the n values, and the expected number of guesses in a brute-force attack is $n/2$. Therefore, the attack will take:

$$\frac{n}{2} \cdot \frac{1}{4 \cdot 10^6} \approx \frac{2^{48}}{2} \cdot \frac{1}{2^{22}} = 2^{25} \text{ seconds} \approx 388 \text{ days} \approx 1 \text{ year}$$

(b) ... see lectures ... A RT is built as a set of chains. The stored head of a chain is a password, while the stored tail of the chain is a digest; ... the time-memory tradeoff provided by a RT depends on the number of chains and the length of each chain ... Given a digest **dgs**, the breaker checks if it is equal to any of the tails in the table, if one

the tails matches then ok! we can start from the head of that chain and interleave the computation of a hash and a reduction function until the target password is derived ...; if no digest stored as tails of the chains matches, then the breaker computes `HASH(dgs)` and checks it against all the tails as described before; if no-one matches he computes `HASH(HASH(dgs))` etc..

(c) ... compute the digest of any password employing the same `salt` (with proper length...), or better (at cost of a moderate additional storage) apply a different `salt` for each password.

Question 2 [3 pts]
A programmer wants to use CBC in order to protect both the integrity and confidentiality of network packets. She attaches a block of zero bits M_{n+1} to the end of the plaintext $M_1||M_2||\dots||M_n$ as redundancy, then encrypts with CBC. At the receiving end, she verifies that the added redundant bits are still all zero after CBC decryption. Does this test ensure the integrity of the transferred message? Justify your answer.

Solution:
The CBC decryption function derives the plaintext blocks m_i with $i \in \{1, 2, \dots, n\}$ as $m_i = \mathsf{Dec}_k(c_i) \oplus c_{i-1}$, $c_0 = \texttt{IV}$. If an adversary leaves untouched the last two blocks of the ciphertext message c_{n-1} and c_n, but tamper with the remaining blocks of the encrypted material, the modifications will not be detected via the proposed integrity mechanism. Therefore, the proposed mechanism does not provide integrity for the transferred message.

Question 3 [8 pts]
Consider the finite field $\mathbb{F}_{3^4}$.

(a) Establish if $f(x) = x^4 - x - 1 \in \mathbb{F}_3[x]$, and $g(x) = x^4 - x^2 - 1 \in \mathbb{F}_3[x]$ are irreducible polynomials or not. Establish also if $f(x)$ and $g(x)$ are primitive polynomials. Justify your answer.

(b) Exhibit the number fo generators in $\mathbb{F}_{3^4}$, and the number of both monic irreducible polynomials and monic primitive polynomials having degree 4 and coefficients in $\mathbb{F}_3[x]$.

(c) Explain what is the hard problem in the multiplicative group of a finite field suitable to set up a cryptosystem and describe at least one algorithm (except brute-forcing) to solve it, specifying its computational complexity.

Solution:

(a) $f(x) = x^4 - x - 1 \in \mathbb{F}_3[x]$ ... $\alpha^{81} \not\equiv \alpha \Rightarrow$ Reducible!
$g(x) = x^4 - x^2 - 1 \in \mathbb{F}_3[x]$ ... $\alpha^{81} \equiv \alpha \Rightarrow$ Irredcible! not primitive as $\alpha^8 = 1$

(b) $n = |\mathbb{F}_{3^4}^*| = 3^4 - 1 = 80 = 2^4 \cdot 5$. Num. of generators $= \varphi(n) = (16-8)\cdot(5-1) = 32$.
$4 \cdot N_4(3) + 2 \cdot N_2(3) + 1 \cdot N_1(3) = 3^4$
$N_1(3) = 3$
$N_2(3) = \frac{3^2-3}{2} = 3$
Num. of monic irr. poly. $= N_4(3) = \frac{81-6-3}{4} = 18$.
Num. of monic primitive poly. $= M_4(3) = \frac{\varphi(n)}{4} = 8$.

(c) see lectures ...

Question 4 [8 pts]
Consider an El-Gamal signature scheme based on the cyclic group $G=(\mathbb{F}_{2^6}^*, \cdot)$, with order $n=|G|=63$. Let $P(x)=x^6+x^5+x^4+x+1 \in \mathbb{F}_2[x]$ be the generating polynomial of the field $\mathbb{F}_{2^6} \cong \mathbb{F}_2(\alpha)$, $P(\alpha)=0$, $\alpha \in \mathbb{F}_{2^6} \setminus \mathbb{F}_2$, and assume to employ a hash function $(h : \{0,1\}^* \mapsto \mathbb{Z}_n)$ that maps its input binary sequences in their corresponding decimal values modulo n.
Given the key-pair:

$$k_{\text{priv}} = \langle\ s \in \mathbb{Z}_{63}\ \rangle = \langle\ 11\ \rangle, \quad k_{\text{pub}} = \langle\ n,\ \alpha,\ \alpha^s\ \rangle = \langle\ 63,\ \alpha,\ \alpha^{11} \equiv \alpha^3+1 = (001001)\ \rangle$$

(a) Check if the following signatures are correct:

(i) $S_1 = \langle\ m_1,\ \gamma,\ \delta\ \rangle = \langle\ (100000),\ (010000),\ 27\ \rangle$

(ii) $S_2 = \langle\ m_2,\ \gamma,\ \delta\ \rangle = \langle\ (111111),\ (001000),\ 0\ \rangle$

(b) In a practical implementation of the ElGamal signature scheme based on the arithmetic of a finite field, what are the criteria to select the cryptosystem parameters and establish its (mathematical) security level?

Solution:

(a) $Verify_{K_{\text{pub}}}(\langle m_1, S_1 \rangle) = \text{true} \Leftrightarrow (\alpha^s)^{h(\gamma)} \cdot \gamma^\delta = \alpha^{h(m_1)}$

(i) $\gamma \equiv \alpha^4$
$h(m_1) \equiv 32$
$h(\gamma) \equiv 16$
$(\alpha^s)^{h(\gamma)} = (\alpha^{11})^{16} \equiv \alpha^{176 \bmod 63} \equiv \alpha^{50}$
$(\alpha^s)^{h(\gamma)} \cdot \gamma^\delta = \alpha^{50} \cdot (\alpha^4)^{27} \equiv \alpha^{50} \cdot \alpha^{108 \bmod 63} \equiv \alpha^{95} \equiv \alpha^{32} \equiv \alpha^5+\alpha^4+\alpha^2+\alpha+1$
$\alpha^{h(m_1)} = \alpha^{32} \equiv \alpha^5+\alpha^4+\alpha^2+\alpha+1$
The signature S_1 is correct!

(ii) δ should be different from zero! Moreover,
$\gamma \equiv \alpha^3$
$h(m_2) \equiv 63 \equiv 0$
$h(\gamma) \equiv 8$
$(\alpha^s)^{h(\gamma)} = (\alpha^{11})^8 \equiv \alpha^{88 \bmod 63} \equiv \alpha^{25}$
$(\alpha^s)^{h(\gamma)} \cdot \gamma^\delta = \alpha^{25} \cdot (\alpha^3)^0 \equiv \alpha^{25} \equiv \ldots \equiv \alpha^5+\alpha^3+\alpha^2+\alpha+1$
$\alpha^{h(m_2)} = \alpha^0 \equiv 1$
The signature S_2 is **not** correct!

(b) see lectures ...

Question 5 [12 pts]

(a) Apply the Pollard's ρ method to factorize the RSA modulus $n = p \cdot q = 713$.
Assume $f(x) = x^2 + 1 \bmod n$ as "random-walking" function. Show every step of the computation. (As a backup alternative, apply a "trivial division" strategy).

(b) Describe the Miller-Rabin primality test, and show its application to the greatest factor of n, with base $a = 3$.

(c) Choose an admissible public exponent e between the values $e = 11_{\texttt{dec}}$ and $e = 13_{\texttt{dec}}$ and compute the value of the corresponding RSA private key $k_{priv} = (p, q, \varphi(n), d)$. Show every step of the computation.

(d) Sign the message $m{=}100_{\texttt{dec}} \in \mathbb{Z}_n$ (without employing any padding scheme) through applying the CRT. Describe each step of the procedure.

(e) Assume to work into the Montgomery domain: $(\widetilde{\mathbb{Z}}_p, +, \times),\ p = 13$

- Show the definition of the Montgomery Multiplication and the smallest admissible value for the Montgomery Radix: R
- Show the high-level pseudo-code to implement the Montgomery Reduction procedure `MRed(...)`, and prove the correctness of the algorithm.
- Compute a pair of integer values R', p' that satisfy the relation: $\gcd(R, p){=}R\,R'{-}p\,p'{=}1$.
- Compute the Montgomery multiplication $C = A \times B \bmod p$, where $A = 11_{\text{dec}}$ and $B = 4_{\text{dec}}$ are values in the Montgomery domain, assuming a binary encoding of the operands

Solution:

(a) *Apply the Pollard's ρ method to factorize the RSA modulus $n = p \cdot q = 713$. Assume $f(x) = x^2 + 1 \bmod n$ as "random-walking" function.*

Starting point $x_0 = 2$

$\mathbf{i}$	$\mathbf{x = x_i}$	$\mathbf{y = x_{2i}}$	$\lambda = \mathbf{gcd}(\lvert\mathbf{x - y}\rvert, \mathbf{n})$
0	2	2	1
1	5	26	1
2	26	584	31

$p = 31$, $q = n/p = 23$

(c) Choose an admissible public exponent e between the values $e = 11_{\texttt{dec}}$ and $e = 13_{\texttt{dec}}$ and compute the value of the corresponding RSA private key $k_{priv} = (p, q, \varphi(n), d)$. Show every step of the computation.

$\varphi(n) = 660 = 2^2 \cdot 3 \cdot 5 \cdot 11$, $e \in \mathbb{Z}^*_{\varphi(n)} \Rightarrow e = 13$
$d = e^{-1} \bmod \varphi(n) = e^{\varphi(\varphi(n))-1} \bmod \varphi(n)$,
$\varphi(\varphi(n)) = 160$
$d = 13^{-1} \bmod \varphi(n) = 13^{159} \bmod 660 = 13^{\langle 10011111 \rangle_2} \bmod 660 = \ldots = 457 \bmod 660$

(d) *Sign the message $m{=}100_{\texttt{dec}} \in \mathbb{Z}_n$ (without employing any padding scheme) through applying the CRT. Describe each step of the procedure.*

$$s \equiv_{713} 100^{457} = \ldots \texttt{CRT} \ldots = 59.$$

(e) ... ...
$C{=}\mathsf{MonPro}(11, 4) \stackrel{\text{def}}{=} 11 \cdot 4 \cdot R^{-1} \bmod p {\equiv_{13}} 44 \cdot 16^{-1} {\equiv_{13}} 5 \cdot 9 \equiv_{13} 6_{\text{dec}}$