Univ. Grenoble Alpes/Grenoble INP **2016–2017**
M2 Cybersecurity

Exercises for Crypto's lectures (Part 2)

Questions on Elliptic Curves

Notice : *the last exercises on the arithmetic of elliptic curves are optionnal.*

1 *ECKDSA : A certificat based version of ECDSA* : We consider the following signature scheme, called ECKDSA. Let H be a cryptographic hash functions whose outputs are bit strings of length ℓ_H. The bitlength of the domain parameter n should be at least ℓ_H. We consider an elliptic curve domain parameter

$$D = (q, FR, S, a, b, P, n, h).$$

Furthermore, we denote by $hcert$ the hash value of the signer's certification data that should include the signer's identifier, domain parameters and public key. The signer's private key is a random integer $d \in [1, n]$. We assume that d is invertible modulo n and we denote d^{-1} its inverse (mod n). The signer's public key is $Q = d^{-1}P$. The scheme is as follows :
ECKDSA signature generation
Input : the domain parameter D, private key d, hashed certification data $hcert$, message m.
Output : signature (r, s).

(a) Choose randomly $k \in [1, n-1]$.
(b) Compute $kP = (x_1, y_1)$.
(c) Compute $r = H(x_1)$.
(d) Compute $e = H(hcert||m)$.
(e) Compute $w = r \oplus e$ and convert w to an integer w'.
(f) If $w' \geq n$ then replace w' by $w' - n$.
(g) Compute $s = d(k - w') \mod n$. If $s = 0$ then go to step (a).
(h) Return (r, s).

ECKDSA signature verification
Input : the domain parameter D, public key Q, hashed certification data $hcert$, message m, signature r, s).
Output : Acceptance or rejection of the signature.

(a) Verify that the bitlength of r is at most ℓ_H and that s is an integer in the interval $[1, n-1]$. If any verification fails then return("Reject signature").
(b) Compute $e = H(hcert||m)$.
(c) Compute $w = r \oplus e$ and convert w to an integer w'.
(d) If $w' \geq n$ then replace w' by $w' - n$.
(e) Compute $X = sQ + w'P$.
(f) Compute $v = H(x_1)$ where x_1 is the x-coordinate of X.
(g) If $v = r$ then return("Accept the signature") else return("Reject signature").

Questions :

(a) Prove that signature verification works for ECKDSA.

(b) Compute the costs of ECDSA (as seen in the lectures) and ECKDSA in terms of; scalar multiplications (i.e., λZ with Z point on the curve and λ integer), simple addition on the curve, modular inversions, modular multiplications and modular additions. Is ECKDSA more expensive than ECDSA?

(c) What is the interest of using $d^{-1}P$ as public key instead of dP?

2 *The Station-to-Station (STS) protocol*: We consider the following key agreement scheme based on elliptic curves. The goal is for two entities A and B to establish a shared secret key. We assume that $D = (q, FR, S, a, b, P, n, h)$ is the elliptic curve domain parameter for A and B, KDF is a key derivation function, MAC is a message authentication code, and $SIGN$ is a signature generation algorithm (e.g., ECDSA or RSA). If any verification in the following protocol fails, then the protocol run is terminated with failure.

Station-to-station (STS) key agreement

(a) A selects randomly $k_A \in [1, n-1]$, computes $R_A = k_A P$ and sends A, R_A to B.

(b) B does the following :
— Check that R_A is a point on the curve, $R_A \neq \infty$ and $nR_A = \infty$.
— Select randomly $k_B \in [1, n-1]$ and compute $R_B = k_B P$.
— Compute $Z = hk_B R_A$ and verify that $Z \neq \infty$.
— Compute $(k_1, k_2) = KDF(x_Z)$ where x_Z is the x-coordinate of Z.
— Compute $s_B = SIGN_B(R_B, R_A, A)$ and $t_B = MAC_{k_1}(R_B, R_A, A)$.
— Send B, R_B, s_B, t_B to A.

(c) A does the following :
— Check that R_B is a point on the curve, $R_B \neq \infty$ and $nR_B = \infty$.
— Compute $Z = hk_A R_B$ and verify that $Z \neq \infty$.
— Compute $(k_1, k_2) = KDF(x_Z)$ where x_Z is the x-coordinate of Z.
— Verify that s_B is B's signature on the message (R_B, R_A, A).
— Compute $t = MAC_{k_1}(R_B, R_A, A)$ and verify that $t = t_B$.
— Compute $s_A = SIGN_A(R_A, R_B, B)$ and $t_A = MAC_{k_1}(R_A, R_B, B)$.
— Send s_A, t_A to B.

(d) B does the following :
— Verify that s_A is A's signature on the message (R_A, R_B, B).
— Compute $t = MAC_{k_1}(R_A, R_B, B)$ and verify that $t = t_A$.

(e) The session key is k_2.

Questions:

(a) Prove that the scheme works.

(b) What are the roles of s_A, s_B, t_A and t_B?

(c) Compute the costs of ECMQV (as seen in the lectures) and STS in terms of; scalar multiplications (i.e., λZ with Z point on the curve and λ integer), simple addition on the curve, modular inversions, modular multiplications and modular additions. Is STS more expensive than ECMQV?

3 *Fully Hashed MQV*: We propose the following key establishment protocol (derived from the MQV protocol) and called FHMQV.

Let G be a cyclic group (denoted multiplicatively with 1 as neutral element) of prime order q and let g be a generator of G. We assume that solving the DLP is hard for G. Let H' be a cryptographic hash function which output ℓ-bits integers with $\ell =$(size in bits of q)/2 and H be a general cryptographic hash function. Let $\hat{A}$ and $\hat{B}$ be two entities who want to establish a common session key. We will denote $A \in G$ (respectively B) the public key of $\hat{A}$ (resp. $\hat{B}$) and $a \in \{1, ..., q-1\}$ (resp. b) its private key. If F is a hash function, by $F(X_1, \dots, X_n)$ we mean $F(X_1 || \cdots || X_n)$. The protocol consists of the following steps :

(a) $\hat{A}$ choose randomly $x \in \{1, ..., q-1\}$, compute $X = g^x$ and send (A, B, X) to $\hat{B}$

(b) $\hat{B}$ does the following :

  (i) Check that $X \in G$ with $X \neq 1$.

  (ii) Choose randomly $y \in \{1, ..., q-1\}$, compute $Y = g^y$ and send (A, B, Y) to $\hat{A}$.

  (iii) Compute $d = H'(X, Y, A, B)$, $e = H'(Y, X, A, B)$, $s_B = (y + eb) \mod q$, $\sigma_B = (XA^d)^{s_B}$ and $K = H(\sigma_B, A, B, X, Y)$.

(c) $\hat{A}$ does the following :

  (i) Check that $Y \in G$ with $Y \neq 1$.

  (ii) Compute $d = H'(X, Y, A, B)$, $e = H'(Y, X, A, B)$, $s_A = (x + da) \mod q$, $\sigma_A = (YB^e)^{s_A}$ and $K = H(\sigma_A, A, B, X, Y)$.

The session key is K.
Questions :

(a) Show that the scheme works (i.e., $\hat{A}$ and $\hat{B}$ compute the same session key K).

(b) What is the role of s_A and s_B ?

(c) When G is an elliptic curve, compute the costs of FHMQV in terms of; scalar multiplications (i.e., λZ with Z point on the curve and λ integer), simple addition on the curve, modular inversions, modular multiplications and modular additions. Compare (and discuss) the cost of FHMQV with the one of ECDH.

4 *Revisiting ECDSA* : The Elliptic Curve Digital Signature Algorithm (ECDSA), as seen in the lectures, is described by the two following algorithms :

Algorithm 1: ECDSA signature generation

Input : Domain parameters $D = (q, FR, S, a, b, P, n, h)$, private key d, message m
Output: Signature (r, s)
1 Select $k \in_R [1; n-1]$;
2 Compute $kP = (x_1, y_1)$ and convert x_1 to an integer $\bar{x}_1$;
3 Compute $r = \bar{x}_1 \mod n$. If $r = 0$ go to step 1;
4 Compute $e = H(m)$;
5 Compute $s = k^{-1}(e + dr) \mod n$. If $s = 0$ then go to step 1;
6 Return (r, s);

Algorithm 2: ECDSA signature verification

Input : Domain parameters $D = (q, FR, S, a, b, P, n, h)$, public key $Q = dP$, message m, signature (r, s)

Output: Acceptance or rejection of the signature

1 Verify that r and s are integers in $[1; n-1]$. If any verification fails, then return "Reject the signature";
2 Compute $e = H(m)$;
3 Compute $w = s^{-1} \mod n$;
4 Compute $u_1 = ew \mod n$ and $u_2 = rw \mod n$;
5 Compute $X = u_1P + u_2Q$;
6 If $X = \infty$ then return "Reject the signature";
7 Convert the x-coordinate x_1 of X to an integer $\bar{x}_1$ and compute $v = \bar{x}_1 \mod n$;
8 If $v = r$ then return "Accept the signature" ; else return "Reject the signature";

We recall that

— the domain parameters $D = (q, FR, S, a, b, P, n, h)$ stand for the field order q, an indication FR of the representation used for the elements of $\mathbb{F}_q$, two coefficients $a, b \in \mathbb{F}_q$ that define the equation of the elliptic curve E over $\mathbb{F}_q$ given in affine Weierstrass form, two field elements x_P and y_P in $\mathbb{F}_q$ that define a point $P = (x_P, y_P) \in E(\mathbb{F}_q)$ with prime order and called *the base point*, the order n of P and the cofactor $h = \#E(\mathbb{F}_q)/n$;

— H denotes a cryptographic hash function whose outputs have bitlength no more than that of n.

Questions :

(a) Prove that signature verification works.

(b) Show that if an adversary knows a single nonce k used for computing the signature (r, s) of a message m, then he is able to recover the private key d.

(c) Show that if the same nonce k is used to generate two ECDSA signatures, then an adversary is able to recover the private key d.

(d) Show that if H is not collision-resistant (i.e., we can find in polynomial running time two distinct inputs m_1 and m_2 such that $H(m_1) = H(m_2)$), then an adversary may be able to forge signatures.

(e) Show that if H is not preimage-resistant (i.e., for any hash value v, we can find in polynomial running time an input m such that $H(m) = v$), then an adversary can forge signatures (hint : consider a random integer l and compute r as the x-coordinate of $Q + lP$ reduced modulo n, then consider a message m such that $H(m) = rl \mod n$).

(f) We suppose in this question that the check $r \neq 0$ is not performed.
In order to minimize the size of domain parameters, Alice choses an elliptic curve in affine Weierstrass equation $y^2 = x^3 + ax + b$ over a prime field $\mathbb{F}_p$ $(p > 3)$ where b is a square modulo p (i.e., there exists b' such that $b'^2 = b \mod p$) and takes for the base point $P = (0, b')$ of prime order n.
Show that Eve is able to forge Alice's signature on any message of her choice.

5 *Dual EC pseudorandom generator*[1] : Let p be an odd prime number and E an EC over $\mathbb{F}_p$. Denote by $x : E(\mathbb{F}_p) \to \{0, \ldots, p-1\}$ a function which return the x-coordinate of a point in $E(\mathbb{F}_p)$

1. see NIST SP800-90

as a natural number $< p$. Assume $E(\mathbb{F}_p)$ cyclic and let P be a generator. Let Q be an other point of $E(\mathbb{F}_p)$ distinct than P. Let $\text{lsb}_i(x)$ the i least significant bits of a natural number (e.g., $\text{lsb}_3(23) = 7$). Fix also an integer b ("the number of extracted bits" per point). Let consider the following process : suppose given a "seed" $s_0 \in \{0,\dots,\#E(\mathbb{F}_p)-1\}$ and k a positive integer ($k > 0$). For $i = 1$ to k, compute $s_i = x(s_{i-1}P)$ and $r_i = \text{lsb}_b(x(s_iQ))$. The output is a sequence $r_1,\dots,r_k$ of bk bits (if p is 256 bits, we take $b = 240$). The goal is that if an attacker has the outputs r_i, he/she should not be able to recover the s_i.

Questions :

(a) Let $b = 240$ and p is 256 bits. Suppose an attacker knows a value α such that $\alpha Q = P$. Show that by computing all the possible values $x = u||r_i$ with $u < 2^{16}$ and testing which one correspond to a point of $E(\mathbb{F}_p)$ we can reconstruct the s_i (and thus deduce the next terms of the sequence).

(b) What countermeasures do you suggest?

6 *SMQV-C* : We propose the following key establishment protocol (derived from the MQV protocol) and called SMQV-C.
Let G be a cyclic group (denoted multiplicatively with 1 as neutral element) of prime order q and let g be a generator of G. We assume that solving the DLP is hard for G. Let H' be a cryptographic hash function which output ℓ-bits integers with ℓ =(size in bits of q)/2. We assume given a cryptographic MAC function and two key derivation functions KDF_1 and KDF_2 compatible with the setting. Let $\hat{A}$ and $\hat{B}$ be two entities who want to establish a common session key. We will denote $A \in G$ (respectively B) the public key of $\hat{A}$ (resp. $\hat{B}$) and $a \in \{1,\dots,q-1\}$ (resp. b) its private key. Namely, we have $A = g^a$ and $B = g^b$. If F is a cryptographic function (hash, MAC or key derivation), by $F(X_1,\dots,X_n)$ we mean $F(X_1||\cdots||X_n)$.
If any check fails then we abort the protocol and return FAILURE.
The protocol consists of the following steps :

(a) The initiator $\hat{A}$ choose randomly $x \in \{1,\dots,q-1\}$, compute $X = g^x$ and send (A,B,X) to $\hat{B}$

(b) At receipt $\hat{B}$ does the following :

  (i) Check that $X \in G$ with $X \neq 1$.

  (ii) Choose randomly $y \in \{1,\dots,q-1\}$, compute $Y = g^y$ and send (A,B,Y) to $\hat{A}$.

  (iii) Compute $d = H'(X,Y,A,B)$, $e = H'(Y,X,A,B)$, $s_B = (ye+b) \mod q$, $\sigma_B = (X^dA)^{s_B}$

  (iv) Compute $K_1 = KDF_1(\sigma_B,A,B,X,Y)$ and $t_B = MAC_{K_1}(B,Y)$.

  (v) Send (B,A,Y,t_B) to $\hat{A}$.

(c) At receipt of (B,A,Y,t_B), $\hat{A}$ does the following :

  (i) Check that $Y \in G$ with $Y \neq 1$.

  (ii) Compute $d = H'(X,Y,A,B)$, $e = H'(Y,X,A,B)$, $s_A = (xd+a) \mod q$, $\sigma_A = (Y^eB)^{s_A}$

  (iii) Compute $K_1 = KDF_1(\sigma_A,A,B,X,Y)$

  (iv) Check that $t_B = MAC_{K_1}(B,Y)$

  (v) Compute $t_A = MAC_{K_1}(A,X)$

  (vi) Send t_A to $\hat{B}$.

  (vii) Compute $K_2 = KDF_2(\sigma_A,A,B,X,Y)$

(d) At receipt of t_A, $\hat{B}$ does the following :

(i) Check that $t_A = MAC_{K_1}(A, X)$

(ii) Compute $K_2 = KDF_2(\sigma_B, A, B, X, Y)$

(e) The shared key is K_2

Questions :

(a) Show that the scheme works (i.e., $\hat{A}$ and $\hat{B}$ compute the same shared key K_2).

(b) What is the role of s_A, s_B, t_A and t_B?

(c) When G is an elliptic curve, compute the costs of SMQV-C in terms of; scalar multiplications (i.e., λZ with Z point on the curve and λ integer), simple addition on the curve, modular inversions, modular multiplications and modular additions. Compare (and discuss) the cost of SMQV-C with the one of ECDH.

optionnal part

Adding points on elliptic curves : Consider a curve in affine Weierstrass form $y^2 = x^3 + ax + b$ over an abstract field K (but, if it helps, you can think K as the field of rational numbers $\mathbb{Q}$ or the real numbers $\mathbb{R}$, even the complex numbers $\mathbb{C}$ or your favorite finite field). To be defined over K, means $a, b \in K$. We consider the "point at infinity" ∞ that you can think as a "point" of coordinates (∞, ∞). *We will nevertheless assume that the characteristic of K is different from 2, as in this case the situation degenerates.* Let denotes by $E(L)$ the set of L-rational points of the elliptic curves asssociated to the previous equation with L a field containing K. By construction, we have

$$E(L) = \{\infty\} \cup \{(x, y) \in L^2, \text{such that} y^2 = x^3 + ax + b\}.$$

Consider two points $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$ both on the curves with coordinates in L and distincts from the ∞. It means the following equations hold;

$$y_1^2 = x_1^3 + ax_1 + b,$$
$$y_2^2 = x_2^3 + ax_2 + b.$$

Now we want to compute P_3 as the sum of P_1 and P_2. Assume first, that $P_1 \neq P_2$. Draw the unique line D through P_1 and P_2. The general affine equation of such line is $\alpha x + \beta y + \gamma = 0$ (with $\alpha, \beta, \gamma \in L$). As both P_1 and P_2 are on D, it implies the following equations;

$$\alpha x_1 + \beta y_1 + \gamma = 0,$$
$$\alpha x_2 + \beta y_2 + \gamma = 0.$$

By substraction the above equations, we get $\alpha(x_1 - x_2) + \beta(y_1 - y_2) = 0$. If $x_1 = x_2$ we deduce $\beta = 0$ and if $y_1 = y_2$ then $\alpha = 0$. Notice that in both cases, we cannot have all coefficients equal to zero. Suppose $x_1 \neq x_2$. Then, we deduce that α and β are both non zeros (so, as we work over a fields, it means that they are both invertibles) and moreover

$$\frac{\alpha}{\beta} = -\frac{y_1 - y_2}{x_1 - x_2}.$$

Set $\alpha' = \frac{\alpha}{\beta}$ and $\gamma' = \frac{\gamma}{\beta}$. Dividing the equation of D by β, we get as new equation $y = -\alpha' x - \gamma'$ with $\alpha' = -\frac{y_1 - y_2}{x_1 - x_2}$. But as $y_2 = -\alpha' x_2 - \gamma'$, we deduce $\gamma' = \frac{y_1 - y_2}{x_1 - x_2} x_2 - y_2 = \frac{y_1 x_2 - x_1 y_2}{x_1 - x_2}$. Now, in order to compute the remaining point of the line intersecting the elliptic curves, we need to solve the equations

$$(\alpha' x + \gamma')^2 = x^3 + ax + b,$$

which leads to the equation

$$x^3 - \alpha'^2 x^2 + (a - 2\alpha'\gamma')x + b - \gamma'^2 = 0.$$

This last equation has three roots which are x_1, x_2 and x'_3 (check!). We need to obtain x'_3 in terms of our data. But from the factorisation

$$(x-x_1)(x-x_2)(x-x'_3)=0\,,$$

we deduce (check!)

$$x'_3=\alpha'^2-x_1-x_2\,,$$

and then $y'_3=-\alpha' x'_3-\gamma'$. Now, reflect across the x-axis to obtain the point $P_3=(x_3,y_3)$:

$$x_3=\alpha'^2-x_1-x_2\,,\quad y_3=\alpha' x_3+\gamma'=-\alpha'(x_1-x_3)-y_1\,.$$

In the case $x_1=x_2$ (but recall that $y_1\neq y_2$), the line through P_1 and P_2 is a vertical line, which therefore intersects the elliptic curve in ∞. We can even check, that the point P_3 computed above goes to (∞,∞) when $x_1=x_2$ using the formal rule "$\infty=\frac{1}{0}$". Therefore, in this case $P_1+P_2=\infty$.
Now consider the case where $P_1=P_2=(x_1,y_1)$. As shown in the lectures, we need to write the equation of the tangent line through P_1 that we will denote again D. Setting $f(x,y)=y^2-x^3-ax-b$, This equation is given by

$$\frac{\partial f}{\partial x}(x_1,y_1)(x-x_1)+\frac{\partial f}{\partial y}(x_1,y_1)(y-y_1)=0\,.$$

We thus obtain

$$(-3x_1^2-a)(x-x_1)+2y_1(y-y_1)=0\,.$$

If $y_1=0$, then again the line D is vertical and we get $2P_1=\infty$. Suppose $y_1\neq 0$. We observe that as the characteristic is different from 2, we have $2y_1\neq 0$ and the above equation becomes

$$y=\alpha(x-x_1)+y_1\,,\quad \alpha=\frac{3x_1^2+a}{2y_1}\,.$$

As before, we have a cubic equation $x^3-\alpha^2x^2+\cdots=0$. This time, x_1 is a double root of the equation (since D is the tangent line to the curve at P_1). Therefore, we obtain

$$x_3=\alpha-2x_1\,,\quad y_3=\alpha(x_1-x_3)-y_1\,.$$

Now, if $P_2=\infty$, the line through P_1 and ∞ is a vertical line that intersects the curve in the point P'_1 that is the relection of P_1 across the x-axis. As a result we get $P_1+\infty=P_1$. This is formaly extended to $\infty+\infty=\infty$. Notice that in projective coordinates, all the operations become natural without the need to add specific rules to the "point at infinity". It is also possible to write similar formula for a more general equation of the type $y^2=\alpha x^3+\beta x^2+\gamma x+\delta$.

1 Check carefully the formula given above for the addition.

2 Can we have an elliptic curve in affine Weierstrass form such that $(0,0)$ is a point of the curve?

3 Lists all the possible equations of elliptic curves in affine Weierstrass form over $\mathbb{Z}/p\mathbb{Z}$ for $p=2,3,5$.

4 Set $L=K=\mathbb{Q}$. Consider the (affine) equation $y^2=x^3-25x$.
- (a) Is it an elliptic curve over K?
- (b) Check that $(-4,6)$, $(0,0)$, $(5,0)$ are on the curve.
- (c) Compute $2(-4,6)$, $(0,0)+(-5,0)$, $2(0,0)$ and $2(-5,0)$.
- (d) Consider the same questions, but with $L=K=\mathbb{Z}/3\mathbb{Z}$.

5 Set $L=K=\mathbb{Q}$. Consider the equation $y^2=\frac{1}{6}x(x+1)(2x+1)$.
- (a) Check that this (affine) equation defines an elliptic curve over K.
- (b) Put this equation in affine Weierstrass form.
- (c) Check that $(0,0)$ and $(1,1)$ are on the curve.
- (d) Compute $(0,0)+(1,1)$ and $(\frac{1}{2},-\frac{1}{2})+(1,1)$.
- (e) Consider the same questions but with $L=K=\mathbb{Z}/5\mathbb{Z}$.