FTK3, WS 2023/24
5. Übungsblatt für den 11. bzw. 12.12.2023

1. Gegeben ist die elliptische Kurve ε: $y^2 = x^3 + 3x + 6$ über $\mathbb{Z}_{11}$. Bestimme die Ordnung der Kurve, indem du alle Punkte der Kurve ermittelst.

2. Gegeben ist wieder die elliptische Kurve ε: $y^2 = x^3 + 3x + 6$ über $\mathbb{Z}_{11}$, von der du bereits die Ordnung kennst. Überlege, welche Punktordnungen auf dieser Kurve überhaupt möglich sind. Bestimme dann die Ordnung der folgenden Punkte.

   (a) $(9, 5)$

   (b) $(4, 4)$

   (c) $(2, 3)$

3. Du kennst aus Beispiel 2 ein erzeugendes Element auf der elliptischen Kurve ε: $y^2 = x^3 + 3x + 6$ über $\mathbb{Z}_{11}$. Berechne daraus mithilfe von Satz 2.13 ein Element der Ordnung 5.

4. Gegeben ist die elliptische Kurve ε: $y^2 = x^3 + 28x + 42$ der primen Ordnung 103 über $\mathbb{Z}_{89}$. Berechne den diskreten Logarithmus von $(47, 28)$ zur Basis $(2, 27)$ auf ε mit dem Baby-Step-Giant-Step-Algorithmus.

5. Realisiere folgenden ECDH-Schlüsselaustausch. Alice und Bob einigen sich auf die elliptische Kurve ε: $y^2 = x^3 + 13x + 13$ über $\mathbb{Z}_{23}$ und den Punkt $G = (1, 2)$ mit Ordnung $\omega = 29$ auf ε.

   (a) Alice wählt zufällig $\alpha = 8$ und Bob wählt zufällig $\beta = 18$. Berechne, welche Nachrichten die beiden einander schicken.

   (b) Berechne den gemeinsamen Schlüssel, auf den die beiden sich so einigen.

6. Du wählst als ECDSA-Parameter die elliptische Kurve ε: $y^2 = x^3 + 5x + 200$ über $\mathbb{Z}_{601}$ und den Punkt $G = (3, 38)$ mit der Ordnung $\omega = 577$ auf ε. Als Private Key wählst du zufällig $\alpha = 281$.

   (a) Berechne deinen Public Key.

   (b) Berechne mit deinem Private Key eine Signatur für die Nachricht m mit dem Hashwert $h(m) = 333$ und $k = 3$.

   (c) Prüfe mit deinem Public Key die Signatur.

7. Finde für deine Lieblingsprogrammiersprache eine Kryptobibliothek, die Diffie-Hellman mit elliptischen Kurven unterstützt.

   (a) Welche Kurven stehen zur Auswahl? Welche davon sind klassische (Weierstrass-)Kurven, welche sind Montgomery-Kurven, welche sind Edwards-Kurven?

(b) Implementiere damit einen Diffie-Hellman-Schlüsselaustausch. Es ist ausreichend, wenn dein Code beide Parteien simuliert[1].

8. Finde für deine Lieblingsprogrammiersprache eine Kryptobibliothek, die Signaturen mit elliptischen Kurven unterstützt.

   (a) Welche Kurven/Verfahren stehen zur Auswahl? Welche davon sind klassische (Weierstrass-)Kurven, welche sind Montgomery-Kurven, welche sind Edwards-Kurven?

   (b) Erstelle damit einen Signaturschlüssel und eine Signatur über eine selbstgewählte Nachricht. Verifiziere die Signatur.

[1] Die Kommunikation zum Austausch der berechneten Werte muss also nicht implementiert werden.