16-124-836
Marcel ZAUDER

11.1 Computing with encrypted messages

ElGamal

The encoding process looks as follows:

$$Enc(pk, m) \;=\; (g^r, m \cdot Y^r)$$

For m_1 and m_2 we then get:

$$\begin{aligned} Enc(pk, m_1) &= (g^{r_1}, m_1 \cdot Y^{r_1}) \\ Enc(pk, m_2) &= (g^{r_2}, m_2 \cdot Y^{r_2}) \end{aligned}$$

For the operations $\otimes$ and $\oplus$ we then get:

$$\begin{aligned} Enc(pk, m_1) \otimes Enc(pk, m_2) &= (g^{r_1}, m_1 \cdot Y^{r_1}) \otimes (g^{r_2}, m_2 \cdot Y^{r_2}) \\ &= (g^{r_1} \otimes g^{r_2}, m_1 \cdot Y^{r_1} \otimes m_2 \cdot Y^{r_2}) \\ \textit{The } \otimes \textit{ can be replaced with a multiplication } (\cdot)&\textit{:} \\ &= (g^{r_1} \cdot g^{r_2}, m_1 \cdot Y^{r_1} \cdot m_2 \cdot Y^{r_2}) \\ &= (g^{r_1+r_2}, \underbrace{m_1 \cdot m_2}_{m_3} \cdot Y^{r_1+r_2}) \\ &= Enc(pk, m_3) \qquad \textit{with } m_3 = m_1 \cdot m_2 \end{aligned}$$

RSA

The encoding process looks as follows:

$$Enc(pk, m) \;:=\; m^{pk} \,\%N$$

For m_1 and m_2 we then get:

$$\begin{aligned} Enc(pk, m_1) &= m_1^{pk} \,\%N \\ Enc(pk, m_2) &= m_2^{pk} \,\%N \end{aligned}$$

For the operations $\otimes$ and $\oplus$ we then get:

$$\begin{aligned} Enc(pk, m_1) \otimes Enc(pk, m_2) &= m_1^{pk} \,\%N \otimes m_2^{pk} \,\%N \\ &= (m_1^{pk} \otimes m_2^{pk}) \,\%N \\ \textit{The } \otimes \textit{ can be replaced with a multiplication } (\cdot)&\textit{:} \\ &= (m_1^{pk} \cdot m_2^{pk}) \,\%N \\ &= (\underbrace{(m_1 \cdot m_2)}_{m_3}{}^{pk}) \,\%N \\ &= Enc(pk, m_3) \qquad \textit{with } m_3 = m_1 \cdot m_2 \end{aligned}$$

16-124-836
Marcel ZAUDER

11.2 RSA parameters

11.2.a Why e must be odd?

e must be **odd**, because if e would be **even** we can only reach **even** values ibn $\mathbb{G}$ (see Exercise 02).

11.2.b Given N and $\Phi(N)$

We have given:

$$\begin{aligned} N &= pq \\ \Phi(N) &= (p-1) \cdot (q-1) \end{aligned}$$

Therefore we can compute:

$$\left| \begin{array}{l} N = pq \\ \Phi(N) = (p-1) \cdot (q-1) \end{array} \right| \Leftrightarrow \left| \begin{array}{l} N = pq \\ q = \frac{\Phi(N)}{p-1} + 1 \end{array} \right| \Rightarrow N = p \cdot \left(\frac{\Phi(N)}{p-1} + 1 \right)$$

With this we can now compute p:

$$\begin{aligned} & N = p \cdot \left(\frac{\Phi(N)}{p-1} + 1 \right) \\ \Leftrightarrow \quad & N \cdot (p-1) = p \cdot \Phi(N) + p^2 - p \\ \Leftrightarrow \quad & 0 = p^2 + p \cdot (\Phi(N) - N - 1) + N \\ \Leftrightarrow \quad & p_{1/2} = -\frac{\Phi(N) - N - 1}{2} \pm \sqrt{\left(\frac{\Phi(N) - N - 1}{2} \right)^2 - N} \end{aligned}$$

The solutions p_1 and p_2 are then the prime factorization of N.

11.3 Bad choice of prime factors

11.3.a p is "small"

11.3.a $|\ p - q\ |$ is "small"

11.4 RSA oracle

We have:

$$\begin{aligned} public\ key &= \{N, e\} \\ ciphertext\ c & \end{aligned}$$

We can now choose an x and compute $c' = c \cdot x^e$. Because we know that $Enc(pk, m_1) \cdot Enc(pk, m_2) = Enc(pk, m_1 \cdot m_2)$, it follows that in the decryption Alice computes:

$$\begin{aligned} Dec(d, c') &= c'^d \bmod N = c^d \cdot x^{e^d} \bmod N \\ &= m \cdot x \bmod N \end{aligned}$$

With this we can recover the plaintext m.