## 12 Security II (MGK)

The RSA public-key crypto system performs calculations in the group $\mathbb{Z}_n$, with $n = pq$ being the product of two large prime numbers $p$ and $q$. The public key consists of the tuple $(n, e)$, with $\gcd(\phi(n), e) = 1$, and the corresponding private key is $(n, d)$. A message $m \in \mathbb{Z}_n$ is encrypted via $c = m^e \bmod n$ and decrypted as $m = c^d \bmod n$.

(*a*) Given $p$, $q$, and $e$, how can you apply the extended Euclidian algorithm to find a suitable $d$? [6 marks]

(*b*) If we modified RSA to use as the public modulus a prime number instead of a composite of two large prime numbers, that is $n = p$ instead of $n = pq$, would this affect its security, and if so how? [4 marks]

(*c*) In the *UltraSecure* virtual-private network, each router knows of each of its remote communication peers the RSA public key $(n, e)$, which all have $e = 3$ and $2^{1023} \leq n < 2^{1024}$. If router *alice* needs to establish a shared 256-bit AES secret key $k$ with remote router *bob*, it looks up *bob*'s $(n, e)$ and then uses this method:

- *alice* picks $k \in \{0, 1\}^{256}$ by reading 32 bytes from `/dev/random`
- *alice* interprets $k$ as binary integer $m$ with $0 \leq m < 2^{256}$
- *alice* sends $c = m^e \bmod n$ to *bob*
- *bob* decrypts $c$ into $m$ and recovers $k$ (by removing leading zeros)

Then *alice* and *bob* secure the rest of their communication with shared secret $k$.

(*i*) How could an eavesdropper obtain $m$ from $c$? [4 marks]

(*ii*) Suggest a better method of using RSA to establish an AES key than the one given above. [6 marks]