Zadání úlohy – RSA

Nyní ochutnávka veřejné (asymetrické) kryptografie

Pozor, opět samotná šifra je velice jednoduchá, ale ten balast okolo Vám dá trochu zabrat, takže opět - nepodcenit!!!

Popis RSA šifry najdete ve svých zápiscích z přednášky a jinak například na Wikipedii. Vytvořte 5 funkcí:
1. Generování veřejného a privátního klíče.
2. Převod textu do numerické reprezentace.
3. Šifrování pomocí veřejného klíče.
4. Dešifrování pomocí privátního klíče.
5. Převod z numerické hodnoty zpět na text

Stručný popis šifry:
Tato verze pracuje s běžnými znaky ASCII anglické abecedy, je Case Sensitive, takže rozeznává malé a velké písmena, je možno použít i čísla.

Generování klíčů:

1. Zvolí se dvě různá velká náhodná prvočísla p a q. (Využijte Random z šíleně velkého čísla(rozsahu čísel) (v rozsahu 19 číslic)
2. Spočítá se jejich součin $n = pq$.
3. Spočítá se hodnota Eulerovy funkce $\varphi(n) = (p - 1)(q - 1)$.
4. Zvolí se celé číslo $1 < e < \varphi(n)$, které je s $\varphi(n)$ nesoudělné. (tj největší společný dělitel je roven maximálně hodnotě 1)
5. Nalezne číslo d jako inverzni modulo pro vyraz e mod $\varphi(n)$ - Využijte funkci PowerMod a mrkněte do helpu, jak získat inverzní modulus

Veřejným klíčem je dvojice (n, e)
Soukromým klíčem je dvojice (n, d)

Převod textu do numerické reprezentace:
Existuje celá řada způsobů, jednoduchých i složitých, půjdeme zlatou střední cestou. Každý znak převedete do ASCII hodnoty a tuto do binární reprezentace. Poněvadž ale čísla mají např. 5 bitů, písmena 7 bitů atd, je nutné toto sjednotit, tj, je nutné projít všechny znaky a doplnit nuly v bitovém zápisu tak, aby všechny znaky měly rovnoměrný počet bitů (11). Z těchto jednotlivých 11-ti bitových zápisů vytvořte jedno super n-bitové číslo (88) a převeďte jej zpět do dekadické formy a máme hotovo. Pro opačný převod jen musíte brát v úvahu, že případné nuly na začátku velkého binárního čísla nebyly brány v potaz, takže teď by chyběly. Musíte proto opět vhodně doplnit na začátek tento bitový zápis, aby byl dělitelný hodnotou 11, tj, šel rozdělit na n bloků po 11 bitech. Zbývající postup jej již inverzí výše uvedeného. Pro rozdělení vstupního textu na bloky, využijte 8 B (to znamená po osmicích).

Šifrování:
Šifrování je jednoduchá matematická operace $c = m^e \bmod n$
kde m - zpráva, c - šifra

Dešifrování:
Dešifrování je opět jednoduchá matematická operace $m = c^d \bmod n$
kde m - zpráva, c - šifra
U obou využijete s výhodou funkci PowerMod alternativu pro Python 3.x
UVĚDOMTE SI, ŽE NENÍ MOŽNÉ PŘEVÉST CELÝ TEXT NA JEDNO HYPER ČÍSLO, NA TO BYSTE PŘIŠLI BRZO SAMI, ŽE TO NEFUNGUJE, JE POTŘEBA VYMYSLET NĚJAKÝ BLOKOVÝ PŘÍSTUP. (po osmicích)

Tento úkol je hodnocený a jeho správné, včasné a samostatné vypracování je podmínkou pro získání zápočtu.
Celkově je možno získat 10 bodů.

- 1,5b za převod textu do numerické reprezentace a zpět

- 2,5b za správné generování klíčů.
- 0,5b za funkci pro šifrování s blokovým přístupem
- 0,5b za funkci pro dešifrování s blokovým přístupem
- 4b za GUI (Pole pro zadání textu, pole pro zadání klíče pro dešifrování (složky n a d), pole pro zadání klíče pro šifrování (složky n a e), přepínač pro šifrování a dešifrování, pole pro zobrazení zašifrované zprávy (zobrazujeme numerické hodnoty po blocích), výpis klíčových párů).
- 1b za umělecký dojem

Bodování je opět orientační a může se měnit na základě kvality !!!