Úloha č. 1
Tučňáci z Madagaskaru

Zamysli se! 10 b

Tato úloha je čistě teoretická, tvým úkolem zde není napsat program. Namísto toho si dej záležet na kvalitním slovním popisu, kde mimo jiné jasně zdůvodníš, proč tvůj postup skutečně bude fungovat.

Naší ZOO se podařilo sehnat čtveřici moc roztomile vypadajících tučňáků. Nejenom, že jsou krásně *fluffy*, ale předběžné testy naznačují i velmi vysokou inteligenci. Jeden ze zaměstnanců byl zvědavý na to, co všechno dokážou, a proto se rozhodl jim zařídit přístup k internetu[1]. *What could possibly go wrong?*

Pro jistotu se ale rozhodl, že bude jejich síť monitorovat. Přece jenom, *co kdyby.*

A udělal dobře. Do ZOO sice proudí zásoby jídla, ale ryb je zoufale málo. A to především těch dobrých! Tučňákům už nestačí jenom usmívat se a mávat. Na vyřešení této prekérní situace bude potřeba pořádný plán.

Jednomu z tučňáků se podařilo objevit nedaleký obchod s rybami, který je nacpaný k prasknutí těmi nejvydatnějšími úlovky, kdo kdy viděl. Je tedy jasné, že řešení problému je na dosah. Stačí se v noci vyplížit ze ZOO a ukořistit pár rybiček.

Skupina čtyř tučňáků se tedy rozdělí na dvě menší. Dva tučňáci povedou výpad za hranice ZOO a zbylí dva budou v bezpečí doma. Naštěstí je moderní doba, a tak mezi sebou mají kvalitní internetové spojení. Tučňáci doma mají dobrý výhled po celé ZOO, a tak je budou moci varovat, kdyby se dělo něco nekalého. A výzkumná skupinka zase bude žádat o informace a podávat hlášení. Velitel si ale pořád není jistý, jestli jde o bezpečnou formu komunikace. Pomůžeš mu vymyslet takový způsob komunikace, aby určitě byli všichni v bezpečí?

Předpokládáme, že zprávy posílané výzkumnou skupinkou mohou být jak odposlechnuty, tak v tomto směru může být se sítí libovolně manipulováno (tedy zprávy mohou být třeba pozměněny). Pro zjednodušení slíbíme, že v opačném směru (tedy od základny k výzkumné skupince) může být síť pouze odposlechnuta, ale nemůže být pozměněna.

[1]: `https://www.youtube.com/watch?v=dLRLYPiaAoA`

Budeme tedy posílat zprávy mezi dvěma místy. Tvým cílem bude navrhnout protokol tak, aby na něj bylo obtížné zaútočit. Konkrétně je potřeba zabránit tomu, aby nepřítel dokázal něco nekalého. Například:

- zjistit, který typ zprávy je posílán;
- zjistit nebo odhadnout, co může obsahovat daná zpráva;
- změnit vyznění zpráv, které jsou přijaty (třeba změnou pořadí, nebo tím, že některé budou přerušeny);
- manipulovat s obsahem zprávy.

Čím více různých útoků znemožníš, tím víc dostaneš bodů. Některé útoky je možné znemožnit úplně, u něčeho stačí, když se dozvíme o tom, že útok proběhl. Za ošetření útoku, který není zmíněný v seznamu výše, můžeš získat bonusové body. (Víc než 10 bodů nezískáš, ale můžeš si tak vynahradit ztrátu bodů z jiné části).

Předpokládej, že spolu komunikují body $\boldsymbol{A}$ a $\boldsymbol{B}$. Po tobě chceme chránit pouze zprávy jdoucí z $\boldsymbol{A}$ do $\boldsymbol{B}$. Zprávy jdoucí z $\boldsymbol{B}$ do $\boldsymbol{A}$ jsou důvěryhodné (tedy nepřítel je nemůže pozměnit), ale jsou nepřítelem úspěšně odposlouchávané. Zároveň si před spuštěním mise můžeš vyměnit libovolné konstatní množství informací (například si dohodnout strategii a klíče). Nepřítel bude znát dohodnutou strategii, ale ne případné klíče.

Budeš tedy posílat zprávu od výzkumné skupiny na základnu po nehlídané síti, kterou má možnost nepřítel odposlouchávat a zároveň do ní zasahovat. Zpráva bude vždycky v jednom z následujících formátů:

- VYCKEJTE
- POKRACUJEM
- POSLETE MI n (kde $n \in \mathbb{N}, n \leq 10^6$)
- OPAKUJ
- STAHUJEME SE, NEMA TO CENU

K dispozici budeš mít několik speciálních kryptografických funkcí, jiné funkce tohoto typu nepoužívej. Zároveň můžeš používat všechny ostatní nekryptografické funkce. Cílem je navrhnout protokol. (Tedy můžeš používat například funkce *split* pro rozdělení textu, *append* pro spojení dvou řetězců, *date* pro získání momentálního času a data, nebo jakékoli další, dokud nějak nepracují s kryptografií).

Než se pustíme do popisu operací, které ti pomůžou splnit úkol, zavedeme si nové značení. Pokud matematiku úplně nemusíš, neboj se ho přeskočit, tento úkol můžeš splnit i tak. Možná to ale pomůže s formálním pochopením funkcí, které budeš mít k dispozici.

Definice 1. ***Abeceda*** *je jakákoli konečná množina. Prvky abecedy nazýváme* ***písmena*** *(příp.* ***znaky****) a jakýkoli řetězec n písmen $a_1 a_2 \cdots a_n$ kde n je celé nezáporné, nazýváme* ***slovo*** *délky n. Je-li $n = 0$, mluvíme o* ***prázdném slově*** *(značíme ε).*

Množinu všech (konečných) slov nad abecedou $\mathcal{A}$ značíme $\mathcal{A}^*$ a množinu všech konečných neprázdných slov nad abecedou $\mathcal{A}$ značíme $\mathcal{A}^+$. Množinu všech slov nad abecedou $\mathcal{A}$, jejichž délka je n, značíme $\mathcal{A}^n$.

Z předchozího je tedy důležité si odnést, že $\mathcal{A}^n$ je řetězec délky n.

Definice 2. ***Zobrazení*** *f je předpis, který každému prvku množiny $\mathbb{X}$ přiřadí právě jeden prvek množiny $\mathbb{Y}$. Zapisujeme $f : \mathbb{X} \to \mathbb{Y}$.*

Speciálně: zobrazení g z dvojice prvků z množin $\mathbb{A}$ a $\mathbb{B}$ do množiny $\mathbb{Y}$ zapisujeme $g : \mathbb{A} \times \mathbb{B} \to \mathbb{Y}$.

Z předchozího je tedy důležité si odnést, že $\mathcal{A}^n \times \mathcal{A}^k \to \mathcal{A}^n$ je značení pro (laicky řečeno) funkci, která bere dva textové řetězce – jeden délky n, druhý délky k – a vrátí textový řetězec délky n.

A teď se už můžeme pustit do popisu kryptografických funkcí, které budeme moct v popisu protokolu použít.

- *Zašifruj:* $\mathcal{A}^n \times \mathcal{A}^k \to \mathcal{A}^n$

  Tato funkce bere zprávu délky n a šifrovací klíč délky k, a vytvoří zašifrovanou zprávu velikosti n. Zároveň platí, že pokud nepřítel v zašifrované zprávě změní třeba páté písmeno, bude páté písmeno změněno i po dešifrování (ne nutně však na stejnou hodnotu). Toto je vlastnost, která se skutečně často objevuje u symetrických šifer, kdy flipnuté bity poškodí dešifrovaný text ve stejné oblasti.

- *Dešifruj:* $\mathcal{A}^n \times \mathcal{A}^k \to \mathcal{A}^n$

  Tato funkce bere zašifrovanou zprávu délky n a šifrovací klíč délky k, a dešifruje zprávu do textu velikosti n.

- *Otisk:* $\mathcal{A}^n \to \mathcal{A}^k$

  Tato funkce bere zprávu délky n, a vytvoří její otisk, a to text délky k (kde typicky $k \ll n$ pro netriviální n). Z tohoto textu délky k nelze v rozumném čase vypočítat původní zprávu délky n. Tedy, tato funkce vytvoří dokonalý *hash* zprávy.

- *Náhodná zpráva:* $\emptyset \to \mathcal{A}^n$

  Tato funkce vytvoří náhodnou zprávu délky n. Na základě předchozí vygenerované zprávy nelze předvídat následující.

Bodu **B** se bude posílat mnoho zpráv a chceš se ujistit, že před nepřítelem zůstanou v bezpečí. Tvým úkolem bude navrhnout protokol (tedy sérii kroků, které uděláš před odesláním zprávy, respektive které se provedou po přijetí zprávy) pro přenos dat. Snaž se ho udělat co nejodolnější jak vůči odposlechům, tak zásahům zvenčí. Tato úloha nemá jediné správné řešení. Plné body dostaneš, pokud ve tvém protokolu nebude žádná zranitelnost, kterou by mohl nepřítel zneužít. Snaž se také popsat, proč jednotlivé kroky v protokolu využíváš, a proti kterému útoku se pomocí nich bráníš.

Ukázkové vstupy

Zde je příklad protokolu, který je zranitelný na několika místech. Je pouze velmi stručně rozepsaný, po tobě budeme chtít, abys každý krok toho svého detailně odůvodnil. (Ukázkové řešení by za tento popis mnoho bodů nezískalo!)

Protokol

Výchozí zprávu $\mathcal{A}^n$ bez jakýchkoli úprav zašifruji pomocí předem dohodnutého klíče (který jsem si dohodl s bodem **B** před začátkem mise, takže ho nepřítel nezná) operací *Zašifruj*. Výslednou zašifrovanou zprávu délky n pošlu bodu **B**, kde bude pomocí stejného klíče dešifrována operací *Dešifruj*.

Problém

Nepřítel si může zapamatovat odeslané zašifrované zprávy, a poté je posílat základně **B** sám. Jelikož má kontrolu nad sítí, základna nepozná, jestli zprávy posíláme my, nebo nepřítel. Zprávy od nepřítele tedy bude pokládat za naše, což určitě nechceme. Nepřítel nemusí vědět, co zpráva ve skutečnosti znamená – důležité je, že umí základnu zmást a donutit ji udělat něco, co my nechceme. (Pokud tě tento druh útoků zajímá, najdi si tzv. *Replay attack*.)

Nutno podotknout, že tento jednoduchý protokol obsahuje víc, než pouze tuto jednu chybu. Zvládneš je najít a popsat?

Hodnocení

Z celé úlohy můžeš dostat až deset bodů. Tato úloha má tři části.

a) *Návrh protokolu (5b)* – je potřeba jasně a srozumitelně navrhnout protokol pro šifrování a dešifrování zpráv tak, aby bylo zamezeno co nejvíce útokům. Není potřeba psát kód ani pseudokód, hrubý popis bohatě postačí. Můžeš používat jakékoli funkce a algoritmy, kromě kryptografických – tam používej jenom ty, které jsme si definovali.

b) *Odůvodnění správnosti (2b)* – je potřeba jasně říct, proč vypadá navržený protokol tak, jak vypadá. Ideálně by u každé části protokolu mělo být napsáno, proč tam je, a kterému útoku zamezuje.

c) *Útok na ukázkový protokol (3b)* – ukázkový protokol obsahuje mnoho zranitelností. Za nalezení alespoň některých z nich je možné získat až tři body. Je potřeba ale pečlivě zdůvodnit, jak by se zranitelnost zneužila, a jak by se útočilo.