Máme tu man in the middle attack. Základna A komunikuje se základnou B. Man in the middle dokáže číst všechny zprávy. Zároveň když posílá základna A zprávu do základny B, tak má man in the middle schopnost zprávu pozměňovat. Naším cílem je, aby man in the middle nedokázal úspěšně odposlechnout (a pochopit) o co ve zprávě jde, a také aby když zprávu pozmění, tak aby to základna B poznala.

1. A i B mají stejný seed na náhodnou generaci.
2. Oba náhodně vygenerují stejný klíč.
3. B zašifruje zprávu a pošle ji.
4. A ji rozšifruje klíčem a vygeneruje si další klíč s použitím seedu.
5. A podepíše zprávu a pošle ji B.
   1. Zprávě přidáme padding
   2. Vytvoříme hash ze zprávy, který bude fungovat jako podpis
   3. Podpis se zašifruje pomocí vygenerovaného klíče
   4. zpráva se zašifruje pomocí stejného klíče
   5. Pošleme zprávu a podpis
6. B vygeneruje stejný klíč jako A a ověří podepsanou zprávu
   1. Zprávu dešifruje pomocí vygenerovaného klíče
   2. Podpis dešifruje pomocí stejného klíče
   3. Ze zprávy vytvoří Hash
   4. Porovná hash zprávy a podpisu, pokud jsou stejné, tak je zpráva ověřena
   5. Odstraníme padding ze zprávy a přečteme
7. Opakujeme od kroku 1

# Fiks 2021/2022 úloha č. 10 „Tučňáci z Madagaskaru“

**Kryštof Olík**

Tučňáci, značení jako A, se vydávají z naší základny, značenou jako B, na tajnou misi a potřebují se základnou komunikovat. Je tu však jeden háček, mezi našimi zprávy je tak zvaný Man in the middle, značený jako M, který dokáže naše zprávy číst a dokonce zprávy mířené z A do B dokáže pozměnit. Naším cílem je vymyslet bezpečný protokol komunikace, tak, aby M nedokázal pochopit o co ve zprávě jde, abychom poznali zda útočník zprávu nějakým způsobem pozměnil a abychom dokázali přečíst zprávy i po změně pořadí a přerušení.

## Návrh protokolu a odůvodnění

Ještě před tím, než se tučňáci vydají na cestu, tak si se základnou domluví nějaký společný náhodný seed, který budeme používat na pseudo-náhodnou generaci klíčů. Tímto zajistíme, že obě strany pokaždé vygenerují nové a stejné klíče. Poté co se vydá A na cestu, tak můžeme začít bezpečně komunikovat. Řekněme, že první potřebujeme poslat zprávu z B do A.

Nejprve si náhodně vygenerujeme klíč s použitím společného seedu. Pokaždé používáme jiný náhodný klíč na šifrování, aby M neměl možnost rozpoznávat stejné zprávy a zároveň k tomu, aby nemohl náhodou naše zprávy znovu použít a poslat. Jelikož bývají naše zprávy krátké, tak k ním přidáme trochu výplně (náhodná výplň, která nemá s naší zprávou nic společného). To děláme jenom pro to, abychom ještě více znemožnili možnost zprávu dešifrovat metodou bruteforce. Zprávu poté zašifrujeme naším klíčem a pošleme jí A.

Když zpráva dorazí do A, tak vygenerujeme stejný klíč (opět, protože máme stejný začáteční seed), zprávu dešifrujeme, výplň odstraníme a nakonec si zprávu můžeme přečíst. Avšak teďka, když chceme poslat zprávu z A do B, musíme zprávu před odesláním více ošetřit, jelikož je M schopný zprávu pozměnit. Aby B poznalo, zda bylo se zprávou od A manipulováno, tak musíme zprávu tzv. podepsat.

Vygenerujeme nový klíč a dáme se do procesu podepisování. Nejprve přidáme k naší zprávě výplň a poté z ní vytvoříme otisk. Tedy, vytvoříme tzv. hash této zprávy s výplní. Tomuto říkáme podpis. Podp