Cheatsheet ŘSSS-ŘK

Tento dokument by měl sloužit jako cheatsheet a ke každému tématu poskytovat vždycky 1-2 věty jako pick-up line, když se ho na dané téma zeptají. Např. “Cocomo - Model používaný k odhadování ceny SW. Idea je taková, že cena vývoje aplikace přímo závisí na velikosti SW.”

tags: `řsss-řk`, `cheatsheet`

01 Teorie kódování

Cílem teorie kódování je vytvořit systémy a metody, které nám dovolí detekovat/opravovat chyby způsobené přenosem informací přes hlučné kanály (noisy channels). Kódování je používáno pro kompresi dat, v kryptografii a error-correction pro přenos informací.

Rozdělení: 1. Noiseless coding theory - Shannon entropie - Huffman coding 2. Noisy coding theory - Error correcting codes - Block codes

Typy kanálů

Hlavní typy kanálů jsou diskrétní (*discrete*) a kontinuální/spojité (*continous*) kanály.

Diskrétní Shannonův stochastický kanál je trojice $(\sum, \Omega, p)$, kde: - $\sum$ je vstupní abeceda - Ω je výstupní abeceda - Pr je pravděpodobnostní distribuce $\sum \times \Omega$ a pro každé $i \in \sum, o \in \Omega, Pr(i, o)$ je pravděpodobnost, že výstup z kanálu je o pokud je vstup i.

Kódovací metody

1. BEC (Backward error correction)
    - Příjemci dovoluje pouze detekovat chyby.
2. FEC (Forward error correction)
    - Příjemci dovoluje opravit určité množství chyb.

Hammingova vzdálenost

- Podobnost dvou kódových slov
- $h(x, y)$
- **Minimální vzdálenost kódu je důležitá vlastnost kódu a značí nejmenší počet chyb, které změní jedno kódové slovo do druhého.**

Entropie

- Shannonova entropie značí střední hodnotu množství dat v zasílané informaci.
- Je měřena v bitech.
- Entropie je definována jako:
    - $S(X) = -\sum_x p(x) \log p(x)$
- Nulová entropie - dokážeme předpovědět výstup v každém případě.
- Maximální entropie - všechny hodnoty generuje se stejnou pravděpodobností.

Shannonův teorém

- Shannonův teorém říká, že pro přenos n hodnot informace X, potřebujeme použít $nS(X)$ bitů.

Kódy s proměnnou délkou

- Takový kód, který mapuje zdrojové symboly na proměnnou délku bitů

Třídy: - Non singular codes - $\{a \mapsto 1, b \mapsto 011, c \mapsto 1100, d \mapsto 11001\}$ - Uniquely decodable codes - $\{a \mapsto 0, b \mapsto 01, c \mapsto 011\}$ - Preffix/Suffix codes - $\{a \mapsto 0, b \mapsto 10, c \mapsto 110, d \mapsto 111\}$

Huffmanovo kódování

- Algoritmus pro bezeztrátovou kompresi dat.
- Konvertuje znaky vstupního souboru do bitových řetězců různé délky podle frekvence výskytu.

Extended Huffman coding

- Kóduje sekvence zdrojových symbolů a ne jednotlivé symboly.

Generování skutečně- a pseudo-náhodných sekvencí.

Náhodná data jsou stěžejní pro kryptografické klíče, *padding values*, nebo *nonces*. Potřebujeme kvalitní (aby splňovala statistické vlastnosti) a nepředvídatelná data. Nicméně v deterministickém prostředí (= počítače), je občas složité získat opravdu náhodná data v rozumném čase.

True random number generators (TRNGs)

Bývají pomalé a kvalita závisí na zdroji náhodnosti: - **Ideální** - nedeterministický fyzický jev - **Excelentní** - HW-based - **Dobré** - Jakýkoliv vstup od uživatele - **Přijatelné** - SW-based - **Špatné** - předvídatelné - systémové datum a čas

Pseudo-random number generators (PRNGs)

Deterministický konečný automat

Linear feedback shift register (LFSR)

Shift register, kde vstupní bit je lineární funkcí jeho předchozího stavu.

Kryptografické protokoly

Protokol

- Algoritmus definovaný sekvencí kroků, které musí splnit každý (dva a více) účastník komunikace, abychom dosáhli cíle protokolu.
  - Cíle: důvěryhodnost, autentizace, integrita, ustanovení klíčů, non-repudiation

Nonce

- *Number used once*
  - Unikátní/náhodné
  - Sekvenční
  - Timestamps

Protokoly pro ustanovení klíčů

- Ustanovené klíče jsou sdíleným *secretem*
- *Key transport*
- *Key agreement*
- Koncepty jistoty
  - *Implicit key authentication*
  - *Key confirmation*
  - *Explicit key authentication*

    - *Entity authentication*
- Session klíče:
    - Short-term klíče
    - *Perfect forward secrecy* znamená, že kompromitace dlouhodobého klíče nijak nekompromituje zpětně session-klíče.
        - Budoucí odhalení klíče nekompromituje zprávy z minulosti
    - *Perfect backward secrecy* kompromitace klíče z minulosti nijak neovlivní zprávy v budoucnosti.

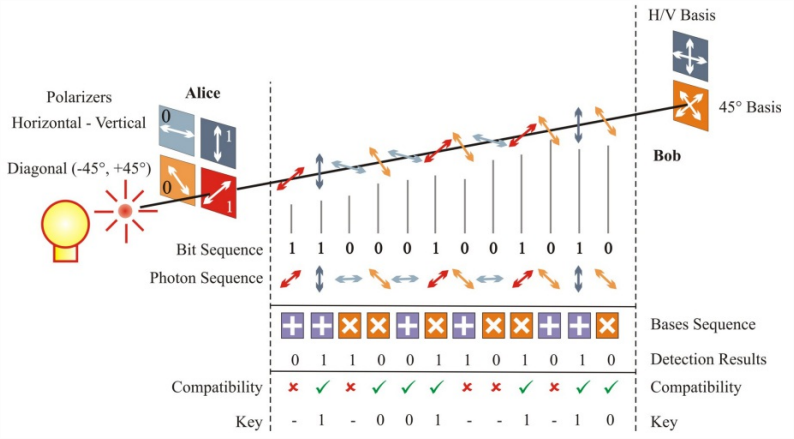
Zero-knowledge protokoly

- Dovoluje účastníkům komunikace dokázat, že drží určitý *secret*, aniž by odhalili jakékoliv informace, které by ho mohly kompromitovat.
- *Completeness*
- *Soundness*

Kvantová kryptografie

- Založena na kvantové fyzice (přírodní zákony), ne na předpokladu, že některé problémy je velice složité vypočítat.

BB-84 shared key establishment

- Ustanovení sdíleného klíče za pomocí kvantového kanálu.
- 

02 Symetrické a asymetrické šifry

Symetrická kryptografie

Matematická definice:

- *Symmetric cryptosystem* je pětice (P, C, K, E, D):
    - P: množina všech možných plaintextů nad danou abecedou
    - C: množina všech možných ciphertextů
    - K: množina všechn možných klíčů
    - E: šifrovací funkce
    - D: dešifrovací funkce
- Měl by splňovat jednoduché spočítání ciphertextu a generování klíčů, složité cracknout ciphertext.
- **Kerckhoffsův princip**: Bezpečnost kryptosystému musí záviset pouze na klíči a prozrazení principu šifrování ho nemá ohrozit.

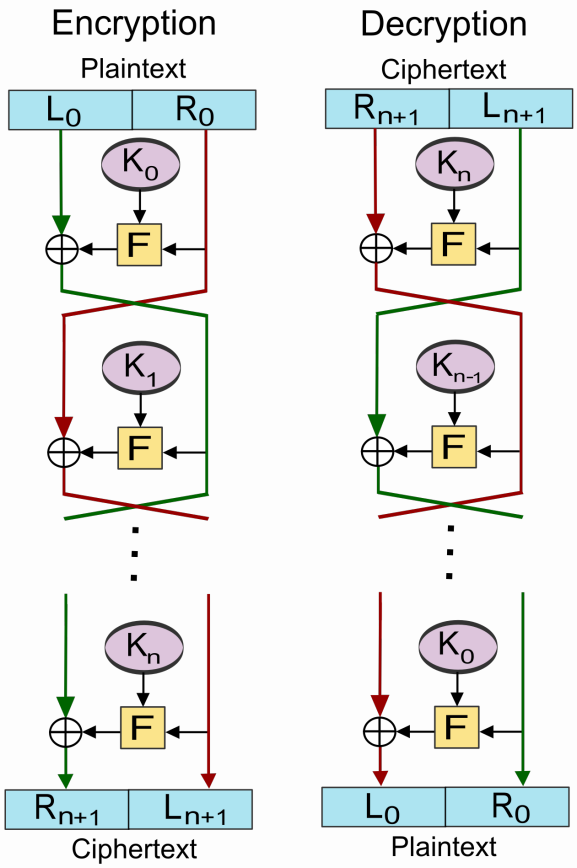
Blokové šifry

- Symetrické šifry, které procesují data v blocích o fixní délce bitů.
  - DES, AES

Stream (proudové) šifry

- Symetrické šifry, které procesují data jako stream bitů.
- Generuje keystream a XORuje s ním plaintext data bit po bitu.
  - One Time Pad

Feistelovy šifry



DES - Data encryption standard

- Délka bloku: 64 bitů
- Délka klíče: 56b + 8 paritních
- Počet kol: 16
- Algoritmus si předpočítá 16 subklíčů o délce 48 bitů pro každé kolo a pak šifruje stejně jako ve Feistelově šifře.

Triple DES

- DES používá relativně krátký klíč. Triple DES provede tři operace na jeden blok dat
- $E_{K3}(D_{K2}(E_{K1}(plaintext)))$

Meet-in-the-middle attack

- Tl;dr Příklad s 2DES - Vezmeme $E_{K1}(p) = X \rightarrow E_{K2}(X) = c$, pro dešifrování potřebujeme vypočítat $D_{K2}(c) = X \rightarrow D_{K1}(X) = p$. Z toho vidíme, že útočník stačí cracknout 2 klíče o délce 2^{56}.

Asymetrická kryptografie

- Použití dvojice klíčů - veřejný a privátní.
- 3 hlavní použití
  - Šifrování/dešifrování
  - Digitální podpisy
  - Inicializace sdíleného tajemství
- Pomalejší výpočet a potřeba key managementu.

RSA

- Založen na výpočetní náročnosti faktorizace prvočísel.

DSA (Digital Signature Algorithm)

- Asymetrická kryptografie umožňuje taky podpis digitální zprávy.
- Pokud chceme šifrovat i podepisovat, musíme nejdřív zprávu podepsat a až pak zašifrovat - abychom měli jednoznačně spojenou zprávu+podpis.

Diffie-Hellman (DH)

- Algoritmus, který umožňuje přes nezabezpečený kanál vytvořit mezi komunikujícími stranami šifrované spojení, bez předchozího dohodnutí šifrovacího klíče.

Hybrid encryption

- Hybridní šifrování kombinuje symetrickou a asymetrickou kryptografii.
- Asymetrická pro ustanovení klíčů. Symetrická pro šifrování.

Faktorizace a testování prvočíselnosti

- V kryptografii: Vynásobit dvě prvočísla je jednoduché, ale extrémně pomalé udělat opak. Tzn. z x získat dvě prvočísla.
- **Jednoduché algoritmy:**
  - Brute-force
- Obecné pravděpodobnostní algoritmy
  - **Fermatův test**
    - Založen na malé fermatově větě
    - Neodhalí carmichaelova čísla.
  - **Algoritmus Rabin-Miller**

Kryptosystémy na bázi eliptických křivek.

- ECC je způsob asymetrické kryptografie založené na vlastnostech eliptických křivek.
- Dvojice (E, O), kde:
  - E je set bodů v rovině daných Weierstrassovou funkcí
    - $y^2 = x^3 + ax + b$
  - $O \in E$ je zvolej jako "*point at infinity*" na křivce.

Principy konstrukce hashovacích funkcí

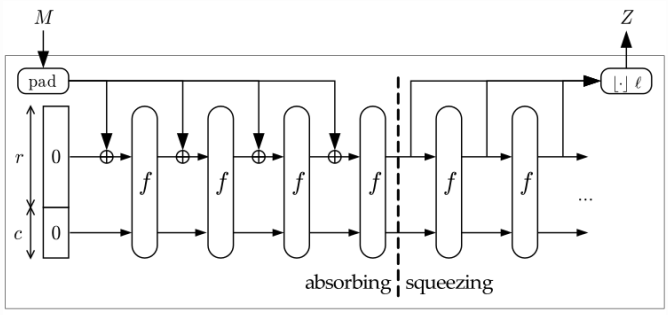
- $h : \{0,1\}^a \rightarrow \{0,1\}^b$, kde $a, b \in \mathbb{N}$
  - $a >> b$
    - a je mnohem větší, než b
  - a je libovolné a b ma fixní délku pro svou funkci
- Jinými slovy, hash funkce h promítne hodnotu z množiny s hodně (nebo dokonce nekonečno) členy na hodnotu z množiny s pevným počtem (méně) členy.
- **Vlastnosti dobré hash funkce:**
  - **Deterministická**: Jeden vstup vyprodukuje vždy stejný výstup
  - **One-way**: Z hashe nelze vypočítat vstup
  - **Weak collision resistant**: Pro jeden konkrétní vstup nelze najít jiný vstup, který vyprodukuje stejný hash
  - **Strong collision resistant**: Je složité najít jakékoliv x, y, takové, že $H(y) = H(x)$
  - **Strict avalanche criterion**: Malá změna na vstupu by měla způsobit velkou změnu na výstupu.
  - **Fast**

Merkle-Damgard construction of hash function

- Message M je rozdělena do bloků o fixní délce $M = m_0||m_1||\ldots||m_t$ a je použit vhodný *padding*

Sponge construction of hash functions: SHA-3 Keccak

- SHA-3 (Secure Hash Algorithm) je standard, Keccak je název algoritmu.
- Délka hashe: 224 bitů do 512 bitů
- SHA-3 pracuje s bloky o velikosti w (běžně 64 bitů; *padding* je použit, pokud je to potřeba). Uchovává si interní stav $5 \times 5 \times w$ jako pole, které se nazývá *sponge*.
- SHA-3 má dvě hlavní fáze: V první jsou data *absorbed* do *sponge* a ve druhé je výsledek *squeezed*.



- sponge
- keccak

Password hashing functions

bcrypt (1999) je hash funkce určena hashování hesel. Je založena na Blowfish block cipher. Má v sobě automaticky zahrnut salt, viz. výše. - $**2a**$**12**$R9h/cIPz0gi.URNNX3kh2OPST9/PgBkqquzi.Ss7KIUgO2t0jWMUW - **2a** - specifikuje verzi algoritmu - **12** - specifikuje *input cost*, tzn. počet iterací - čím víc, tím pomalejší je výpočet = zpomalení útočníka. **scrypt** (2009) je další hash funkce pro hashování hesel, navrhnuta tak, aby byla náročně vypočitatelná pomocí custom HW.

---

03 Aplikace šifrování

Bezpečnostní cíle

- **Důvěrnost:** Zprávu mohou číst jen autorizované subjekty
  - Zajistíme šifrováním
- **Integrita:** Zpráva není nijak modifikována neautorizovaným subjektem - Detekci zajistíme použitím MAC, hash + podpis
- **Autentizace:** Ověření identity zařízení, procesu, původce zprávy
  - Podpis, heslo/PIN, biometrika, karta

Symetrické blokové šifry

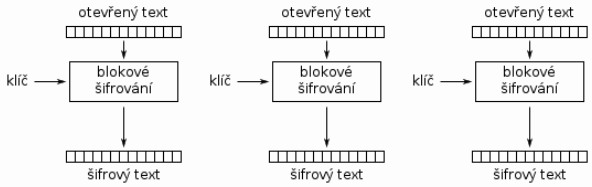
- Stejný klíč pro šifrování a dešifrování sdílený mezi oběma účastněnými se stranami
- Blokové a proudové šifry
- **Výhody**: rychlejší než asymetrické šifrovací algoritmy, menší klíče
- **Nevýhody**: problémem je distribuce klíče

Implementační problémy pro symetrické šifrování: - **Zarovnání (padding)** - Délka plaintextu musí být dělitelná požadovanou velikostí bloku - Padding slouží jako výplň, po dešifrování je odstraněno - **IV (inicializační vektor)**: - Přidává do šifrování více náhodnosti - Cílem je nešifrovat dva stejné bloky na stejný výsledný šifrovaný text

Operační módy / provozní režimy (DES & AES)

Electronic CodeBook (ECB), blokový mód

- Zpráva je rozdělena do 64bitových nezávislých bloků, které jsou samostatně zašifrovány.
- $C_i = E_K(P_i)$

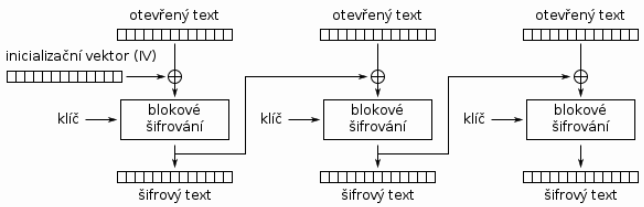


Šifrování v režimu kódové knihy (ECB)

- šifrování v ECB režimu

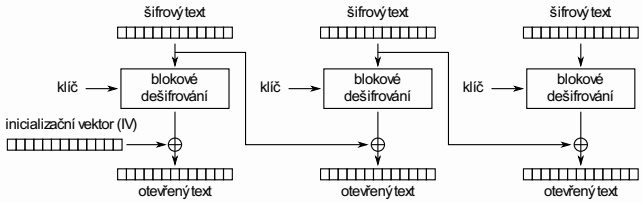
Cipher Block Chaining (CBC), blokový mód

- Zpráva je rozdělena do 64bitových bloků, které jsou XORovány s předchozím šifrovým blokem (začíná se s IV)
- $C_i = E_K(P_i \oplus C_{i-1})$, kde $C_{-1} = IV$



Šifrování v režimu řetězení šifrových bloků (CBC)

- šifrování v CBC režimu

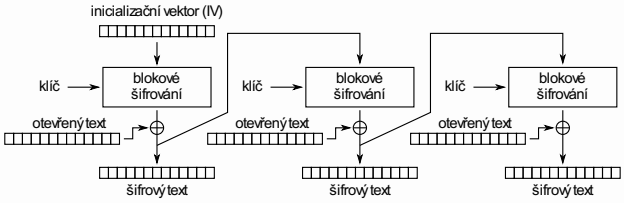


Dešifrování v režimu řetězení šifrových bloků (CBC)

- dešifrování v CBC režimu

Cipher FeedBack (CFB), proudový mód

- Zpráva se zpracovává jako proud bitů a přidává se k výstupu (XOR) a výsledek se užívá v další fázi
- $C_i = E_K(C_{i-1}) \oplus P_i$, kde $C_{-1} = IV$

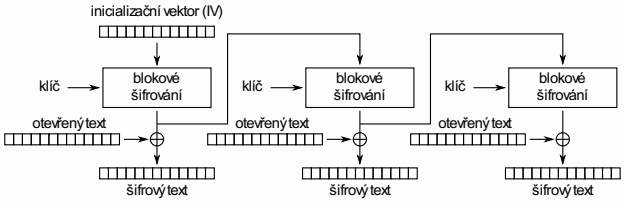


Šifrování v režimu šifrové zpětné vazby (CFB)

- šifrování v CFB módu

Output Feedback (OFB), proudový mód

- Zpráva se zpracovává jako proud bitů, přidána ke zprávě, ale feedback je nezávislý na zprávě
- $C_i = O_i \oplus P_i$, kde $O_i = E_K(O_{i-1})$ a $O_{-1} = IV$

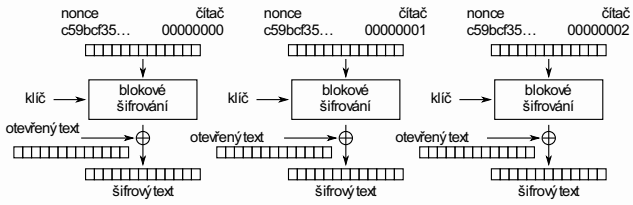


Šifrování v režimu zpětné vazby z výstupu (OFB)

- šifrování v OFB módu

Counter (CTR), proudový mód

- Podobné OFB, ale šifruje počítadlo (*nonce*) místo feedback hodnoty
- ◦ $C_i = P_i \oplus E_K(i)$ (iniciální hodnota počítadla musí být náhodná)



Šifrování v čítačovém režimu (CTR)

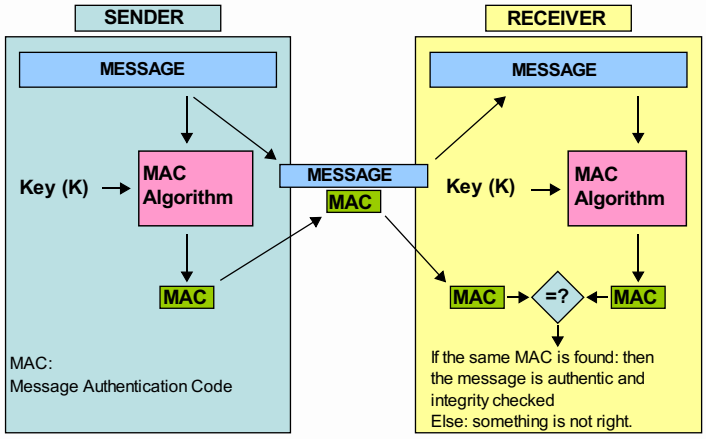
- šifrování v CTR módu

Shrnutí vybraných vlastností šifrovacích módů

	ECB	CBC	CFB	
Předpočítání	:x:	:x:	:x:	:heavy
Paralelní šifrování	:heavy_check_mark:	:x:	:x:	:x:
Paralelní dešifrování	:heavy_check_mark:	:heavy_check_mark:	:heavy_check_mark:	:x:
Náhodný přístup	:heavy_check_mark:	:heavy_check_mark:	:heavy_check_mark:	:x:
Porušených bloků plaintextu při změně 1 šifrového bloku	1	2	2	1

Message Authentication Code (MAC)

- Kryptografický kontrolní součet s pevnou délkou pro variabilně velké zprávy.
- Malý blok generovaný pomocí blokové šifry nebo hashovací funkce.
- Poskytuje integritu a autentizaci



Message Authentication Code příklad

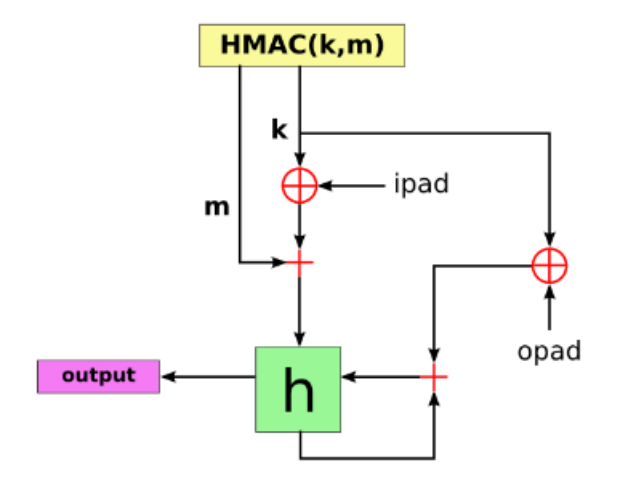
Implementace MAC

Pomocí symetrické šifry pro MAC (DAA - Data Authentication Algorithm)

- Jakýkoliv mód blokové šifry se zřetězením může být aplikován a finální blok slouží jako MAC.

Pomocí hashovacích funkcí pro MAC (HMAC)

- Mac založené na hashovací funkci jsou žádoucí, jelikož hashovací funkce jsou obecně rychlejší a kód pro kryptografické hashovací funkce je široce dostupný.
- `HMAC = Hash[(K+ XOR opad) || Hash[K+ XOR ipad || M]]`
- `K+` je klíč zarovnán nulami na požadovanou délku
- `opad`, `ipad` jsou specifické zarovnávací konstanty definovavé standardem.



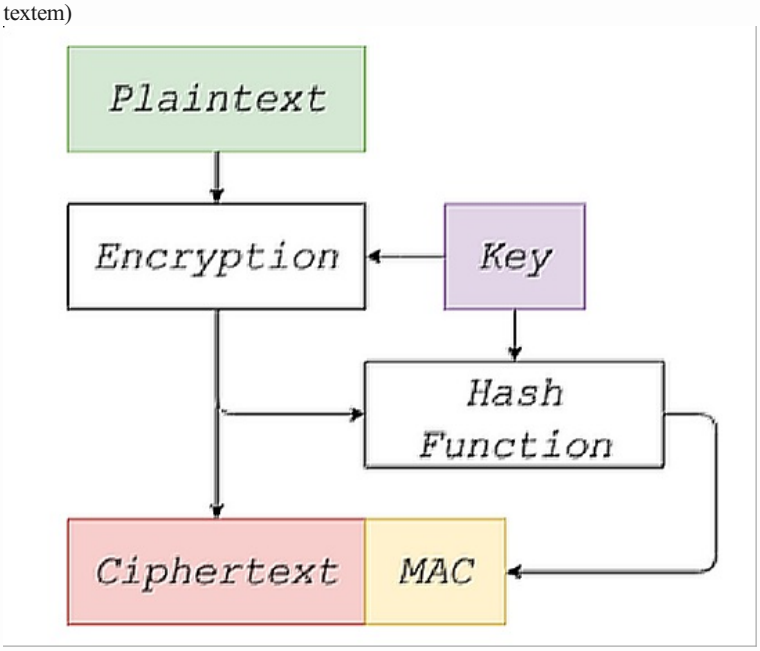
HMAC příklad

Autentizované šifrování

- Obyčejné šifrování poskytuje důvernost, ale ne integritu nebo autentizaci
- Výsledkem autentizovaného šifrování je šifrovaný text a jeho MAC

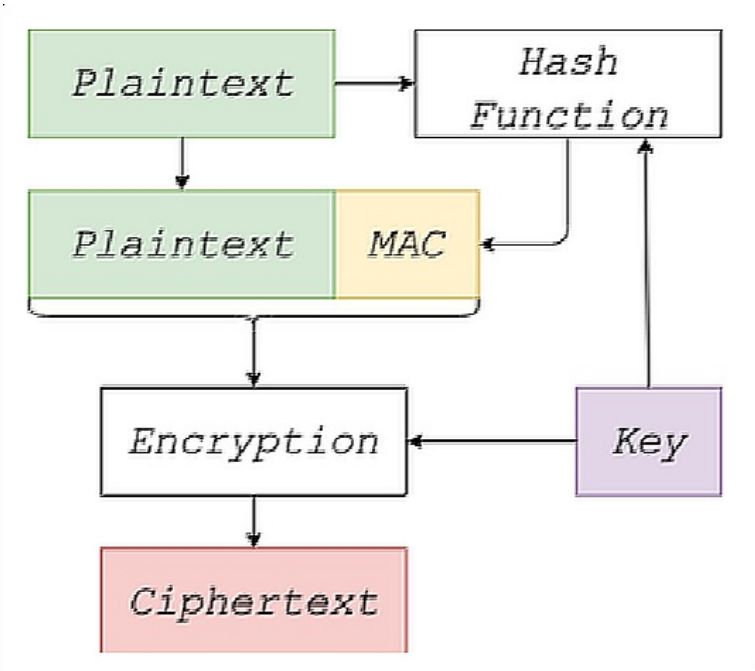
Encrypt-then-MAC (or Encrypt-then-Authenticate)

- Zašifrovat plaintext, pak spočítat MAC nad šifrovaným textem a připojit to k šifrovanému textu (zahrnuje IV a šifrovací metodu)
  - Integrita plaintextu a šifrovaného textu
  - MAC neodhaluje žádné info o plaintextu (protože MAC je spočítán nad šifrovaným textem)



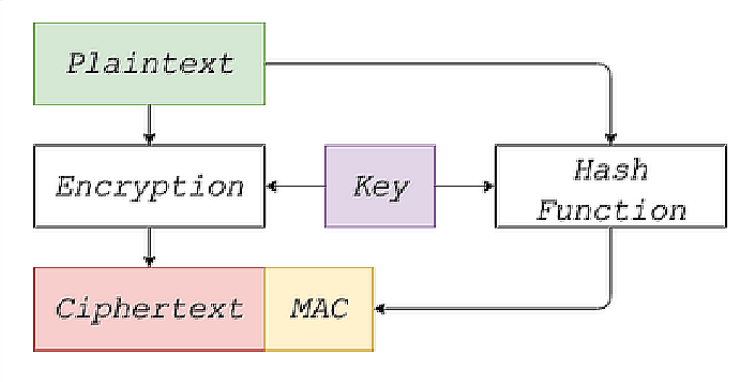
MAC-then-Encrypt (or Authenticate-then-Encrypt)

- Spočítat MAC nad plaintextem, připojit to k datům a pak celé zašifrovat
  - Integrita plaintextu, ale ne šifrovaného textu (může být měněn)
  - MAC neodhaluje žádné info o plaintextu (protože je šifrován)



Encrypt-and-MAC (or Encrypt-and-Authenticate)

- Spočítat MAC nad plaintextem, zašifrovat plaintext a pak připojit MAC na konec šifrovaného textu
  - Integrita plaintextu, ale ne šifrovaného textu (může být měněn)
  - MAC může odhalit informaci o plaintextu



Hybridní kryptosystémy

- Kombinace symetrické a asymetrické kryptografie

---

04 Útoky na kryptografické systémy a protokoly

Útoky

- Offline Brute-force
- Dictionary

- Patterns - kombinace dictionary + bruteforce
- Rainbow tables - předpočítané hashe

Jak se bránit? - Silná hesla - Zvýšit cenu bruteforce: - PBKDF2 - Scrypt: memory hard funkce - Argon2: memory hard hash funkce - Bcrypt: přidává implicitně salt k hashi

Replay útok

- Útočník odchytí validní zprávu a jejím přeposláním přesvědčí poctivou stranu o její validitě, přestože byla použita v jiném kontextu, než pro který byla prvně vytvořena.

Reflection útok

- Útočník použije cíl k tomu, aby authentizoval vlastní challenge.
- 
  reflection attack

Side channels útoky

- Neinvazivní útoky postranními kanály jsou založené na nedokonalosti fyzické impllementace kryptografických algoritmů. Získavají informace o uložených datech ze zdrojů, které nejsou primárně určeny ke komunikaci.

Časová analýza

- Využívá faktu, že čas operace závisí na zpracovávaných datech.

Diferenční power analýza

- Využívá statickou analýzu energetické spotřeby různých operací na různých vstupech.

Oracle útoky

- Využívá padding zprávy. Možný u CBC modulu. Server leakuje informaci, jestli byl použitý padding správný, nebo ne.

Microarchitectural útoky - Meltdown, Spectre

Cache Timing útoky

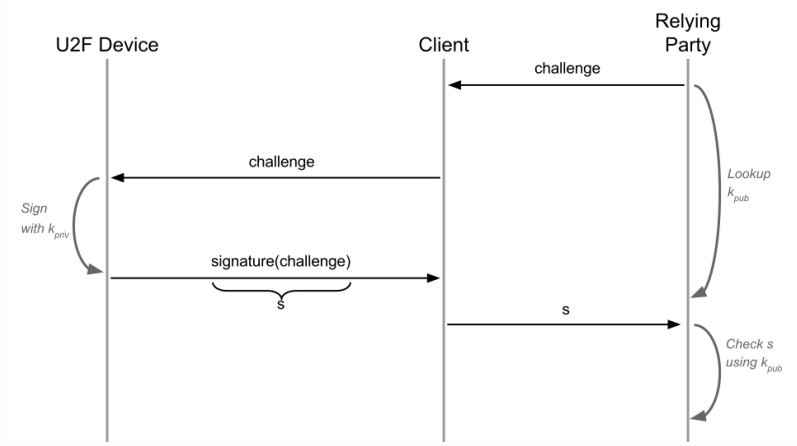
- Využívají časové rozdíly mezi přístupem k datům uloženým v mezipaměti a datům, která nejsou uložena v mezipaměti.

Meltdown a Spectre

- Využívají zranitelnosti moderních procesorů, které používají “speculative execution.”
- **Meltdown**
  - Dokáže přečíst data ze zabezpečené části paměti.
  - Exploits speculative execution - procesor řeší oprávnění instrukce až potom co je vyhodnocená jako oprávněná, mezivýsledky ale zůstávají v cachi než se přepíšou.
- **Spectre**
  - Dovoluje přečíst paměť v rámci sdílené virtuální paměti (např. jeden tab browseru přečte data druhého tabu).

HW Ochrana citlivých dat

Fido U2F tokens

- Autentizační tokeny, náhrada hesel tokenem s asymetrickým párem klíčů + challenge+response protokolem.
- 

TPM - Trusted platform module

- Kryptografické SM, které jsou vevnitř/připojené k dalšímu zařízení.
- Používá se pro bezpečné vytváření a ukládání krypto klíčů, šifrování/dešifrování disku, source of trust, že OS a firmware nejsou změněny.

Hardware Security modules (HSMs)

- Zařízení, které lze přidat do systému pro správu, generování a bezpečné ukládání kryptografických klíčů.

Intel SGX

- Rozšíření architektury Intelu k zabezpečení SW.
- Poskytuje instrukce pro vytvoření izolovaného prostředí - Enklave, která má definované “trusted functions”.

Smartcards

- Navrhnuto tak, aby bylo nejvíce bezpečné, levné a přenosné.
- Primárně využívané v telekomunikacích jako SIM a finančním sektoru - bankovní karty, ePassports.
- **Fyzická ochrana karty**
  - Tamper-evidence
  - Tamper-resistance
  - Tamper-response

Útoky na karty

- Invazivní, semi-invazivní
- Side-channel
- Logické

---

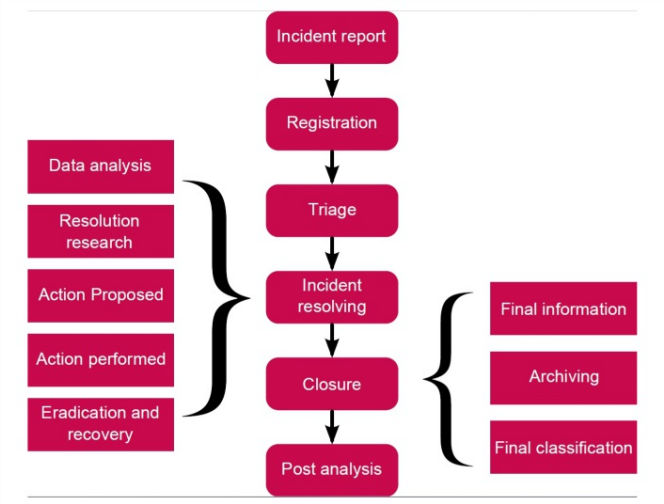
05 Řízení kybernetické bezpečnosti

Computer Security Incident Response Team (CSIRT)

- CSIRT a CERT:
  - kyberbezpečnostní týmy, které řeší bezpečnostní incidenty vzniklé v počítačových

sítích, koordinují jejich řešení a snaží se jim předcházet.
    - CERT je chráněno ochrannou známkou.
    - Každý CSIRT má vytyčeny cíle činnosti, operační hodiny, pole působnosti (síťový rozsah, doména, firma), kontaktní údaje a pod koho spadá
    - V Česku jsou dva týmy na národní úrovni:
        - Vládní CERT jako součást NÚKIB, který dohlíží na kritickou a významnou infrastrukturu
        - Národní CSIRT.CZ, který je provozován sdružením CZ.NIC dle veřejnoprávní smlouvy a Zákona o kybernetické bezpečnosti. Dohlíží nad méně významnou infrastrukturou.
    - CSIRT má koordinační roli, ne represivní.

Řešení incidentů

- Podle toho, jestli je tým interní, nebo koordinační.
- 

  Fáze řešení incidentu

1. **Příjem hlášení**
2. **Registrace hlášení**
3. **Třízení události (Triage)**
    - Ověření události
    - Kompetentnost
    - Kategorizace
    - Priorita
    - Přiřazení incidentu řešiteli
4. **Řešení incidentu**
5. **Uzavření incidentu**
6. **Post analýza**

Upozornění a varování

- Informace o novém, probíhajícím nebo nedávném bezpečnostním útoku, chybě nebo zranitelnosti.

Penetrační testování

- Provedení testu s cílem identifikovat zranitelnosti, které by mohly být přítomny v aktivu.

Základní typy penetračních testů

- **Pozice testera**
    - Externí

    - Interní
- **Množství informací o testovaném prostředí**
    - Black-box
    - White-box
    - Gray-box

Životní cyklus penetračního testování:

1. Plánování a průzkum
2. Skenování
3. Enumerace
4. Zisk přístupu
5. Závěrečná zpráva

Honeypots

- Lákadlo pro potenciální záškodníky, které simuluje reálný počítačový systém.
- Nízko interakční vs. Vysoko interakční
- Výzkumné vs. produkční

Monitorování bezpečnosti sítě

- Pasivní monitoring
    - Wireshark, ntopng
- Aktivní monitoring
    - Generování provozu navíc, jehož účelem je získat informace o zařízeních připojených v síti
    - Ping, traceroute

Analýza paketů

- Deep Packet Inspection
- Vysoké nároky na zdroje

Analýza síťových toků

- Síťový tok: seskupení paketů, které mají stejnou základní pětici vlastností
    - zdrojovou a cílovou IP
    - zdrojový a cílový port
    - typ protokolu
- Netflow/IPFIX

Digitální forenzní vyšetřování

- Digitální forenzní analýza je proces použití vědecky zdůvodněných a ověřených metod zkoumání digitálních stop pro rozhodování státních orgánů (např. policie) a jiných právních subjektů pro účely právních úkonů.
- **Digitální stopa:**
    - Forma důkazu o činnosti uživatele v digitálních světě.
        - Logy, netflow, etc.
    - Vlastnosti:
        - Identifikace času
        - Informační hodnota
        - Životnost
- Zdroje digitálních stop:
    - Integrované
        - permanentní - HDD, SSD
        - volatilní - RAM
    - Externí
        - Děrná páska, magnetická páska, CD/DVD, SD karty

    - Vzdálené
        - NAS, cloud, …
- Při zajišťování postupovat od zdrojů, které mají časově omezenou životnost.
- Nutné vytvořit kopii (bitovou nebo logickou).

---

06 Právo ICT

Základní pojmy

- **Kyberkriminalita:**
    - **Cyber-enabled** - klasická kriminalita páchaná prostřednictvím ICT
        - šíření protiprávního obsahu, porušení autorského práva, …
    - **Cyber-dependent** - kriminální chování, jehož cílem je ICT
    - **Cyber-supported** - kriminalita, při níž došlo využití ICT
        - Např. někomu vyhrožuju v reálném životě a přes SMSky
- **Kyberbezpečnost** - Ochrana infrastruktury
- **Kyberobrana** - Obrana kybernetické suverenity státu, armáda a bezpečnostní složky
- **Kybernetický prostor** - digitální prostřední umožňující zpracování informací tvořené systémy a službami a sítěmi el. komunikací
- **Kritická informační infrastruktura** - prvek nebo systém prvků kritické infrastruktury a informační systémy v oblasti kybernetické bezpečnosti

Směrnice o eletronickém obchodu

- ISP jsou odpovědni za obsah pouze pokud:
    - Mere conduit - prostý přenos - sám ho iniciuje nebo změní obsah
    - Caching - podobně jako u Mere conduit
    - Hosting - pokud-li:
        - Mohl vědět, že obsah informací je protiprávní
        - Dozvěděl-li se prokazatelně o protiprávním obsahu a neučinil protiakci
- ISP není povinno
    - Dohlížet na obsah přenášených informací
    - Aktivně vyhledávat skutečnosti poukazující na protiprávnost
- ISP mají povinnost součinnosti při trestním vyšetřování
- Data retention: po dobu 6 měsíců ukládat provozní a lokalizační údaje

Internetová jurisdikce

- Snaha o regulaci společenských vztahů vznikající v souvislosti s kontaktem lidské společnosti a internetu
- ICT systémy nejsou teritoriálně vymezené
    - Určení jurisdikce stanovují mezinárodní úmluvy, nebo právo příslušného státu

Ochrana autorských práv k softwaru

- Autorská práva je souborem automatických oprávnění, které náleží tvůrcům autorských děl:
    - Práva osobností: právo rozhodnout o zveřejnění díla, právo osobovat si autorství a právo na nedotknutelnost díla
    - Práva majetková: právo dílo užít (rozmnožovat, rozšiřovat, půjčovat, pronajímat, …)
- Nositel autorských práv je vždy pouze **fyzická osoba**, která dílo vytvořila a těchto práv se nedá vzdát
- Autorská práva k software se např. nevztahují na pouhé myšlenky a principy

Patentová ochrana softwaru

- Narozdíl od autorské ochranny, není automatická - patent musí být registrován a udržován
- Patentován není program, ale řešení

Licencování softwaru

- Licence je právní nástroj, definován v rámci autorského zákona, který umožňuje používat nebo redistribuovat software. Je definována v rámci licenční smlouvy.
- **Proprietární**
  - EULA
- **Open source**
  - BSD, MIT, GPL

Ochrana osobních údajů

- Osobní údaje jsou jakékoliv informace, které by mohli vést k identifikaci subjektu.

GDPR

- Nařízení EU, které ukládá povinnosti firmám a definuje práva občanů v oblasti ochrany osobních údajů.
  - Data subject
  - Správce osobních údaje - odpovědný za zpracování osobních údajů
  - Zpracovatel osobních údajů - zpracovává osobní údaje jménem správce
- Základní principy zpracování osobních údajů:
  - Transparentnost
  - Limitace účelem
  - Minimalizace
  - Přesnost
  - Integrita a důvěrnost
- Práva uživatelů
  - Právo být informován
  - Právo být zapomenut
  - Právo opravy
  - Právo přístupu

Zákon o kybernetické bezpečnosti

- Cílem je zvýšit bezpečnost kybernetického prostoru a kritické infrastruktury
- specifikuje bezpečnostní opatření a ukládá povinnost a rozsah bezpečnostních opatření správcům systémů kritické infrastruktury a významných informačních systémů
  - Organizační opatření: řízení rizik, bezp. politika, …
  - Technické opatření: fyzická bezpečnost, deployment nástrojů, …
- Specifikuje kybernetickou bezpečnostní událost a incident
- Specifikuje varování
- Specifikuje opatření
  - Reaktivní a ochranná
- Národní CERT
  - Pod CZ.NIC
  - Bez nařizovacích a sankčních pravomocí
  - Stará se především o poskytovatele služeb elektronických komunikací a správce významných sítí
- Vládní CERT
  - NÚKIB
  - stará se o ochranu kritické informační infrastruktury a významných informačních systémů

Budapešťská úmluva - úmluva o počítačové kriminalitě

- V platnosti od 1.7.2004, 65 států k 2020
  - Indie, Brazílie, Rusko ji odmítají
- První mezinárodní smlouva, která se snaží řešit internetovou a počítačovou kriminalitu
- Zaměřuje se na zlepšení vyšetřovacích technik a zvýšení spolupráce mezi národy
- Sjednocuje defici kyberzločinů ale nepokrývají zdaleka všechno