**BI-SPOL-8 Infrastruktura veřejného klíče, distribuce klíčů, digitální podpis. Certifikáty, certifikační autority. Kryptograficky bezpečné generátory náhodných čísel**

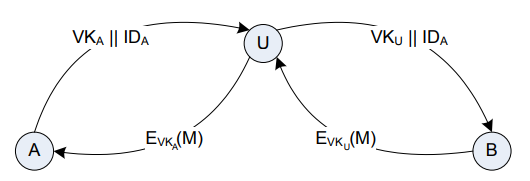
BI-BEZ

### Infrastruktura veřejných klíčů (Public Key Infrastructure)

* systém digitálních certifikátů, certifikačních autorit (CA) a dalších registračních úřadů, které slouží k ověřování platnosti každé strany zúčastněné v určité elektronické transakci.
* jedná o specifikaci technických a organizačních opatření pro vydávání, správu, používání a odvolávání klíčů a certifikátů
* hlavní cíl: zabezpečení kompatibility SW pro Internet

## Distribuce klíčů veřejných klíčů

* nutné pro komunikaci subjektů
* Musí být zajištěno, že klíč k příjemci dorazí v nepozměněné podobě
* Hrozí podvržení útočníkem prostřednictvím útoku man-in-the-middle. Je to útok. kde komunikace probíhá přes prostředníka, který zaměňuje klíče za své a tím může číst obsah zpráv a subjekty to nezjistí. Existuje více technik, jak tento problém řešit.



**Způsob podvržení VK (man-in-the-middle)**

* Subjekt A pošle svůj a svůj identifikátor , tj. zprávu ||, subjektu B
  + Předpoklad: útočník U má aktivní přístup k veřejnému kanálu
* U zachytí zprávu ||, vytvoří novou zprávu || a odešle ji B – podvržení subjektu B
* B se domnívá, že , který dostal, patří A → B zašifruje každou zprávu M klíčem , tj. vytvoří (M), odešle ji A
* U zachytí (M), dešifruje ji a získává zprávu M
* U zná – zašifruje M pomocí tohoto klíče a pošle ji A
* Důsledek: U čte korespondenci zaslanou subjektem B subjektu A. U také může tyto zprávy měnit

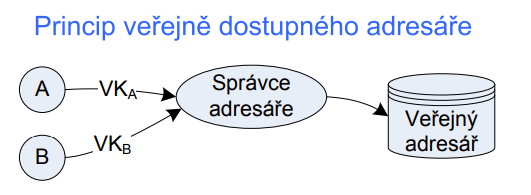
### Techniky distribuce VK

#### Zveřejnění VK

* Zasíláním VK individuálně nebo hromadně v rámci nějaké komunity – přes email, chat
* Nebo dám VK na web
* Rychlý a jednoduchý způsob
* Nízká bezpečnost! Není odolný proti podvržení VK
  + Nepovolaný subjekt může číst zprávy dotčeného subjektu až do odhalení

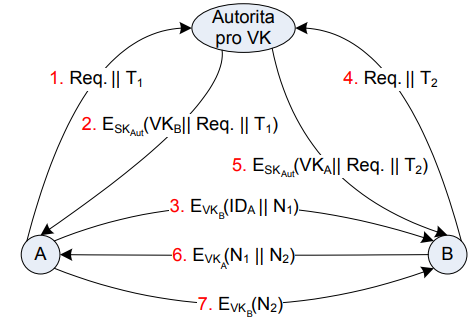
#### Veřejně dostupný adresář

* Vyšší stupeň bezpečnosti
* **Distribuci VK zabezpečuje důvěryhodná autorita** – zodpovídá za obsah, je správcem adresáře
* Účastníci registrují svůj VK prostřednictvím autorizovaného správce adresáře
* Bezpečná registrace: osobně nebo přes bezpečnou komunikaci
* Položky v adresáři jsou tvořeny: [jméno;VK]
* Účastníci mění VK podle potřeby (bezpečnost, velký objem dat jedním VK, zkompromitovaný VK apod.)
* Správce periodicky aktualizuje adresář – elektronicky nebo fyzicky
* Tento systém je bezpečnější než předchozí, má ale slabá místa
  + Pokud nepovolaný subjekt získá SK správce adresáře, může modifikovat adresář a provádět odposlech jako v předchozí metodě



#### Autorita pro VK

* Přísnější dohled na distribuci VK z adresáře znamená vyšší stupeň bezpečnosti
* autorita vykonává činnost správce adresáře VK
* každý účastník zná VK autority , která vlastní odpovídající soukromý klíč



* A chce inicializovat výměnu zprávy s B – **scénář distribuce VK**:

Obsah obrázku text

Popis byl vytvořen automaticky

* Postup v bodech 6 a 7 zvyšuje bezpečnost výměny zpráv
* B zasláním zprávy || subjektu A, kde N2 je náhodné číslo generované B, dá jistotu A, že autorem zprávy je B, protože obsahuje N1 vygenerované A. Dešifrování na obou stranách je možné jen se znalostí příslušných SK A a B
* V kroku 7 získá B jistotu, že zpráva pochází od A, protože obsahuje N2
* Tato metoda má své nevýhody
  + Autorita představuje „úzké hrdlo“ této koncepce
  + Každý účastník na získání VK adresáta musí nejdříve komunikovat s Autoritou pro VK
    - Alternativní přístup v distribuci VK představují **certifikáty**

#### Certifikace VK

* distribuce VK bez kontaktu s třetím důvěryhodným subjektem
* přístup vyžaduje certifikát a certifikační autoritu

**Certifikát**

* struktura obsahující:
  + VK žadatele/držitele certifikátu
  + Identifikační údaje držitele certifikátu
  + Dobu platnosti certifikátu
  + Další údaje vytvořené certifikační autoritou
* Tato struktura je podepsána soukromým klíčem certifikační autority a každý účastník může verifikovat obsah certifikátu pomocí veřejného klíče certifikační autority

**Certifikační autority**

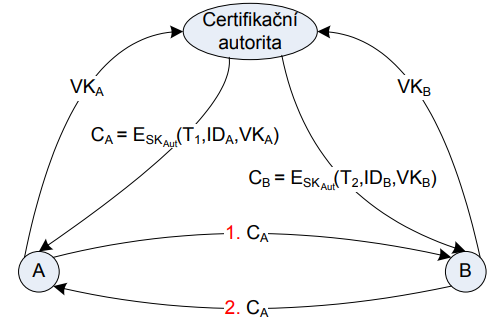
* Certifikační autorita (CA) je důvěryhodná třetí strana, která na základě žádosti vydává a aktualizuje certifikáty
* Každý účastník může verifikovat to, že certifikát byl vytvořen CA, pomocí jejího veřejného klíče
* Žádost o vydání certifikátu lze CA doručit osobně nebo elektronicky s využitím bezpečné komunikace
* Přijímání žádostí, kontrola údajů v žádosti a odevzdávání certifikátů žadatelům se realizuje registrační autoritou

**Princip výměny VK na bázi certifikátů**

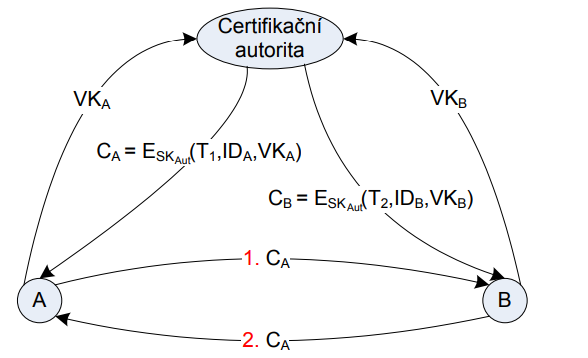
* Subjekt A před započetím jakékoliv komunikace požádá CA o vydání certifikátu na svůj

Obsah obrázku text

Popis byl vytvořen automaticky



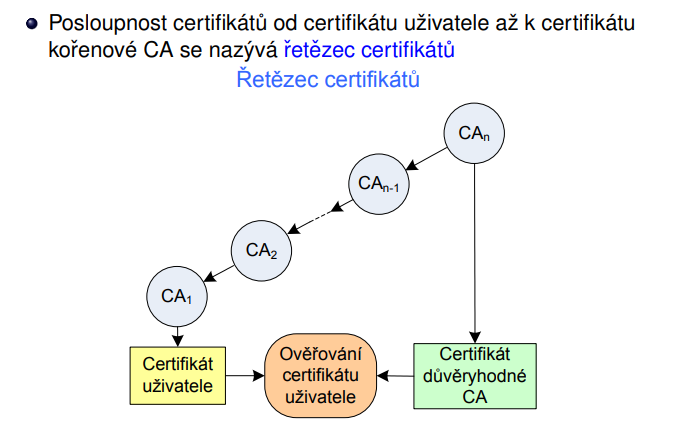
* Dešifrováním se získá jméno, VK držitele certifikátu a časový údaj o platnosti
* Analogicky požádá o vydání certifikátu subjekt B
* Certifikát má analogickou strukturu jako
* Distribuce VK potom představuje výměnu certifikátů a mezi subjekty A a B

****

### Certifikační strom

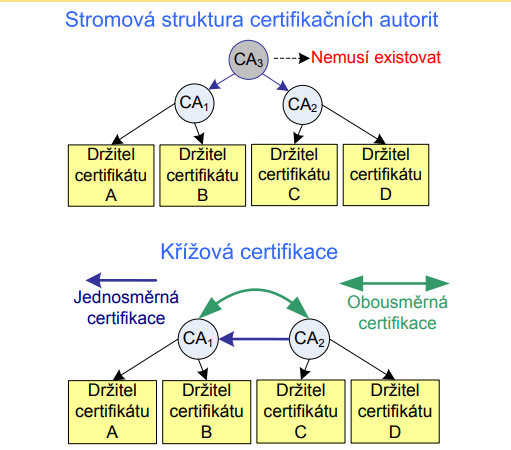
Obsah obrázku text

Popis byl vytvořen automaticky



Obsah obrázku text

Popis byl vytvořen automaticky



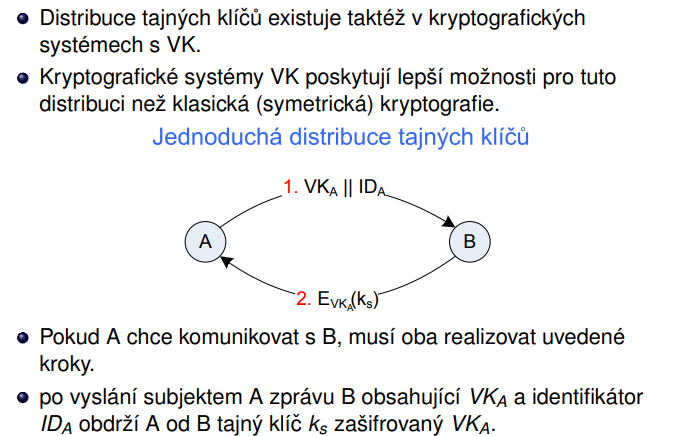
Obsah obrázku text, osoba, snímek obrazovky, dokument

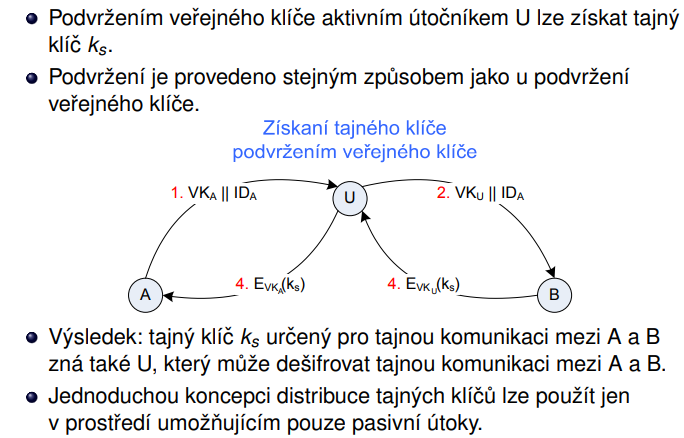
Popis byl vytvořen automaticky

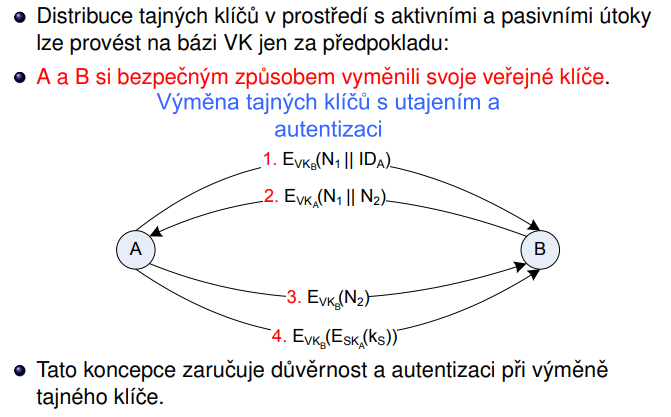
Obsah obrázku text

Popis byl vytvořen automaticky

## Distribuce tajných klíčů

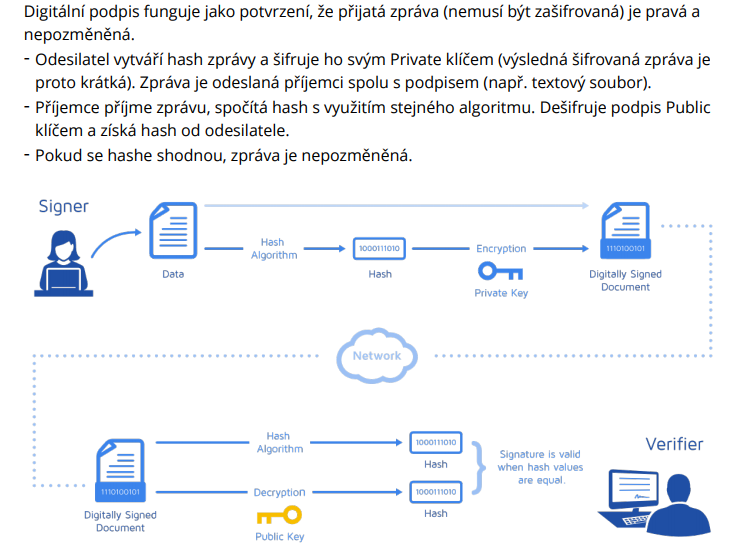






* **Podmínka pro autentizaci je, že si bezpečně vyměnili VK!!**

### Digitální podpis



* SK – podepisuje
* VK – ověřuje

**Vlastnosti**

* **Nezfalšovatelnost, autentizace** – podpis se nedá napodobit jiným subjektem než podepisujícím
* **Integrita** – podepsaná zpráva se nedá změnit, aniž by se zneplatnil podpis
* **Nepopiratelnost** – podepisující nesmí mít později možnost popřít, že dokument podepsal

**Kategorie digitálních podpisů**

* *Přímé digitální podpisy*
  + Mezi dvěma subjekty, příjemce zná VK odesílatele
  + Problém s popiratelností → pokud odesílatel popře podepsání zprávy, příjemce ho nemůže usvědčit (není nikdo třetí, kdo by svědčil proti odesílateli)
* *Verifikované digitální podpisy*
  + Využívá důvěryhodnou třetí stranu (arbitra), který ověřuje podpisy všech zpráv

### Kryptograficky bezpečné generátory náhodných čísel

**Náhodné číslo** – číslo vygenerované procesem, který má nepřevídatelný výsledek a jehož průběh nelze přesně reprodukovat. Tomuto procesu říkáme *generátor náhodných čísel*.

U jednoho čísla nelze dobře diskutovat, zda je nebo není generováno generátorem náhodných čísel. Pracujeme tedy vždy s *posloupností náhodných čísel* neboli *náhodnou posloupností*

V počítači se čísla reprezentují pomocí bitů. Namísto náhodných čísel tedy pracujeme s *náhodnými bity* resp. *generátory náhodných bitů*

Od náhodných posloupností jsou očekávány dobré statistické vlastnosti:

* *rovnoměrné rozdělení* – všechny hodnoty jsou generovány se stejnou pravděpodobností
* generované hodnoty jsou na sobě *nezávislé* (není mezi nimi korelace)

**Entropie** – veličina popisující míru náhodnosti – jak obtížné je hodnotu (náhodné číslo, náhodná posloupnost) uhodnout

* vždy se vztahuje k útočníkovi a jeho schopnosti předpovědět vygenerovanou hodnotu. Pokud útočník následující generovanou hodnotu s jistotou zná, entropie je nulová (a nulová je i bezpečnost aplikace, která tento generátor využívá)
* entropie generátoru je maximální, pokud se pro danou délku (počet bitů) generují všechny možné posloupnosti, každá z nich se stejnou pravděpodobností

#### Generátor pseudonáhodných čísel (PRNG)

* výstup není ve skutečnosti náhodný, ale zdá se jím být, pokud útočníkovi nejsou známy některé parametry generátoru
* rychlé, snadno realizovatelné, dobré statistické vlastnosti
* aby byl kryptograficky bezpečný, musí splňovat tyto vlastnosti:

1. **next-bit test**: pokud je známo prvních k-bitů náhodné posloupnosti, neexistuje žádný algoritmus s polynomiální složitostí, který by dokázal předpovědět (k+1). bit s větší jak 50% šancí
2. **state compromise**: i když je zjištěn vnitřní stav generátoru (ať už celý nebo zčásti), nelze zpětně rekonstruovat dosavadní vygenerovanou posloupnost. Navíc pokud do generátoru za běhu vstoupí další entropie, nemělo by být možné ze znalosti vnitřního stavu předpovědět vnitřní stav v následujících iteracích

**Příklady, jak realizovat kryptograficky bezpečné pseudonáhodné generátory**

Obsah obrázku text

Popis byl vytvořen automaticky

**Blum-Blum-Shub pseudonáhodný generátor**

**Obsah obrázku text

Popis byl vytvořen automaticky**

* zmíněné pseudonáhodné generátory (PRNG) vyžadují náhodný a tajný vstup, seed:
  + bez nějaké skutečné náhody se stejně neobejdeme
  + kvalita PRNG se odvíjí i od kvality hodnoty seed
* Entropie výstupu PRNG: dána entropií, která vstupuje (seed), **algoritmus samotný nikdy nemůže entropii zvyšovat**

#### Skutečně náhodné generátory (TRNG)

* Využívají zdroj entropie, kterým je zpravidla nějaký fyzikální jev nebo vnější vliv, například:
  + radioaktivní rozpad
  + atmosférický šum
  + teplotní šum
  + chování uživatele (pohyb myší, prodlevy při psaní na klávesnici)

**Vlastnosti**

* výstup není předvídatelný, i když známe všechny parametry
* výstup má zpravidla horší statistické vlastnosti – je nutné následné zpracování
* složitější implementace – často je potřeba dodatečná HW
* zdroj entropie je třeba průběžně testovat, časem se mohou zhoršit jeho vlastnosti
* následné zpracování (post-processing) má za cíl vylepšit statistické vlastnosti TRNG, zejména:
  + odstranění nevyváženosti jedniček a nul (bias) a zajištění rovnoměrného rozdělení
  + extrakce entropie – zvýšení entropie výstupních bitů za cenu snížení rychlosti jejich generování (bitrate)
* musí se následně zpracovat, aby se zlepšily statistické vlastnosti (např. odstranění 0/1, nebo se XORuje s výstupem pseudgenerátoru)

**post-processing**

Obsah obrázku stůl

Popis byl vytvořen automaticky

**Testování náhodných generátorů**

**Obsah obrázku text

Popis byl vytvořen automaticky**

**Obsah obrázku text

Popis byl vytvořen automaticky**

**Obsah obrázku text

Popis byl vytvořen automaticky**