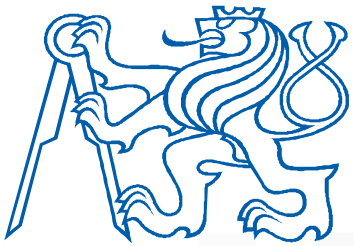
České vysoké učení technické v Praze
Fakulta elektrotechnická
Katedra telekomunikační techniky

A7B32KE – přednáška 9

Autentizace, autentizační protokoly

Ing. Tomáš Vaněk, Ph.D.
tomas.vanek@fel.cvut.cz



Obsah

- Autentizace programů
- Autentizace uživatelů
  - Biometrika
  - Tokeny
  - Hesla
- Autentizační protokoly
  - PAP,CHAP, EAP
  - LANMAN, NTLM
  - TACACS
  - Radius
  - Diameter
  - Kerberos
  - TESLA, DREAM, BiBa,

Autentizace

- Autentizace (proces identifikace nebo verifikace) uživatele/entity je možné provádět podle toho :
  - **kdo jsou** – identifikace podle globálně jednoznačných parametrů (biometrika – otisky prstů, dlaní, sítnice, DNA apod.)
  - **co mají** – identifikace podle vlastnictví určitých předmětů (klíč, magnetická/čipová karta, USB token apod.)
  - **co znají –** identifikace pomocí přístupových hesel, číselných kombinací, osobních identifikačních čísel – PIN apod.

- Výsledek autentizace: povolení / nepovolení přístupu ke zdrojům systému

Biometrické metody autentizace

Definice: „…automatizované metody identifikace nebo verifikace osoby na základě měřitelných fyziologických nebo behaviorálních vlastností člověka.“

- identifikace lidí na základě jejich osobních charakteristik
  - jedinečné
  - minimálně proměnné
  - nelze je zapomenout, ztratit, odcizit (většinou)
- biometriky se liší různou mírou spolehlivosti, ceny a společenské přijatelnosti
- biometrická data nejsou nikdy 100% shodná
- musíme povolit určitou variabilitu mezi registračním vzorkem a později získanými biometrickými daty

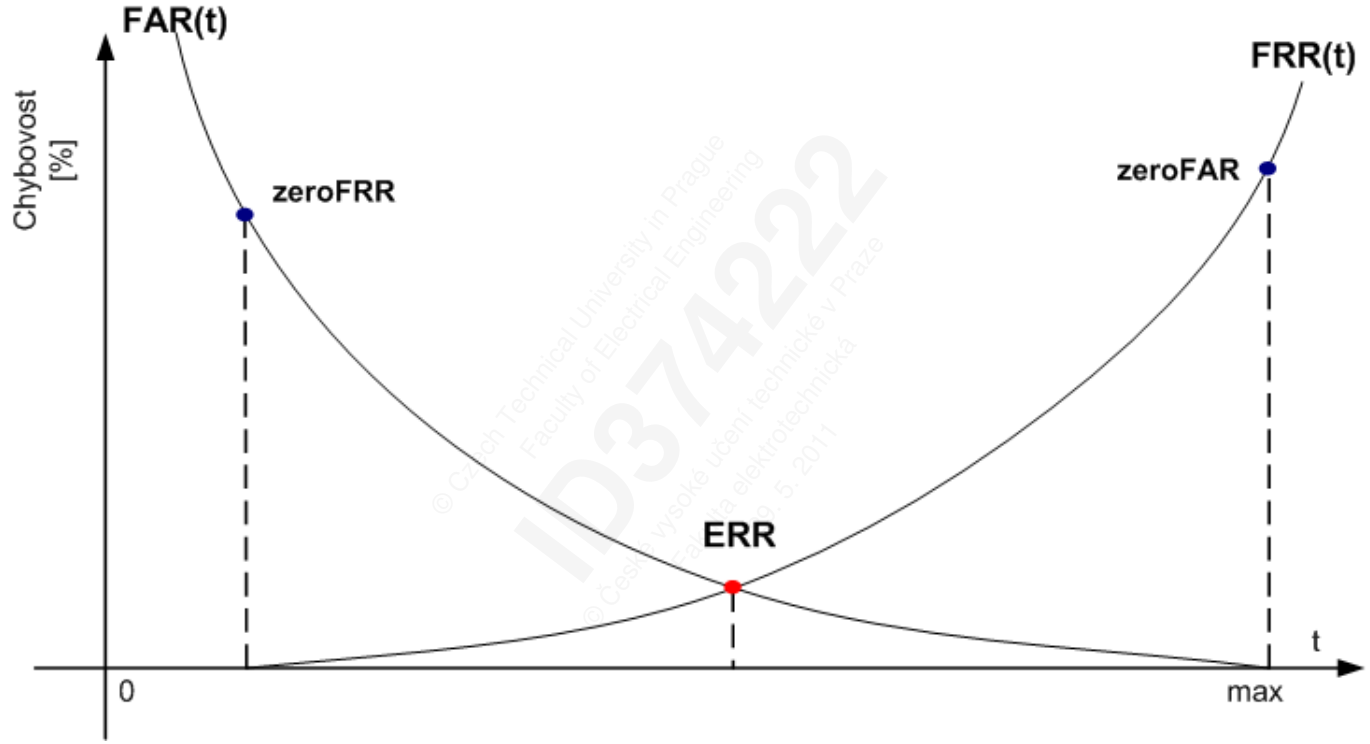
Biometrické metody autentizace

Kvalitu biometrik lze charakterizovat:

- četností nesprávných odmítnutí autorizovaného subjektu
  FRR (False Rejection Ratio)

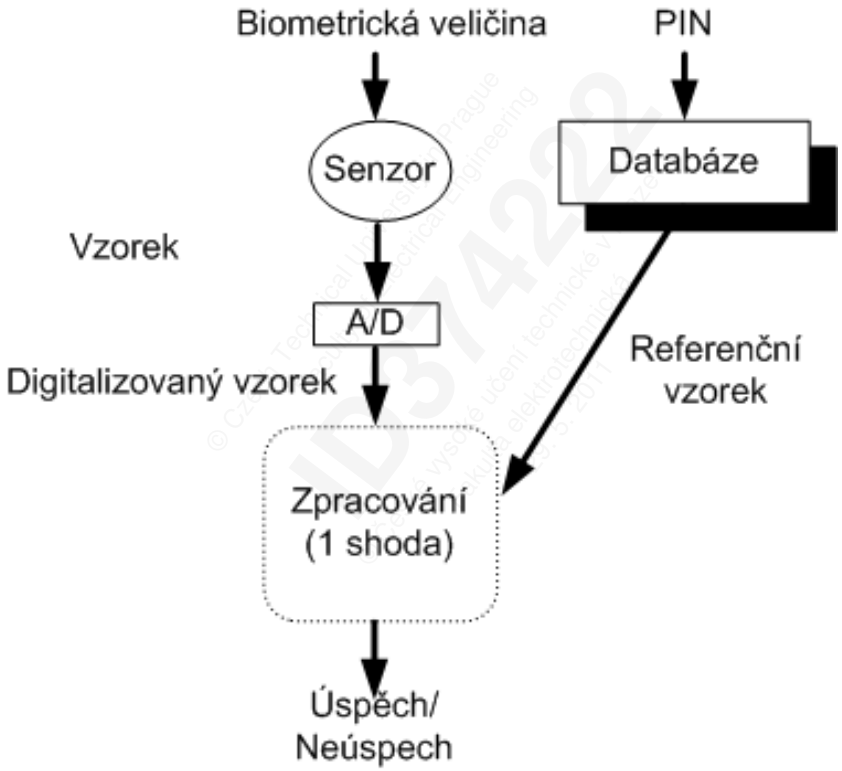
- četností nesprávných přijetí útočníka
  FAR (False Acceptance Ratio)

Biometrické metody autentizace - parametry



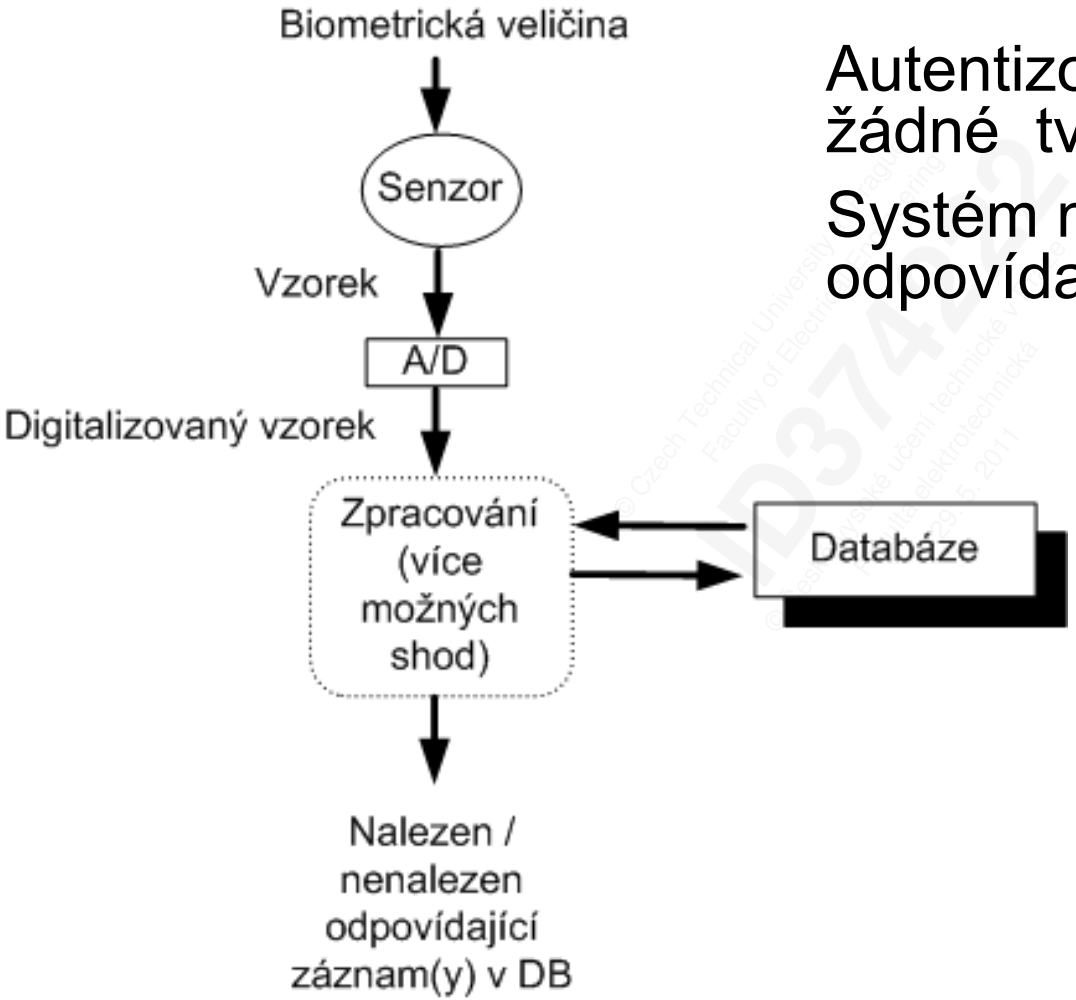
FAR – False Acceptance Ratio
FRR – False Rejection Ratio
EER – Equal Error Rate (CER … Crossover ER)
FTER – Fail To Enroll Rate …hendikepovaní uživatelé
FTAR – Fail to Acquire Rate … hendikepovaní uživatelé

Autentizace pomocí verifikace



Autentizovaná entita předkládá tvrzení o své identitě, které je následně ověřena (1:1)

Autentizace pomocí identifikace



Autentizovaná entita nepředkládá žádné tvrzení o své identitě.

Systém musí sám v DB nalézt odpovídající záznam(y) – 1:n

Klasifikace biometrických metod

Biometrické technologie jsou založeny na

- statických charakteristikách = anatomické charakteristiky jedince
  - otisk prstu
  - otisk sítnice
  - tvar duhovky
  - geometrie ruky

- dynamických charakteristikách = behaviorální charakteristiky vyžadují interakci účastníka
  - dynamika podpisu
  - analýza hlasu
  - analýza chůze
  - …

Otisky prstů

- systém provádí statistický rozbor tzv. markantů - hrbolků, smyček a spirál v otisku prstu a jejich vzájemné polohy
- často se provádí testování uživatelem zvoleného výběru několika prstů

Výhody: velmi vysoká mezilidská variabilita
dobrá zpracovatelnost vstupních dat

Nevýhody: negativní asociace uživatelů
náchylnost snímačů na ušpinění

Snímače: kapacitní
ultrazvukové
optické
termické

FAR < 0,2 % FRR ~ 5%

Geometrie ruky / prstů

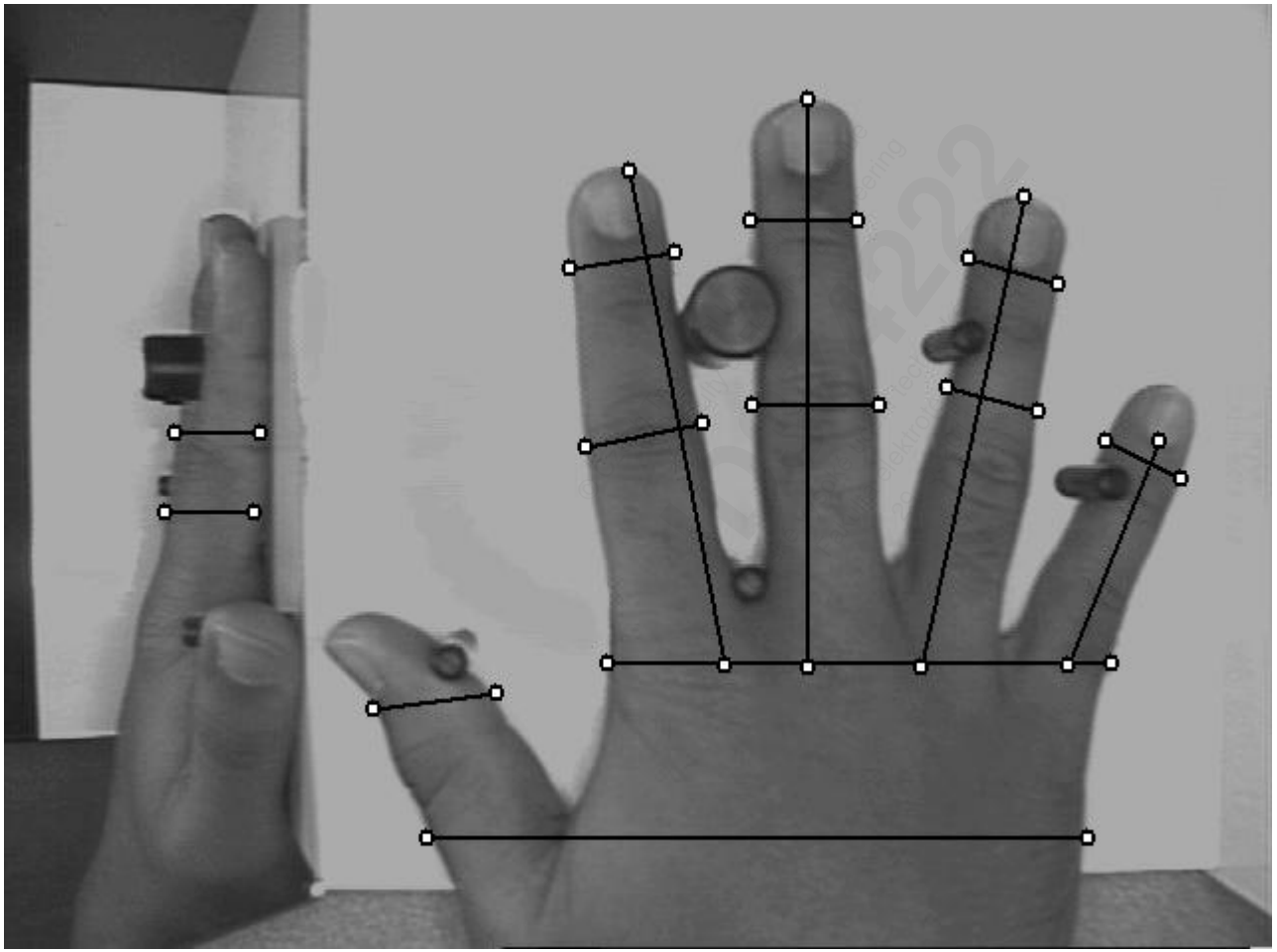
- zkoumá délku a šířku dlaně a jednotlivých prstů, boční profil ruky, rozložení cév, 3D profil ruky apod.
- výsledkem je velmi malý vzorek - cca 18 bytů

Výhody: rychlost

Nevýhody: cena
přesnost (malé rozdíly v populaci)
proměnné v čase (změnou váhy, v dospívání)

FAR ~ 10%
FRR ~ 10%

Geometrie ruky

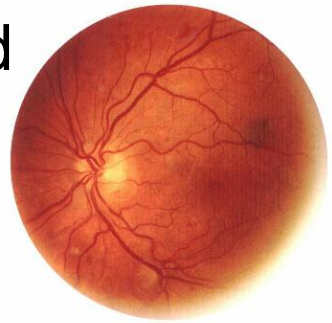


Obrazy sítnice (retinal scan)

- IR skener pořídí obraz struktury sítnice v okolí slepé skvrny, tento obraz je digitalizován a převeden na vzorek délky cca 80 B
- otisky sítnice mají stejné charakterizační vlastnosti jako otisky prstů
- FBI CIA, NASA

Výhody: spolehlivost
obtížná napodobitelnost

Nevýhody: subjektivní nepříjemnost
nižší přesnost u některých očních vad
cena



FAR < 0,001%
FRR ~ 1%


1
2
3
YES
4
5
6
NO
7
8
9
FIN
CLR
0
ENT
CAN
eyedentify inc.

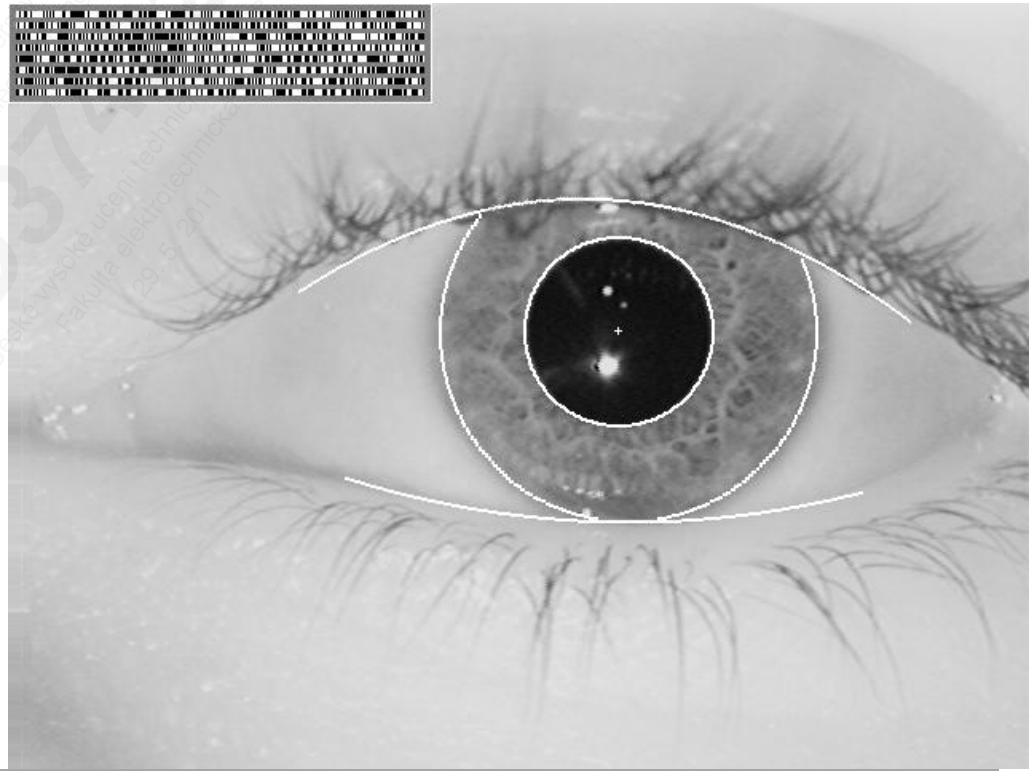
Obraz duhovky (iris scan)

- Srovnává se jedinečný vzor oční duhovky
- ČB kamera ze vzdálenosti 20-30 cm
- 256B popisujících vzor duhovky

Výhody: rychlost
přesnost

Nevýhody:
zatím funguje pouze na krátkou vzdálenost

FAR <<0,1%
FRR cca 3 %



Obraz duhovky (iris scan)

- používá se na letištích ve Frankfurtu, Heathrow –
- urychlení kontroly pro lidi, kteří často cestují

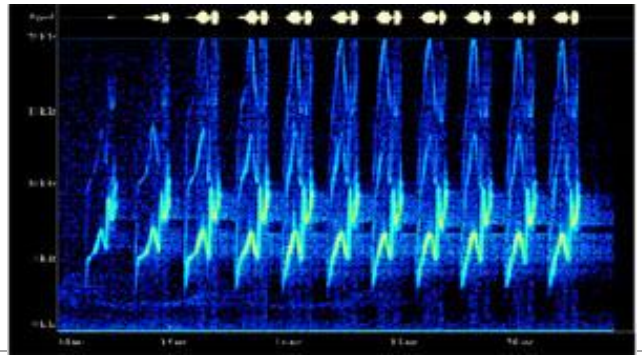
Ověřování hlasu

- testovaný subjekt přečte systémem náhodně zvolenou frázi
- sejmutá zvuková stopa je kmitočtově omezena a je proveden rozbor zvuku na základě původu jednotlivých složek zvuku v činnosti hlasového aparátu
- výsledek je vhodným způsobem komprimován na vzorek velikosti 1-2 kB a porovnán se srovnávacím vzorkem

Výhody: přirozenost

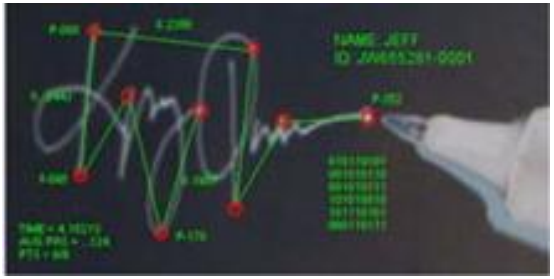
možnost provádět verifikaci prostřednictvím telefonu

FAR i FRR <2%



Dynamika psaní

- sledují se změny tlaku, zrychlení v jednotlivých částech, celkový průběh zrychlení, zarovnání jednotlivých částí podpisu, celková rychlost, celková dráha a doba pohybu pera na a nad papírem apod.
- ze získaných hodnot je vytvořen vzorek, který je srovnán se srovnávacím vzorkem



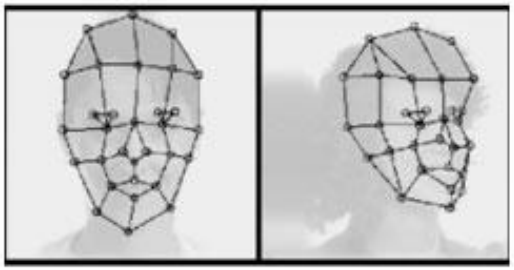
FAR i FRR >10-20%

Výhody: přirozenost
sociální akceptovatelnost

Nevýhody: malá mechanická odolnost snímačů
značná variabilita psaní u některých lidí

Další biometrické metody

- **rysy obličeje**
- analýza chůze
- Bertillonovy míry
- rytmus psaní na klávesnici
- EEG
- EKG
- otisky dlaní a chodidel
- otisky chrupu
- genetické rozbory
- pachové analýzy
- analýza zornice
- **Pozn.: ne všechny jsou vhodné k autentizaci reálném světě…**



Tokeny, smart karty, klíče….

- token je obecné označení pro předmět, který autentizuje svého vlastníka
- musí být
  - unikátní
  - nepadělatelný
- obvyklá implementace jsou magnetické nebo čipové karty (RSA SecurID)
- autentizace pomocí tokenu je často spojena znalostí hesla – dvoufaktorová autentizace

Tokeny, smart karty, klíče….

Výhody tokenů

- rychle se zjistí jejich ztráta
- nejdou kopírovat (jednoduše)

Neýhody tokenů

- bez tokenu není oprávněný uživatel přihlášen
- token se může se rozbít
- token musí být dostatečně složitý, aby se nešel kopírovat

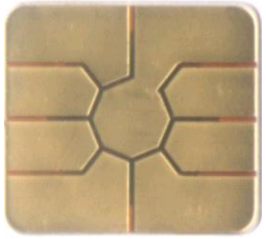
Tokeny, smart karty, klíče….

Historické tokeny:

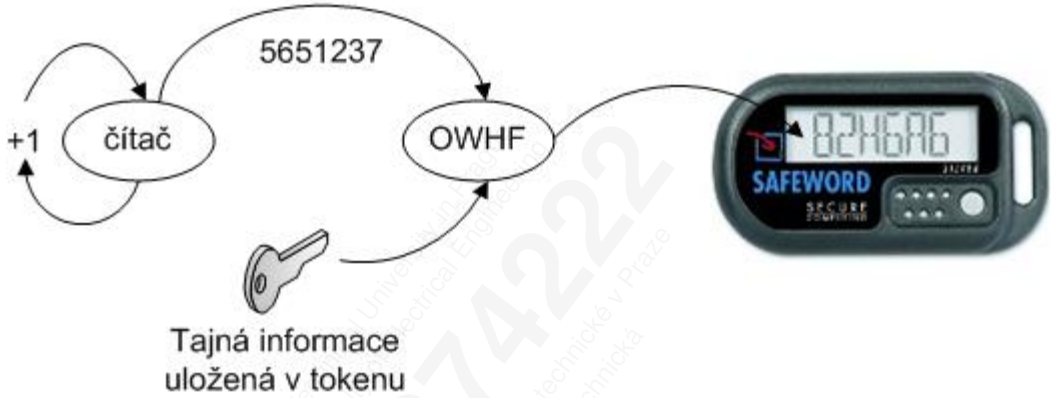
- Pečetní prsten /pečeť
- Konkrétním způsobem roztržená bankovka/pohled…
- Klíče

Dnešní tokeny:

- Karty s magnetickým proužkem (250B)
- Karty s čárovým kódem
- Čipové karty (kontaktní / bezkontaktní)



Tokeny – princip funkce



Hesla

Charakteristika dobrého hesla:

- lehce zapamatovatelné, obtížně uhodnutelné
- obsahuje malá písmena, velká písmena, číslice a jiné znaky
- dostatečná délka (min. 8 znaků)
- nejde o obvyklé slovo nebo známou frázi
- nelze jej odvodit ze znalosti osoby vlastníka
- je často obměňované
- není poznamenáno nikde v okolí místa zadávání

Heslo pro přístup do WiFi sítě v jednom hotelu v Riva del Garda: **k6jjiln6pd93e7sf9lb87cfa**

Hesla

- Člověk nemyslí ve znacích ale slovech
  - lépe se to pamatuje
  - méně překlepů
  - spojovat slova +,*,- , mezera
  - opakování 5oveček6oveček7oveček
  - nápověda („3 slepé myši“ -> TŘI krysy)

- Čísla
  - 0 , 1 ,2 – nejčastější
  - 8 nejméně časté
  - .,!? na konci hesel

- Odvození hesla z okolního kontextu
  - nálepka na NB
  - jméno systému / user name

    Příklad:

    Server „kachna“ a uživatel „root“ - r$o(o(t%kUaQcDDhTnHaQ

Passphrase

- dlouhá hesla, mohou to být části písní, básniček, citátů …
- při použití vhodného kompresního algoritmu, lze passphrase transformovat na velmi kvalitní heslo
- passphrase = dlouhé heslo složené z více slov

http://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-500.html

- dlouhodobý průzkum práci s hesly na studentech prvního ročníku (přes 100 studentů)
- 4 skupiny
  - nepoučení
  - náhodné heslo [a-zA-Z0-9])
  - passphrase
  - kontrolní skupina (8 znaků z toho min. jeden ![a-zA-Z])

Hesla

- Útoky na hesla
  - slovníkový
  - slovník + permutace (0-O, 3-E, …)
  - slovník + 0-3 číslice
  - uživatelské info – userID, jméno,
  - brute-force (do 6 znaků)

Výsledky (prolomeno hesel + brute force)

- nepoučení– 33 % a 2 hrubou silou
- kontrolní skupina – 32 % a 3 hrubou silou
- náhodné heslo – 8 % a 3 hrubou silou
- skupina s passphrase – 6 % a 3 hrubou silou

Hesla

Závěry z průzkumu:

- Náhodně vybraná hesla se obtížně pamatují
- Hesla založená na frázích jsou obtížněji uhodnutelná než naivně zvolená hesla
- Náhodná hesla nejsou lepší než ta založená na frázích
- Hesla založená na frázích se nepamatují hůře než naivně zvolená hesla
- Školení uživatelů nemá výrazný vliv na bezpečnost hesel

Autentizace mezi počítači

- netriviální problém – nelze použít biometriky ani tokeny
- autentizace podle síťových adres
  - L2 – fyzické adresy(MAC, DLCI,…)
  - L3 – logické adresy (IP,IPX,…)
  - není to bezpečné - spoofing

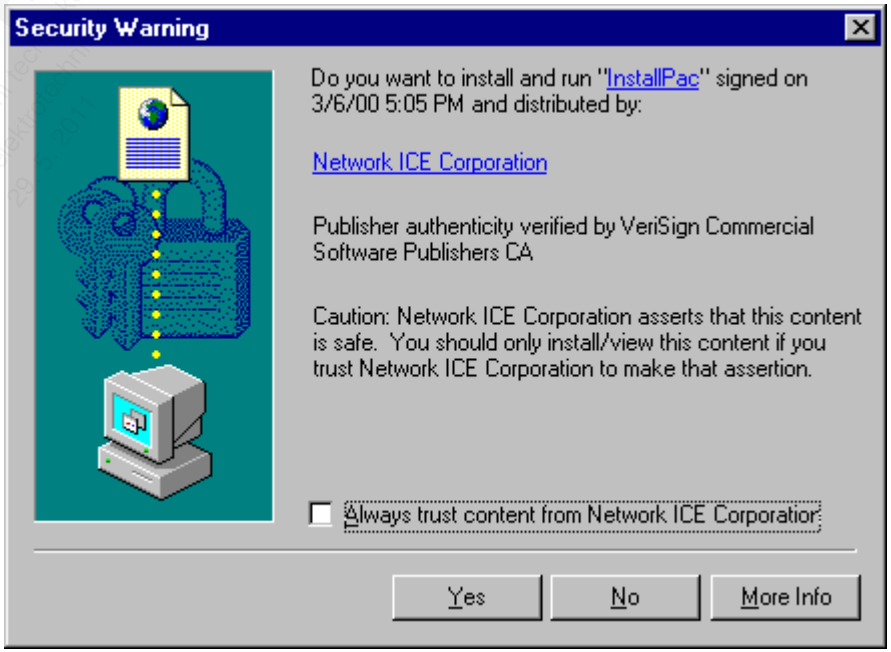
- lze použít digitální certifikáty X.509…
  viz přednáška o SSL/TLS

Autentizace programů

Cíl: zajištění integrity programu
ověření totožnosti autora programu

Spustitelný soubor je digitálně podepsán a před spuštěním je tento podpis ověřen.

Využívá certifikačních autorit.
Více informací na přednášce o SSL a elektronickém podpisu.



Protokol

Definice: Protokol je soubor syntaktických a sémantických pravidel určující výměnu informace mezi nejméně dvěma entitami spojenými například prostřednictvím počítačové sítě.

Protokol zahrnuje :

- Proceduru navázání spojení
- Adresování
- Přenos dat
- Zpracování chyb
- Řízení toku komunikace
- Přidělování prostředků

- protokol může být standardizovaný (např. RFC, IEEE, CCITT, ISO) nebo soukromý (proprietární)
- protokol je navržen k plnění určitých úkolů, které zahrnují dva nebo více účastníků
- protokol musí plnit nějaký účel

Protokol

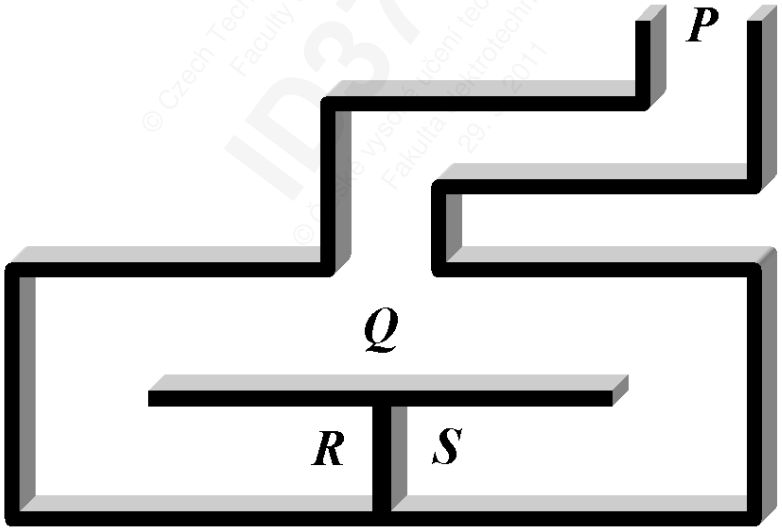
Protokoly mají ještě následující vlastnosti :

- Každý účastník protokolu musí protokol znát a postupně plnit všechny jeho kroky.
- Každý účastník protokolu musí souhlasit s postupným plněním všech jeho kroků.
- Protokol musí být jednoznačný; každý jeho krok musí být přesně definován a nesmí existovat možnost jeho nesprávného chápání.
- Protokol musí být úplný; pro každou možnou situaci musí existovat specifické řešení.

- kryptografický protokol - využívá ke své činnosti kryptografické techniky
- autentizační protokol - slouží k autentizaci
- často dochází ke spojení autentizace a jiných cílů kryptografických protokolů (utajení, integrita…)

Zero-knowledge protokoly

- protokoly s nulovým rozšířením znalostí
- umožňují s libovolnou pravděpodobností dokázat znalost (vlastnictví) nějaké informace (tajemství) aniž by došlo jakémukoliv prozrazení této tajné informace



Zero-knowledge protokoly – Fiat-Shamirův protokol

- důvěryhodná strana T (T=Trusted) zvolí modul $n = p \cdot q$ (stejně jako v RSA),n zveřejní, p a q uchová v tajnosti
- účastník A zvolí tajné číslo s takové, že $\gcd(s,n)=1$, a spočítá $v = s^2 \bmod n$. Číslo v zveřejní.

Průběh autentizace účastníka A vůči účastníkovi B:

1) A zvolí náhodné číslo r,spočte $x = r^2 mod\ n$ a výsledek odešle účastníkovi B
2) B si náhodně zvolí číslo $e = 0$ nebo 1 a odešle ho A
3) A vyřeší rovnici $a = r \cdot s^e \bmod n$
4) B ověří, že $a^2 = v^e r^2 \bmod n$

Pravděpodobnost úspěšného podvodu je 2^{-x}.

Zero-knowledge protokoly

1. pokud A zná protokol a číslo s pak B s kontrolou vždy uspěje

2. pokud A nezná s, pak zodpoví otázku správně pouze s pravděpodobností ½

Celý postup opakujeme dle potřeby x-krát.

Autentizační protokoly

Různé pohledy na autentizaci:

Autentizace – vzájemná
– jednostranná

Autentizace – jednorázová
– kontinuální

Autentizace – na vyžádání
– bez výzvy

Autentizace – přímá (ověření klienta probíhá přímo na serveru)
– nepřímá (ověření klienta pomocí externího autentizačního serveru)

Autentizační protokoly

Autentizace heslem

- účastník „A“ se autentizuje tak, že pošle účastníkovi „B“ tajnou informaci (heslo)
- nebezpečné – heslo lze odposlechnout
- např. protokol PAP

Autentizace hashem z hesla

- účastník „A“ se autentizuje tak, že pošle účastníkovi „B“ hash z hesla
- heslo neputuje sítí v OT
- nebezpečné –lze odposlechnout hash a později ho použít - tzv. replay attack (útok přehráním zprávy)

Autentizační protokoly

Autentizace pomocí mechanismu výzva-odpověď (challenge-response)

- Server vyzve klienta k autentizaci
- výzva obsahuje náhodné číslo
- klient použije hashovací funkci, tajné heslo a náhodné číslo z výzvy a výsledek odešle serveru
- server spočítá totéž a pokud se oba výsledky rovnají autentizace byla úspěšná
- např. protokol CHAP

Autentizace využívající kryptostémy veřejného klíče (včetně X.509….)

- viz. přednáška o SSL/TLS

PAP – Password Authentication Protocol

- RFC 1334
- slabá autentizace
- heslo se přenáší v OT
- autentizace je možná pouze na počátku spojení
- autentizaci iniciuje klient

CHAP – Challenge Authentication Protocol

- RFC 1994
- challenge-response (výzva-odpověď)
- neposílá heslo, ale hash z hesla a výzvy
  response=MD5(heslo,challenge)
- „výzva“ je při každém přihlášení jiná
- obě komunikující strany musejí znát tajné heslo
- autentizaci iniciuje server
- autentizace může být prováděna opakovaně, kdykoliv během spojení

Autentizační protokoly

MS-CHAPv1

- RFC 2433
- ve Windows Vista již není podporován
- jako hashovací algoritmus využívá DES (viz. slide 40)
- na straně serveru nemusí být heslo v otevřeném tvaru, ale jsou podporovány formáty NT-hash a LAN-Manager-hash
- umožňuje změnu už. hesla

MS-CHAPv2

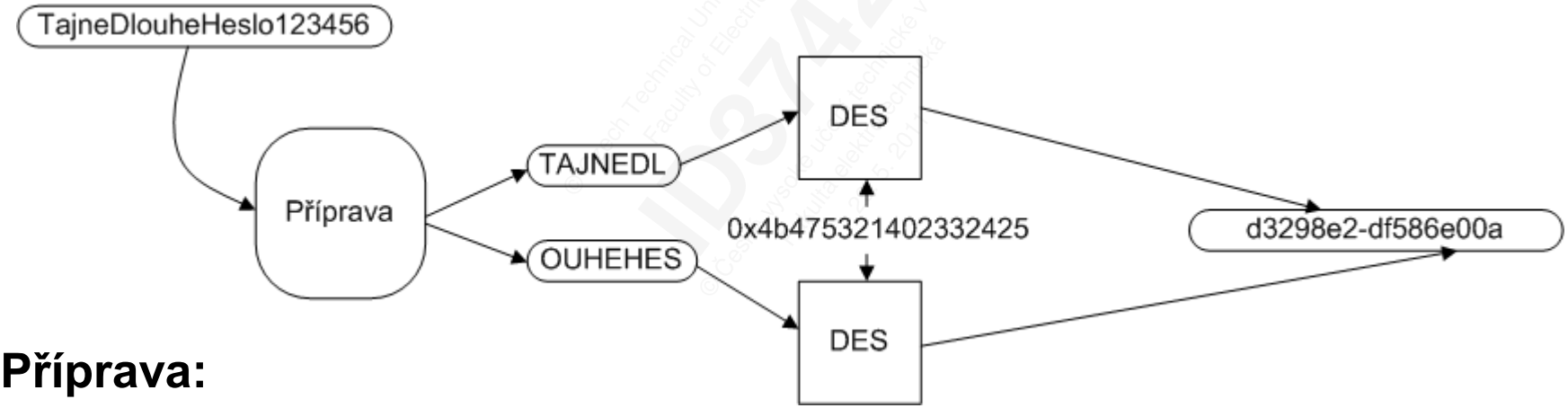
- RFC 2759
- podpora od Windows 2000
- umožňuje vzájemnou autentizaci komunikujících stran
- využívá SHA-1

EAP – Extensible Authentication Protocol

- obecný rámec pro autentizaci
- sám o sobě autentizaci nedělá
- řada variant (EAP-TLS, EAP-MD5, EAP-FAST…)
- používán v IEEE 802.1x
- více v přednášce o zabezpečení bezdrátových sítí

Autentizační protokoly - LANMAN

- používal se ve Win95, Win98, WinME
- z důvodu kompatibility přítomen i ve WinXP
- nebezpečný



Příprava:

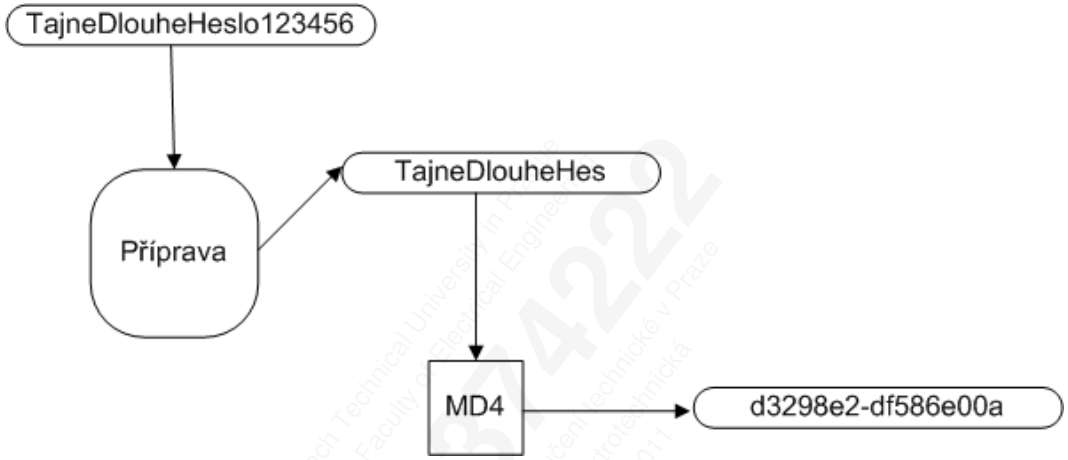
- zkrácení dlouhých hesla na 14 znaků; kratší hesla jsou prodloužena 0 na 14 znaků
- převod na velká písmena; rozdělení na dvě poloviny -> klíč pro DES
- šifruje se konstanta; oba výstupy – LANMAN hash

Autentizační protokoly - NTLM (NT LanManager)

- autentizační protokol, používaný převážně implementacemi síťových protokolů MS Windows a protokolem SMB za účelem ověření uživatele nebo spojení.
- protokol není oficiálně zdokumentován
- existují dvě verze
  - NTLMv1 - MD4 a LMHash
  - NTLMv2 - HMAC
- NTLM používá stejné kryptografické mechanismy jako MS-CHAP
- protokol je typu výzva-odpověď

http://davenport.sourceforge.net/ntlm.html

Autentizační protokoly - NTLM



- od windows NT 4.0
- heslo již rozlišuje malá a velká písmena
- z důvodu kompatibility vyžaduje NTLM autentizace jak NTLM hash, tak i LANMAN hash -> bezpečnostní problém
- SAM obsahuje oba hashe

NTLM

- Útok
  - získat hashe ze SAM (program pwdump)
  - útok pomocí rainbow table na LM
    - získám heslo, ale neznám velká malá písmena
  - spočítat NTLM hash pro každou kombinaci písmen získanou z LANMAN hashe

Řízení přístupu v sítích

- pro ochranu IP sítí z hlediska přístupu se používá tzv. AAA architektura (AAA – authentication, authorization, accounting)

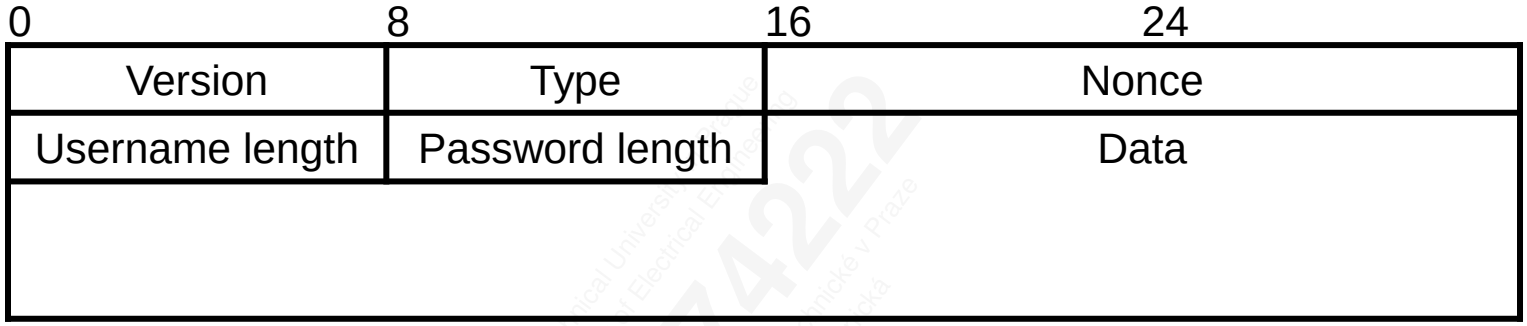
AAA obsahuje položky:

  - Autentizace
  - Autorizace
  - Účtování
- nepřímá autentizace
  - používá se ve větších sítích
  - přesun přihlašovacích údajů z koncových zařízení na jedno místo
  - centrální autentizační server
- v rámci architektury AAA se nejčastěji používají protokoly TACACS+ nebo RADIUS

TACACS

- Terminal Access Controller Access Controll Systém
- RFC1492 - http://www.faqs.org/rfcs/rfc1492.html
- Protokol pro autentizaci, autorizaci a řízení přístupu v sítích na bázi IP protokolu
- vznikl pro potřeby řízení přístupu v síti ARPANET (Advanced Research Projects Agency Network)
- TACACS sám o sobě není zabezpečený(tzn. nešifruje už. jména a hesla) a pokud již máte přístup do sítě lze ho jednoduše odposlouchávat a tak získávat hesla
- C/S architektura
- klient posílá požadavky na autentizaci/autorizaci k příslušnému serveru
- jako transportní protokol používá UDP
- každý TACACS paket je zapouzdřen do jednoho UDP paketu

TACACS

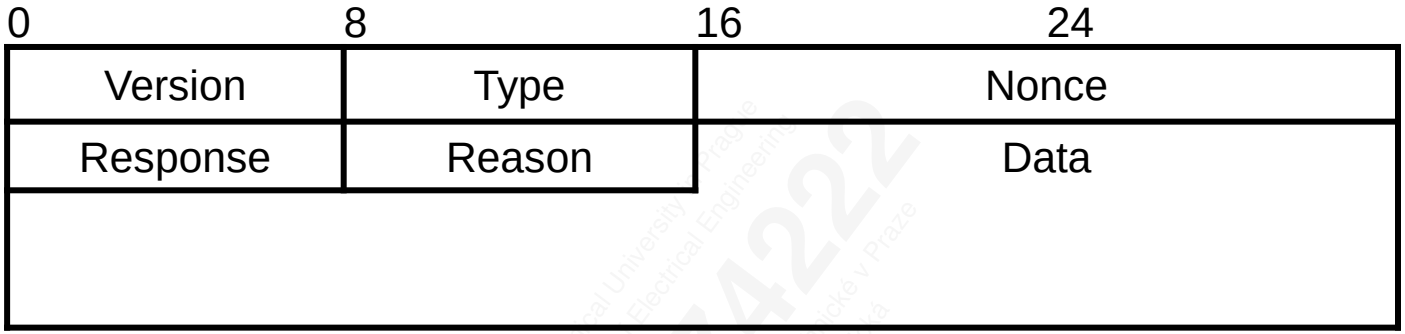


Formát paketu zprávy Request

Každý paket obsahuje pole :

- Version - 1B, verze - obsahuje hodnotu 0
- Type - 1B, typ zprávy – zda-li jde o výzvu nebo o odpověď
- Nonce - 2B, povinná hodnota spojující konkrétní odpověď s konkrétní výzvou
- Username length -1B, délka uživatelského jména
- Password length - 1B, délka hesla
- Data - *x*B, vlastní jméno a heslo

TACACS

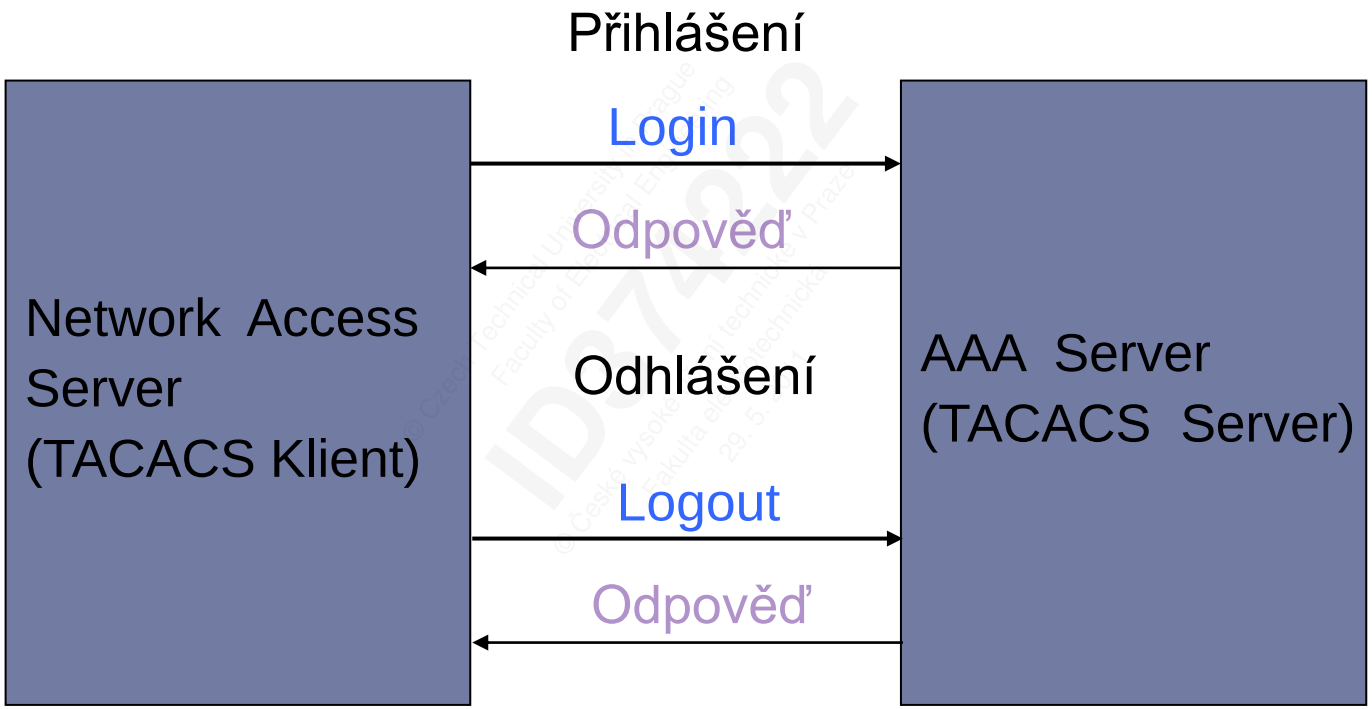


Formát paketu zprávy Reply/Result

Každý paket obsahuje pole :

- Verze (Version) - 1B, obsahuje hodnotu 0
- Typ (Type) - 1B, zda jde o výzvu nebo o odpověď
- Nonce - 2B, číslo spojující konkrétní odpověď s konkrétní výzvou
- Response - 1B, kód odpovědi (odmítnuto – reject nebo přijato – accept)
- Reason - 1B, pokud je odpověď reject, obsahuje toto pole zdůvodnění

TACACS – výměna zpráv

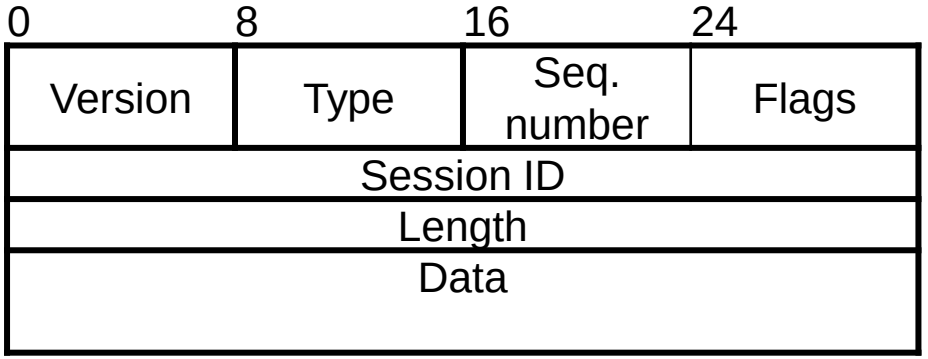


XTACACS

- rozšířená verze protokolu TACACS (eXtended TACACS)
- obsahuje dodatečnou podporu pro účtování (accounting) a bezpečnostní audity (auditing)
- dnes nahrazen protokolem TACACS+
- používá UDP
- XTACACS poskytuje dodatečné informace, které lze využít k auditování stejně podobně jako v UNIXových OS soubor utmp
- lze sledovat jak dlouho je uživatel přihlášen, ke kterým počítačům se připojil a na jak dlouho…

TACACS+

- podporuje všechny tři složky architektury AAA
- není kompatibilní s TACACS a XTACACS
- TACACS+ odděluje proces autentizace a autorizace
- to umožňuje např. použít k autentizaci Kerberos a k autorizaci TACACS+
- šifruje celý paket
- na transportní vrstvě používá protokol TCP
- proprietární CISCO protokol



TACACS+ - Accounting

- Start time – čas v sekundách uplynulý od půlnoci 1.1.1970
- Stop time – čas v sekundách uplynulý od půlnoci 1.1.1970
- Zaznamenává se např.:
  - uplynulý čas, po který byl uživatel přihlášen
  - počet odeslaných a přijatých bytů
  - počet odeslaných a přijatých paketů
  - zdroje ke kterým uživatel přistupoval
  - důvod, proč se uživatel odpojil od sítě
  - …

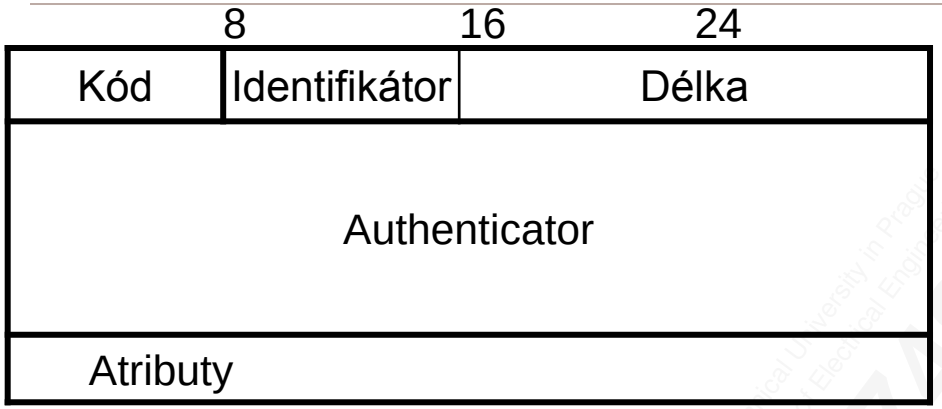
Srovnání TACACS, XTACACS, TACACS+

	TACACS	XTACACS	TACACS+
Použití	Autentizace / Autorizace	kompletní AAA	kompletní AAA
Transportní protokol	UDP	UDP	TCP
Charakter komunikace	Request/ Response	Request/ Response	Request/ Response
Hop-by-hop zabezpečení	žádné	žádné	celý paket zašifrován symetrickou šifrou (heslo se nepřenáší)
End-to-end zabezpečení	žádné	žádné	žádné
Velikost zprávy	6-516B	6-516B	hlavička (12B) + počet atributů (1 .. N) * atribut (8..255B)

RADIUS - Remote Authentication Dial-In User Service

- zahrnuje všechny tři složky architektury AAA
- systém RADIUS má tři složky:
  - protokol
  - autentizační server
  - klienty
- používá transportní protokol UDP
- transakce mezi serverem a klientem se autentizují pomocí sdíleného hesla, které se nikdy nepřenáší v síti
- hesla uživatelů se posílají mezi serverem a klienty zašifrovaná (na rozdíl od TACACS+ se šifrují pouze hesla)
- RADIUS je specifikován v RFC 2865, 2866
- používá se v 802.1x

RADIUS



Struktura paketu RADIUS

- Identifikátor spojuje související výzvy a odpovědi
  - Request Authenticator – obsahuje náhodné číslo
  - ReponseAuthenticator – slouží k autentizaci odpovědi od RADIUS serveru. Obsahuje hodnotu vzniklou z **MD5(Kód||ID||Délka||RequestAuth||Atributy||Tajnýklíč)**

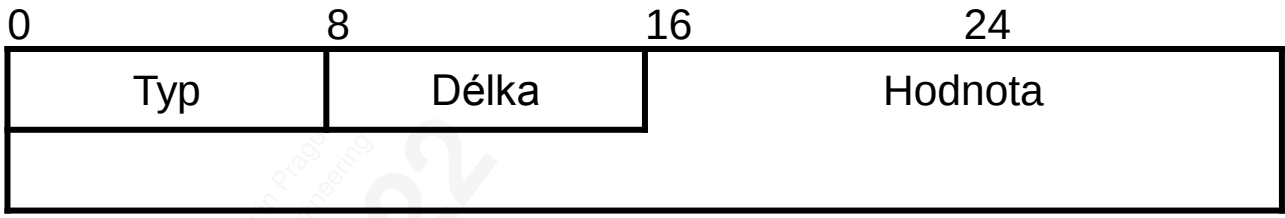
Kód	Popis
1	Access-Request
2	Access-Accept
3	Access-Reject
4	Accounting-Request
5	Acounting-Response
11	Access-Chalenge
12	Status-server (experimental)
13	Status-client (experimental)
255	Reservered

Kód určující typ RADIUS paketu

RADIUS

Příklady atributů

1 User-Name
2 User-Password
4 Login-IP-Host
15 Login-Service
16 Login-TCP-Port
17 (unassigned)
18 Reply-Message
19 Callback-Number
20 Callback-Id
21 (unassigned)
22 Framed-Route
…



Obecná struktura atributu

- RADIUS server/klient může ignorovat atributy neznámého typu (unknown)

DIAMETER

- RFC 3588
- následovník RADIUSu
- není přímo zpětně kompatibilní s RADIUSem
- Hlavní rozdíly:
  - používá spolehlivý transportní protokol (TCP nebo SCTP*)
  - může být zabezpečen pomocí IPsec nebo SSL
  - má dočasnou podporu pro RADIUS (tzn. DIAMETER server může vystupovat jako RADIUS)
  - má více adresního prostoru pro AVP(Attribute Value Pairs) a identifikátory (32 bitů místo 8)
  - je to C/S protokol, ale podporuje i „server-initiated“ zprávy

*SCTP – Stream Control Transmission Protocol

DIAMETER

- dokáže dynamicky objevovat peery (jiné DIAMATER server) (pomocí DNS SRV a NAPTR)
- má implementováno oznamování chyb
- má lepší podporu roamingu
- je snáze rozšiřitelný (dají se definovat nové příkazy a atributy)
- je zarovnán na 32 bitové hranice
- má základní podporu pro uživatelské relace a účtování (accounting)

Kerberos

- síťový autentizační systém sloužící k autentizaci přes nezabezpečené sítě
- vzájemná autentizace komunikujících stran
- **ne**zajišťuje autorizaci ani účtování
- základem jsou tzv. tickety (*ticket-lístek*), kterými navzájem identifikují účastníci komunikace a které slouží k výměně šifrovacích klíčů pro bezpečnou komunikaci s ostatními účastníky
- ticket je sekvence několika stovek bytů
- tickety lze vložit do kteréhokoliv jiného síťového protokolu a tak mu lze zajistit zaručenou identifikaci účastníků komunikace.
- Kerberos lze použít jako primárního autentizační systém, ale není to příliš obvyklé
- všechny zprávy obsahující citlivé informace (hesla) se přenášejí zašifrované

Kerberos

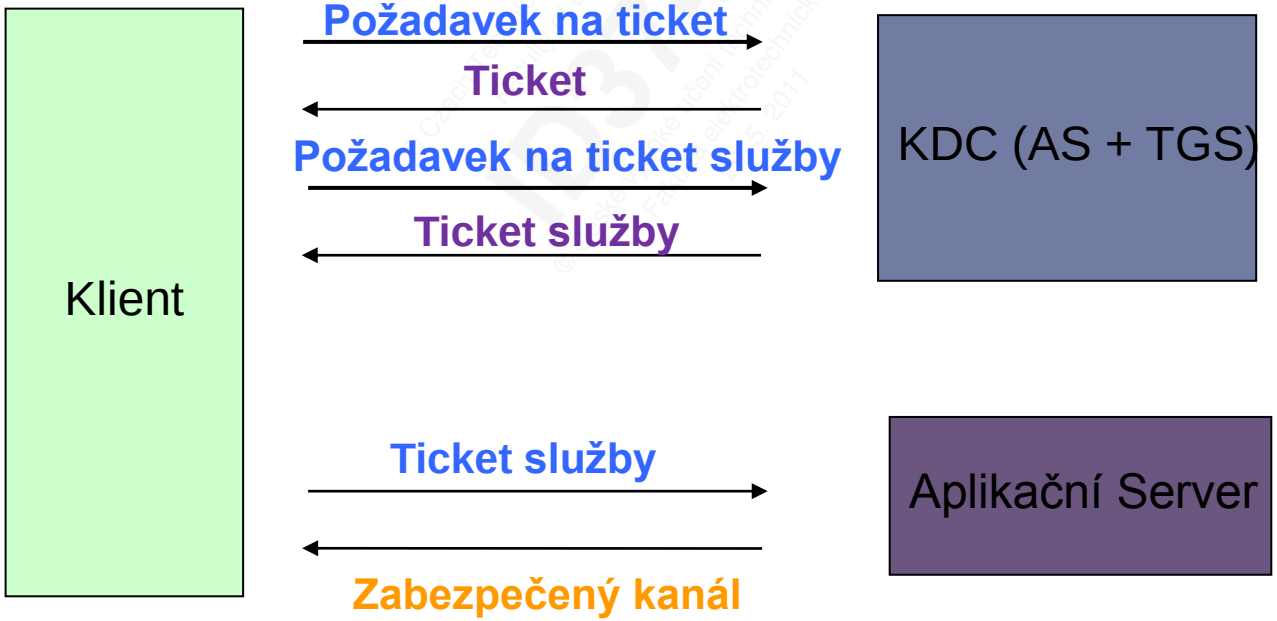
- KDC = AS + TGS
  - KDC – Key Distribution Center
  - AS - Authentication Server
  - TGS -Ticket Granting Server
- KDC využívá Needham-Schröederova schéma
- Kerberos využívá centralizovaný autentizační server , na který jsou směrovány všechny požadavky na autentizaci v síti. Tento server je apriori důvěryhodný.
- Podpora *Single-sign-on*
  - koncovému uživateli stačí jediné přihlášení k přístupu ke všem síťovým zdrojům, které podporují Kerberos . Poté, co se uživatel poprvé (úspěšně) autentizuje jsou jeho oprávnění (credentials) transparentně předávána všem ostatním zdrojům, které využívá .

Kerberos

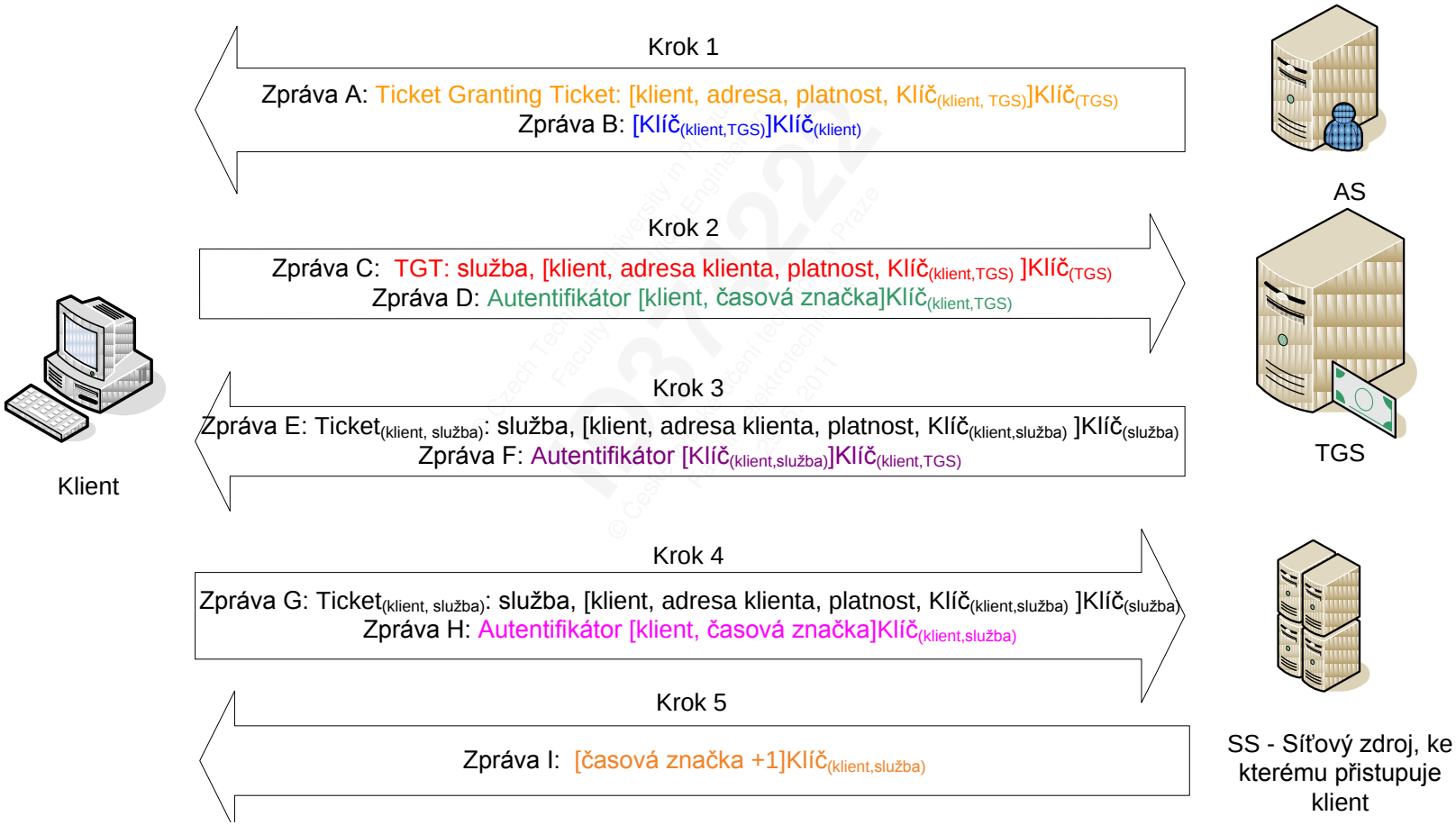
- vznikl v letech 1983-1991 na MIT
- nepřímý autentizační protokol
- C-S model
- vzájemná autentizace
- dnes se používá Kerberos v5
- několik volně dostupných implementací
  - MIT - http://web.mit.edu/Kerberos/
  - Heimdal - http://www.h5l.org/
  - Shishi - http://www.gnu.org/software/shishi/
- RFC 4120
- podpora SSO (Single Sign On)
- výchozí autentizační protokol ve Windows (od verze 2000)
  - používá vlastní implementaci protokolu

Kerberos

- single-point-of-failure
- zprávy šifrovány DES nebo AES (RFC 3962)
- nutná přesná časová synchronizace - NTP



Kerberos


Krok 1
Zpráva A: Ticket Granting Ticket: [klient, adresa, platnost, Klíč(klient, TGS)]Klíč(TGS)
Zpráva B: [Klíč(klient,TGS)]Klíč(klient)
AS
Krok 2
Zpráva C: TGT: služba, [klient, adresa klienta, platnost, Klíč(klient,TGS)]Klíč(TGS)
Zpráva D: Autentifikátor [klient, časová značka]Klíč(klient,TGS)
Klient
Krok 3
Zpráva E: Ticket(klient, služba): služba, [klient, adresa klienta, platnost, Klíč(klient,služba)]Klíč(služba)
Zpráva F: Autentifikátor [Klíč(klient,služba)]Klíč(klient,TGS)
TGS
Krok 4
Zpráva G: Ticket(klient, služba): služba, [klient, adresa klienta, platnost, Klíč(klient,služba)]Klíč(služba)
Zpráva H: Autentifikátor [klient, časová značka]Klíč(klient,služba)
Krok 5
Zpráva I: [časová značka +1]Klíč(klient,služba)
SS - Síťový zdroj, ke kterému přistupuje klient

Kerberos

Slovní popis jednou větou:

Klient se nejprve autentizuje vůči AS, poté prokáže TGS že je oprávněn žádat ticket pro danou službu, obdrží ho a poté prokáže serveru poskytujícímu danou službu, že ji může čerpat.

Slovní popis podrobně:

1. Uživatel zadá do klienta své už. jméno a heslo

2. Klient použije hashovací funkci a heslo k vytvoření tajného klíče klienta.

Kerberos

3. Klient odešle zprávu obsahující už. jméno v OT do AS ve které požaduje přístup k vybrané službě

   Příklad zprávy: "Uživatel XYZ by chtěl požádat o službu Q“
   Poznámka: do AS se neposílá ani heslo ani tajný klíč.

   4. AS ověří jestli má daného klienta v databázi. Pokud ano odešle mu dvě zprávy:

Zpráva A: klíč relace mezi klientem a TGS zašifrovaný pomocí tajného klíče uživatele

Zpráva B: Ticket-Granting Ticket (který obsahuje ID klienta , síťovou adresu klienta, dobu platnosti ticketu a klíč relace mezi klientem a TGS) zašifrovaný tajným klíčem TGS

Kerberos

5. Po doručení zpráv A a B klient dešifruje zprávu A, získá klíč relace mezi klientem a TGS. Tento klíč poté používá při komunikaci s TGS.

Poznámka: Klient nemůže dešifrovat zprávu B, protože je zašifrována klíčem TGS. Nyní se klient může autentizovat vůči TGS.

6. Při požadavku na službu, pošle klient do TGS dvě zprávy:

   Zpráva C obsahující TGT ze zprávy B a identifikátor požadované služby
   Zpráva D autentifikátor (skládá se z identifikátoru klienta a časové značky) zašifrovaná pomocí relačního klíče SS a TGS.

Kerberos

7. Po přijetí zpráv C a D je TGS dešifruje pomocí příslušného relačního klíče klient/TGS a pošle klientovy následující zprávy:

   Zpráva E obsahuje ticket Klient-Server (který zahrnuje ID klienta, síťovou adresu klienta, dobu platnosti) zašifrovaný pomocí tajního klíče SS

   Zpráva F je relační klíč klient/SS zašifrovaný pomocí relačního klíče klient/TGS.

Kerberos

8. Po přijetí zpráv E a F má klient všechny potřené informace nutné k autentizaci vůči SS. Klient se připojí k SS a odešle mu dvě zprávy:

   Zpráva G: ticket klient-Server zašifrovaný pomocí tajného klíče SS
   Zpráva H: nový autentifikátor obsahující ID klienta, časovou zančku. Tento požadavek je zašifrován relačním klíčem klient/Server.

Kerberos

9. SS dešifruje pomocí svého klíče ticket G pošle klientovi zprávu potvrzující jeho identitu a ochotu mu poskytnout požadovanou službu

   Zpráva I: časová značka v nalezená ve zprávě H zvýšená o 1 a zašifrovaná pomocí relačního klíče klient/Server.

10. Klient dešifruje potvrzení sdíleným klíčem klient/Server a ověří aktualizaci časové značky. Poté může klient důvěřovat Serveru a začít od něj čerpat požadovanou službu.

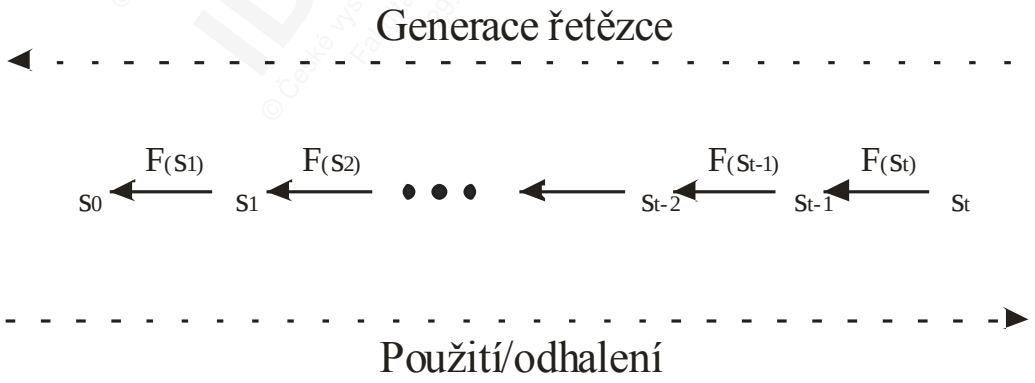
11. Server poskytne klientovi požadovanou službu.

Broadcastové autentizační protokoly

- protokoly pro broadcastovou/multicastovou komunikaci
- požadavky:
  - použitelnost na vysílací i přijímací straně
  - malá výpočetní a komunikační režie
  - rozšiřitelnost na velký počet příjemců
  - odolnost proti ztrátě paketů
  - real-time ověřování (žádné ukládání paketů do vyrovnávací paměti)

TESLA - Time Efficient Stream Loss-Tolerant Authentication

- protokol požaduje volnou časovou synchronizaci vysílače a přijímače (maximální odchylka Δ)
- k ověřování slouží tzv. jednosměrné řetězce
- jednosměrné řetězce jsou řetězce výstupních hodnot z hash funkce



TESLA

- V protokolu TESLA může přijímač informace pouze ověřovat.

Základní popis protokolu:

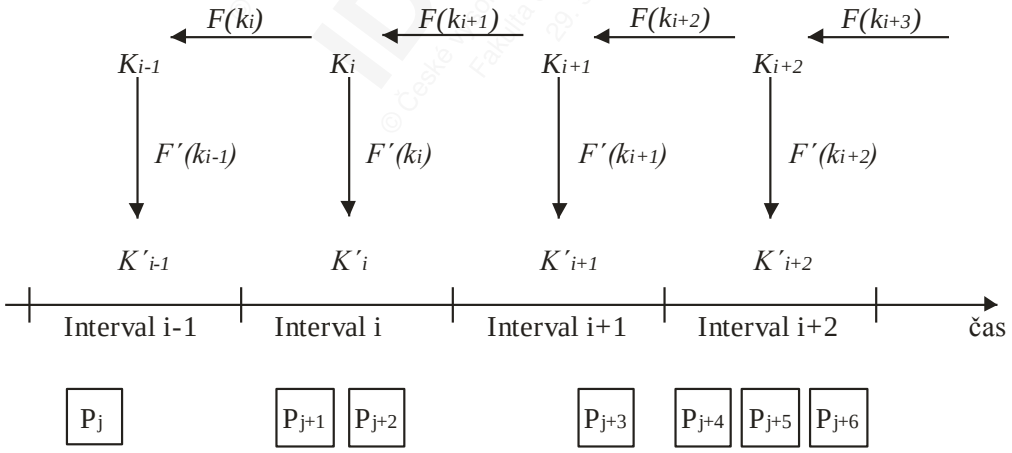
- vysílač rozdělí čas do úseků o stejné délce
- vytvoří jednosměrný řetěz ze samo-autentizačních hodnot (SEAL) a přiřadí tyto hodnoty jednotlivým časovým úsekům.
- jednosměrný řetěz je používám v obráceném pořadí, než byl vytvořen, proto nemůže být žádná hodnota použita k určení hodnoty předcházející.
- Vysílač stanoví čas, kdy budou odhaleny hodnoty jednosměrných řetězců. Hodnoty jsou tedy zveřejněny až v čase odhalení.

TESLA

- Vysílač přiloží ke každému paketu identifikační kód zprávy - MAC, který je pro každý paket unikátní a závislý na jeho obsahu.
- U všech paketů je určen časový interval a odpovídající hodnota z jednosměrného řetězce k vypočtení MAC. Spolu s paketem je odeslána i část jednosměrného řetězce, která může být zveřejněna.
- Paket dorazí k příjemci, který musí provést následující proceduru:
  - příjemce zná časový plán pro odkrývání paketů
  - je-li časově synchronizován, může zjistit, zda klíč použitý pro výpočet MAC je stále platný, tzn. ověří, jestli vysílač již odhalil klíče použité pro výpočet MAC. Pokud je MAC stále tajný, uloží ho příjemce do vyrovnávací paměti.

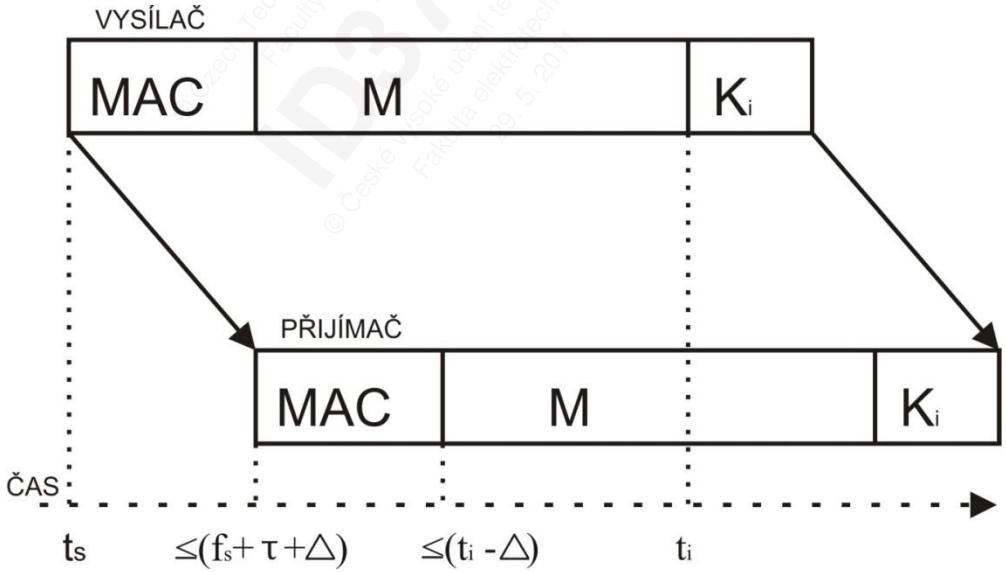
TESLA

- Každý příjemce ověřuje je-li odhalený klíč platný. Pomocí dříve odhalených klíčů každý příjemce ověří, je-li odhalený klíč platný. Pokud je všechno v pořádku, přijme paket k dalšímu zpracování.
- Ztráta paketů na přijímací straně neohrozí samotný autentizační protokol, protože chybějící hodnoty lze dopočítat pomocí následujících hodnot.
- Výhodné pro broadcastové vysílání, příjemce může ověřit pravost paketu, i když jsou některé klíče ztraceny.



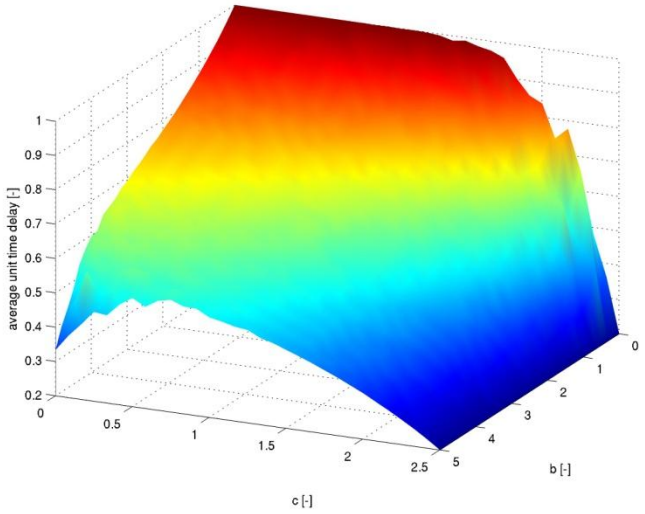
TIK - TESLA with Instant Key Disclosure

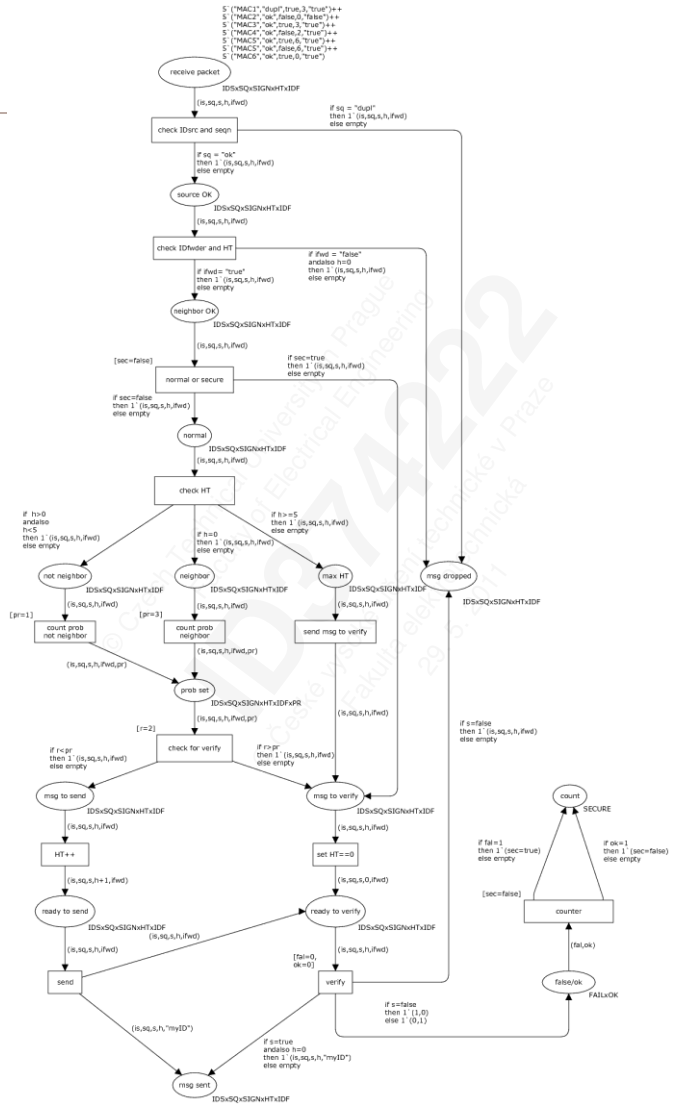
- odstraňuje nutnost ukládat příchozí pakety do vyrovnávací paměti
- vyžaduje přesnou synchronizaci hodin všech uzlů v síti
- vhodné pro bezdrátové sítě



DREAM

- DoS Resistant Authentication Mechanism
- neřeší vlastní autentizaci
- senzorové sítě
- broadcastová komunikace
- rozložení zátěže na více uzlů v síti




receive packet
check IDsrc and seqn
source OK
check IDfwder and HT
neighbor OK
normal or secure
normal
check HT
not neighbor
neighbor
max HT
msg dropped
count prob not neighbor
count prob neighbor
send msg to verify
prob set
check for verify
msg to send
msg to verify
HT++
set HT:=0
ready to send
ready to verify
send
verify
count
counter
false/ok
msg sent

Dotazy