Pretvarjanje identifikacijske sheme v shemo za digitalni podpis

Pokazali bomo še standarden način za pretvarjanje identifikacijske sheme v shemo za digitalni podpis.

Bojana, ki preverja Anitino identiteto, je potrebno zamenjati z javno zgoščevalno funkcijo (sporočilo torej ni zgoščeno pred podpisom, ampak zgoščevanje postane del podpisovanja).

Postopek si poglejmo kar na primeru Schnorrove sheme:

Naj bo p tako 512-bitno praštevilo, da je DLP v $\mathbb{Z}_p^*$ nedosegljiv problem, q 160-bitni delitelj števila $p-1$ in $\alpha \in \mathbb{Z}_p^*$ element reda q. Naj bo h zgoščevalna funkcija z zalogo vrednosti $\mathbb{Z}_q$, $\mathcal{P} = \mathbb{Z}_p^*$, $\mathcal{A} = \mathbb{Z}_p^* \times \mathbb{Z}_q$ in

$$\mathcal{K} = \{(p, q, \alpha, a, v) \mid v \equiv \alpha^{-a} \pmod{p}\}.$$

Vrednosti p, q, α so javne, vrednost a pa zasebna.

V praksi si običajno za zgoščevalno funkcijo h izberemo SHS, s 160-bitno zalogo vrednosti in z rezultatom, zreduciranim po modulu q (odšteti je potrebno največ en q).

V prehodu iz identifikacijske sheme na shemo za podpisovanje zamenjamo 40-bitni izziv z 160-bitno zgostitvijo sporočila:

Za $K = (p, q, \alpha, a, v)$ *in za tajno naključno število* $k \in \mathbb{Z}_q^*$, *definirajmo*

$$\mathrm{sig}_K(x, k) = (\gamma, y),$$

kjer $\gamma = \alpha^k \bmod p$ *in* $y = k + ah(x, \gamma) \bmod q$.

Za $x, \gamma \in \mathbb{Z}_p^*$ *in* $y \in \mathbb{Z}_q$ *definirajmo*

$$\mathrm{ver}(x, \gamma, y) = \mathrm{true} \iff \gamma \equiv \alpha^y v^{h(x,\gamma)} \pmod{p}.$$

Za domačo nalogo poskusite pretvoriti še kakšno izmed opisanih identifikacijskih shem v shemo za podpis.

Upravljanje z javnimi ključi

- Avtentikacijska drevesa
- Certifikatna Agencija (CA)
- Infrastruktura javnih ključev (PKI)
- Proces certifikacije
- Modeli zaupanja

Vprašanja in pomisleki

- Od kje dobimo javne ključe?
- Zakaj zaupamo javnim ključem?
- Kako veš čigav javni ključ imaš?
- Kako omejiti uporabo javnih ključev?
- Kaj se zgodi, če je kompromitiran (izgubljen) zasebni ključ? Kdo je odgovoren?
- Kako preklicati javni ključ ?
- Kako lahko obnovimo javni ključ?
- Kako omogočimo servis preprečitve zanikanja?

Upravljanje ključev

- *Upravljanje ključev*: množica tehnik in postopkov, ki podpirajo dogovor in vzdrževanje relacij ključev med pooblaščenimi strankami/sogovorniki.
- *Infrastruktura javnih ključev (PKI)*: podporni servisi (tehnološki, pravni, komercialni, itd.), ki so potrebni, da lahko tehnologijo javnih ključev uporabimo za večje projekte.

Tehnike za delitev javnih ključev

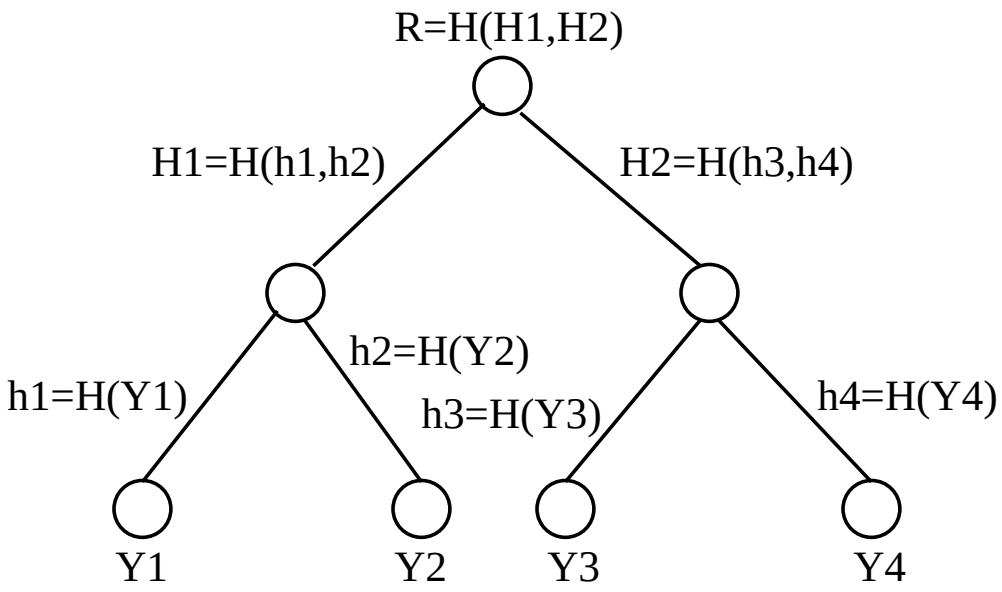
1. Point-to-point delitev po varnem kanalu:
   - zaupni kurir,
   - enkratna registracija uporabnikov,
   - glas.
2. Neposreden dostop zaupne javne datoteke.
   - Digitalno podpisana datoteka,
   - Avtentična drevesa.
3. Uporaba on-line zaupnih strežnikov,
4. Off-line certifikatna agencija (CA).

Avtentična drevesa

- Merkle, 1979.
- Metoda za delanje javno dostopnih in preverljivo overjenih podatkov.
- Aplikacije:
  - Avtentičnost velike datoteke javnih ključev,
  - servis časovnih oznak (Timestamping).

Primer avtentičnih dreves

- H je zgoščevalna funkcija brez trčenj.



- Vzdržujemo avtentičnost korenske vrednosti R (npr. podpis TTP).

- Za avtenticiranje javne vrednosti Y_2:
  - sledi (natanko določeno) pot od Y_2 do korena,
  - pridobi vrednosti h_1, H_2, R,
  - preveri avtentičnost R,
  - preveri $R = H(H(h_1, H(Y_2)), H_2)$.
- Če ima drevo n javnih vrednosti, je dolžina avtenticiranja kvečjemu $\lceil \log_2 n \rceil$.
- Slaba stran: dodajanje in brisanje javnih vrednosti je lahko precej zamudna.

Infrastruktura javnih ključev (PKI)

Nekatere komponente:

- format certifikata,
- proces certificiranja,
- razdeljenanje certifikatov,
- modeli zaupanja,
- preklic certifikatov,
- politika certificiranja: podrobnosti o namenu in obsegu uporabe določenega certifikata.
- Izjava o prakticiranju certificiranja (CPS) (postopki in politike CA).

Format certifikata: X.509 Ver.3

- X.509 originalno predlagan za podporo X.500, ki omogoča servis imenikov na velikih računalniških mrežah.
- Ver. 1 izide leta '88;
  Ver. 2 leta '93;
  Ver. 3 pa leta '97.
- Najnovejši PKI produkti uporabljajo Ver.3.
- Dopušča precejšnjo fleksibilnost.

Format certifikata X.509 Ver. 3

Podatkovna polja zajemajo:

- verzijo številke certifikata,
- certifikatovo serijsko številko,
- CA-jev podpisni algoritem ID,
- CA-jevo ime v X.500,
- rok veljave,
- strankino X.500 ime,
- strankina informacija o javnem ključu,
  - algoritmov ID, vrednost javnega ključa,
- Ext. polja: omogočajo vključevanje poljubnega števila dodatnih polj. Primeri:
  - Politika certifikata in politika prirejanja, pot certificiranja, omejitve.

Proces certifikacije

1. Generiranje para ključev za CA-jev podpis:
   - varnost zasebnega ključa CA je osrednja,
   - po možnosti opravljena v nepropustni napravi,
   - deljenje delov zasebnega ključa večim modulom, tako da certifikat ne more biti izdan s strani posameznega modula.
2. Generiranje para ključev osebe A:
   - bodisi s stani osebe A ali CA.

3. Zahteva za A-jev certifikat:
   - lahko, da bo CA kasneje potrebovala to zahtevo,
   - avtentičnost zahteve je potrebna.
4. Identiteta osebe A je preverjena:
   - to je lahko zamudno in drago v praksi,
   - preložiti to delo na Registration Authority (RA); npr. pošto ali banko,
   - RA generira registracijski certifikat in ga prosledi CA za izdajo certifikata.

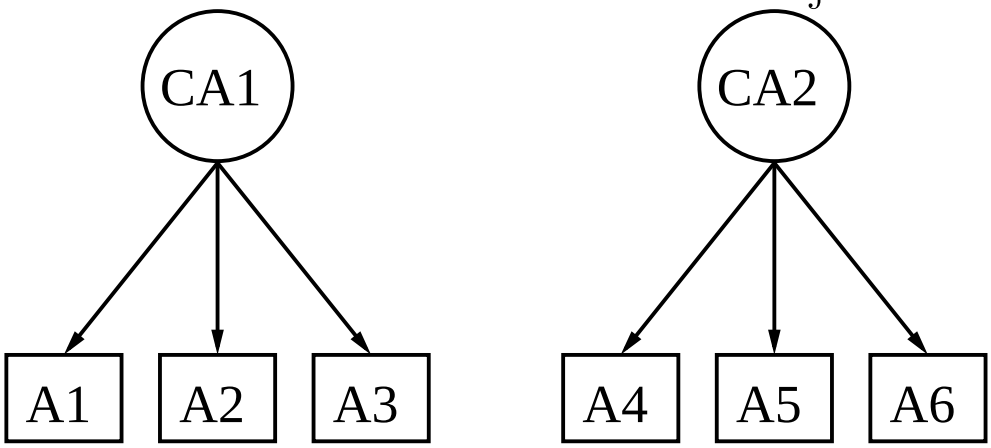
5. A-jev par ključev je preverjen:
   - CA preveri, da je javni ključ veljaven, tj. zasebni ključ logično obstaja,
   - A dokaže, da ima zasebni ključ.
6. CA naredi A-jev certifikat.
7. A preveri, da je certifikat izpraven:
   - CA lahko zahteva od A še potrdilo od prejemu.

Primer: Verisignov digitalni ID

- www.verisign.com/client/index.html
- Certifikat za javno podpisovanje in javno šifriranje.
- Certifikati so hranjeni v brskalniku ali e-poštni prog. opremi.
- Brezplačni certifikati za 60-dnevno preiskusno dobo.
- 3 razredi certifikatov:
  - odgovornost prevzema Verisign (US $100, $5,000, $100,000),
  - potrditev identitete,
  - zaščita CA-jevega zasebnega ključa,
  - zaščita posameznih uporabnikovih zasebnih ključev.
- Verisignov CPS: www.versign.com/repository/index.html

Model zaupanja

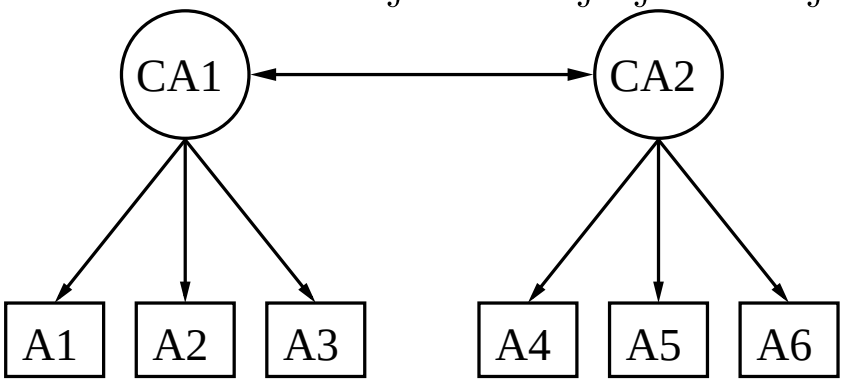
- strukturiran odnos med številnimi CA-ji.



- Stranke dobijo avtentične kopije CA-jevega javnega ključa (zunaj tekočega obsega - out-of-band, npr. med certifikacijo).
- Kako lahko A_1 preveri podpis sporočila osebe A_5? Tj. kako lahko dobi overjeno kopijo javnega ključa od A_1?
- A_1 potrebuje overjeno kopijo javnega ključa od CA_2.

Navzkrižna certifikacija

- CA-ji si lahko medsebojno overijo javne ključe

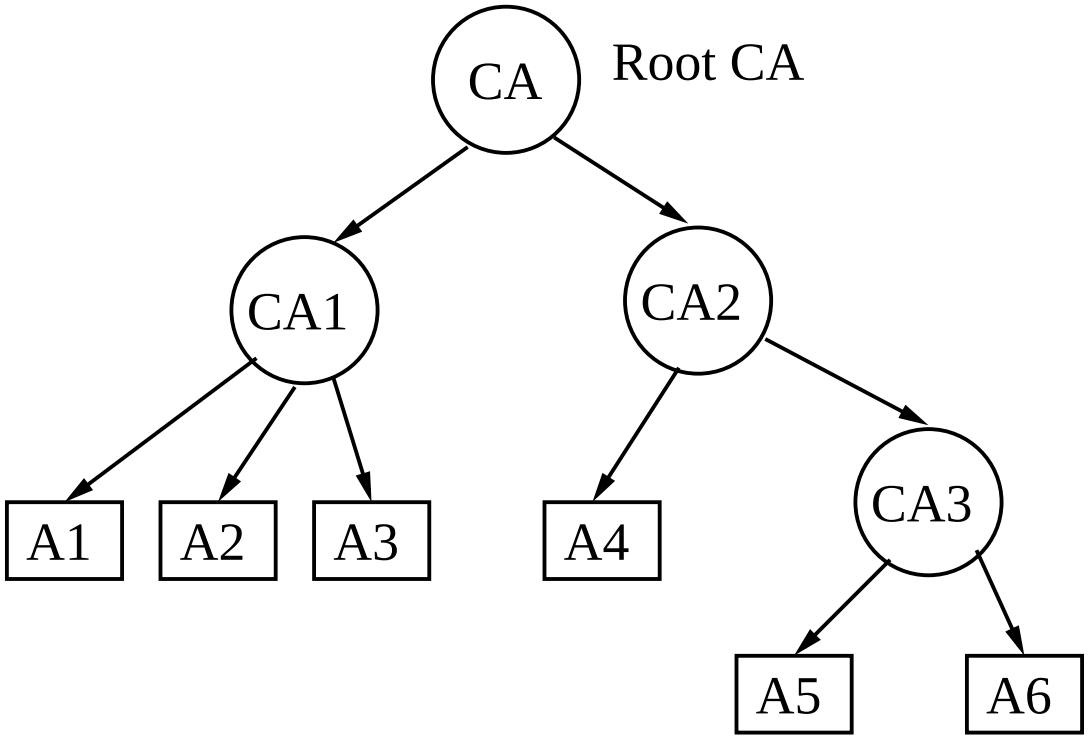


- A_1 pridobi A_5-jev overjeni javni ključ:
  - Pridobitev certifikatov CA_2 in A_5 z javnega (nezaščitenega, ne-overjenega) imenika.
  - Preveri od CA_1 podpisan certifikat CA_2. (s tem dobi overjeno kopijo javnega ključa CA_2).
  - Preveri od CA_2 podpisan certifikat A_5 (s tem dobi overjeno kopijo javnega ključa A_5).

Pomisleki glede navskrižnega certificiranja

- Ali je CA_1 odgovoren osebi A_1 za varnostne probleme v domeni CA_2?
  - Potencialni problemi so lahko omejeni z izjavo v politiki CA_1 za CA_2 certifikate.
  - CA_1 mora previdno preveriti CA_2jev CPS.
  - Neodvisni pregled politike CA_2 bo pomagal.
- Ali je CA_1 odgovoren osebam iz CA_2 domene za varnostne probleme v svoji domeni?
- Vprašanje: ali bodo problemi navskrižnega certificiranja za obsežnejše aplikacije *kdaj* rešeni?

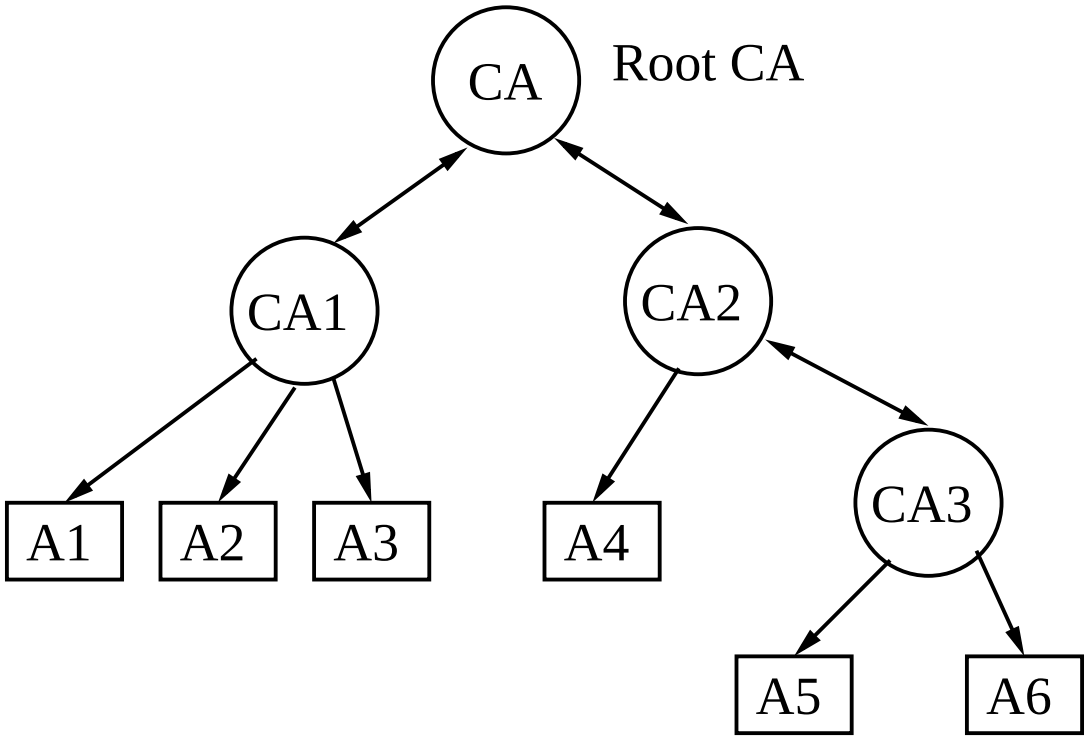
Strogo hierarhičen model



Strogo hierarhičen model (2)

- Vsi vipis začenjajo z overjeno kopijo korenskega javnega ključa.
- Zadrški:
  - vse zaupanje je odvisno od korenskega CA,
    - ∗ rešitev: razdeli dele zasebnega ključa;
  - Certifikatne verige lahko postanejo predolge,
    - ∗ rešitev: nekatere certifikate spravimo v cache.
  - Certifikatne verige zahtevane celo za osebe znotraj iste CA,
    - ∗ rešitev: nekatere certifikate spravimo v cache.

Povratni hierarhičen model



Povratni hierarhičen model (2)

- CA lahko preveri javni ključ starševskega CA.
- Vsaka oseba prične z overjenim javnim ključem svojega CA.
- Najkrajša veriga zaupanja med A in B je pot od A do najmlajšega skupnega prednika od A in B, in nato navzdol do B.

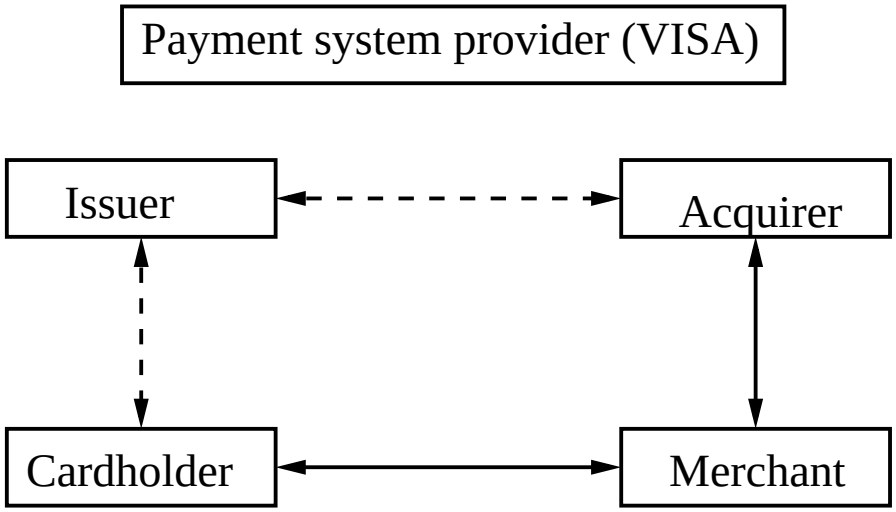
Secure Electronic Transaction (SET)

- Standard, ki sta ga predlagala Visa in MasterCard (Feb 1996).
- Glej www.setco.org
- Cilj: varne transakcije s kreditnimi karticami prko Interneta.

- Sodelujoči pri transakciji s kreditno kartico:
  - *Izdajatelj*: finančno podjetje, ki izdaja kreditne kartice.
  - *Lastnik kartice*: Nepooblaščen imetnik kreditne kartice holder of a credit card who is registered with the corresponding issuer.
  - *Prodajalec*: trgovec, services, or information, who accepts payment electronically.
  - *Dobavitelj*: finančna inštitucija, ki podpira prodajalca s tem, da ponuja servis za procesiranje transakcij z bančnimi karticami.

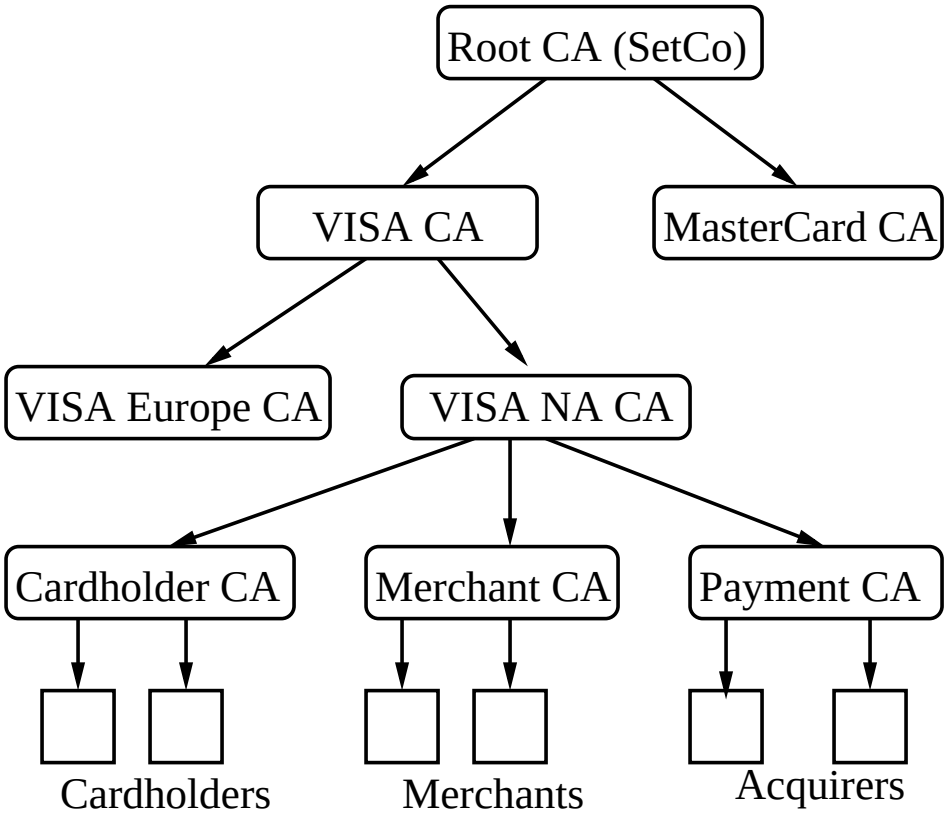
SET (2)

- Plačilo s kreditno kartico:



- Po Internetu: $C \longleftrightarrow M$ in $M \longleftrightarrow A$.
- Šifriranje se uporabi za zaščito številk kreditnih kartic med prenosom po Internetu; številke niso razkrite prodajalcu.
- Digitalni podpisi se uporabljajo za celovitost podatkov in overjanje udeleženih strank.

SET-ov hierarhični PKI



Preklic certifikata

- Razlogi za preklic certifikata:
  - kompromitiran ključ (redko).
  - Lastnik zapusti organizacijo.
  - Lastnik spremeni vlogo v organizaciji.
- Primer: Scotiabank tele-banking PKI:
  - Čez 90,026 certifikatov izdanih do aprila 21, 1999.
  - Čez 19,000 certifikatov preklicanih.
- Uporabnik naj bi preveril veljavnost certifikata pred njegovo uporabo.
- Preklic je enostaven v primeru on-line CA.

Certifikatne preklicne liste (CRL)

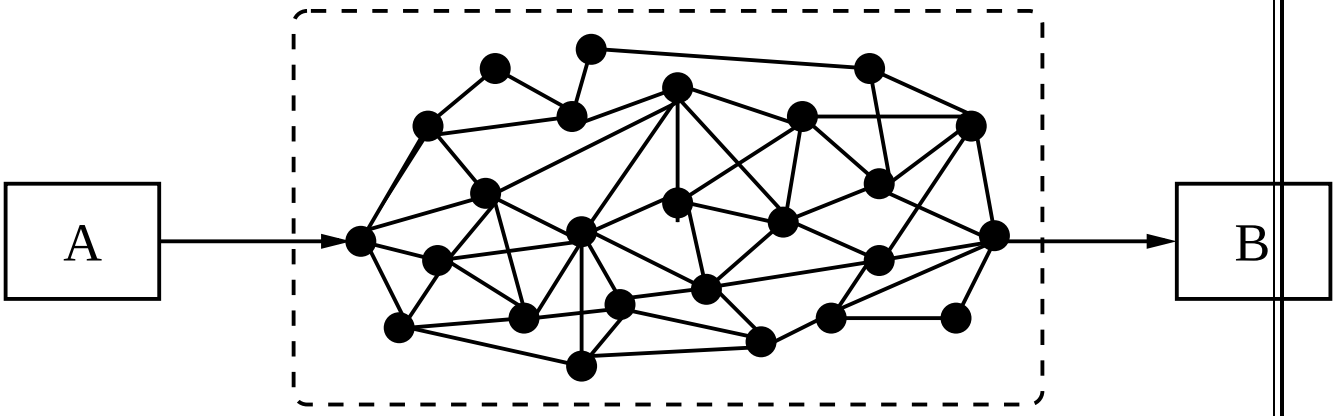
- Lista preklicanih certifikatov, ki je podpisana in periodično izdana od CA.
- Uporabnik preveri CRL predno uporabi certifikat.

Problemi z CRLs

- časovna prerioda CRL
  - Čas med preklicom in obnovitvijo CRL.
- Velikost CRL
  - Delta CRL: vključuje le zadnje preklicane certifikate.
  - Groupiraj razloge za preklic.
  - Distribucijske točke?: revocation data is split into buckets; each certificate contains data that determines the bucket it should be placed in (patent: Entrust Technologies).
  - Uporabi avtentikacijska drevesa (komercializacija: Valicert).

Internetna varnost

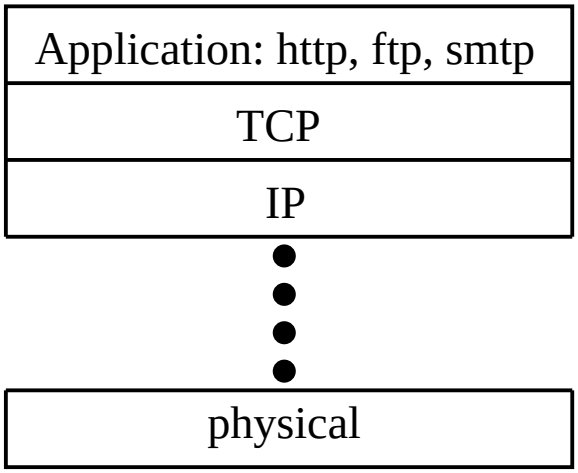
- Internet
- IPsec: Virtual Private Networks
- Secure Sockets Layer (SSL)
- Varna e-pošta



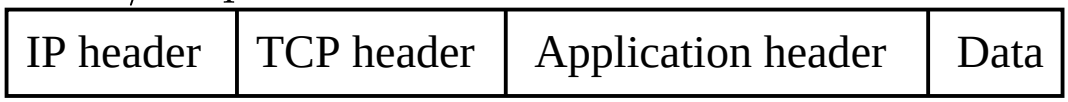
- Aplikacije:
  - ftp: File Transfer Protocol,
  - http: HyperText Transfer Protocol,
  - smtp: Simple Mail Transfer Protocol.
- TCP: Transport Control Protocol.
- IP: Internet Protocol.

TCP/IP

- Protokolov sklad:



- TCP/IP paket:



IP header:

Nekateri napadi

- IP address spoofing (slov. ponarejanje naslovov).
  - Rešitev: overi glavo IP paketa.
- IP packet sniffing (slov. vohljanje za IP paketi).
  - Rešitev: zašifriraj IP payload (tj. vse kar se prenaša).
- Traffic analysis (slov. Analiza prometa).
  - Rešitev: zašifriraj pošiljateljev in prejemnikov naslov.

Varnost znotraj TCP/IP

Varnostni protokoli so prisotni na različnih nivojih TCP/IP sklada.

1. IP nivo: IPsec.
2. Transportni nivo: SSL/TLS.
3. Aplikacijski nivo: PGP, S/MIME, SET, itd.

Internet Engineering Task Force (IETF)

- Sprejema standarde za razvoj Internetne arhitekture in omogoča nemoteno delovanje Interneta.
- Odprta za vse zainteresirane posameznike: www.ietf.org
- Delo, ki ga opravljajo delovne skupine povezane z varnostjo (Security Area) pokrivajo:

– IP Security Protocol (IPsec)

– Transport Layer Security (TLS)

– S/MIME Mail Security

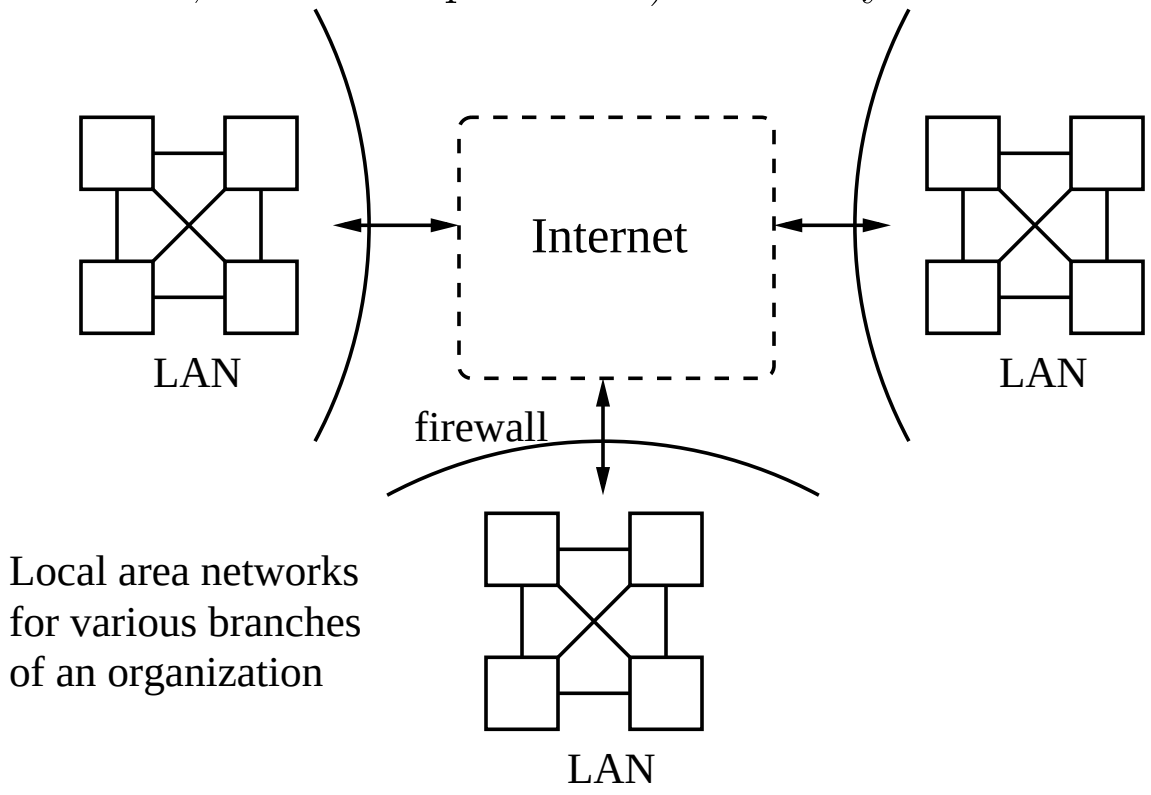
– Odprto specifikacijo za PGP (OpenPGP)

– Secure Shell (secsh)
(Nova verzija ssh protokola, ki omogoča varno prijavo na oddaljene šifre in varen prenos datotek.)

– X.509 Public-Key Infrastructure (PKIX)

IPsec: Virtual Private Networks (VPNs)

Omogočajo šifriranje in overjanje (overjanje izvora podatkov, celovitost podatkov) na IP layer.

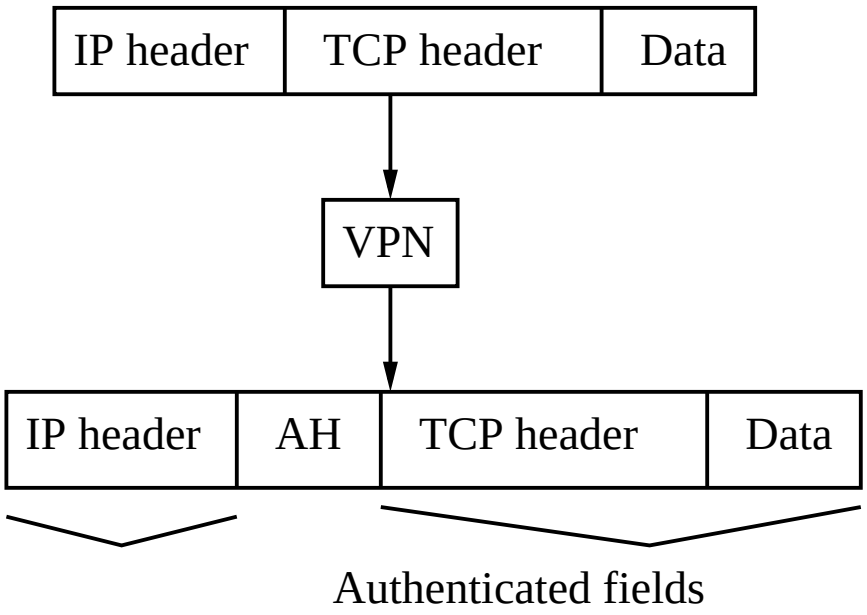


Gradniki IPsec

- Security Association (SA):
  - upravlja algoritme in ključe med sogovorniki,
  - vsaka glava IPsec se nanaša na Security Association preko Security Parameter Index (SPI).
- Upravljanje s ključi:
  - dogovor o ključu z Diffie-Hellmanovo shemo (OAKLEY),
  - kreira ključe za Security Association,
  - upravljanje z javnimi ključi, ki ni pokrito v IPsec.
- Trije načini IPsec servisov:
  - AH: overjanje,
  - ESP: šifriranje + overjanje.

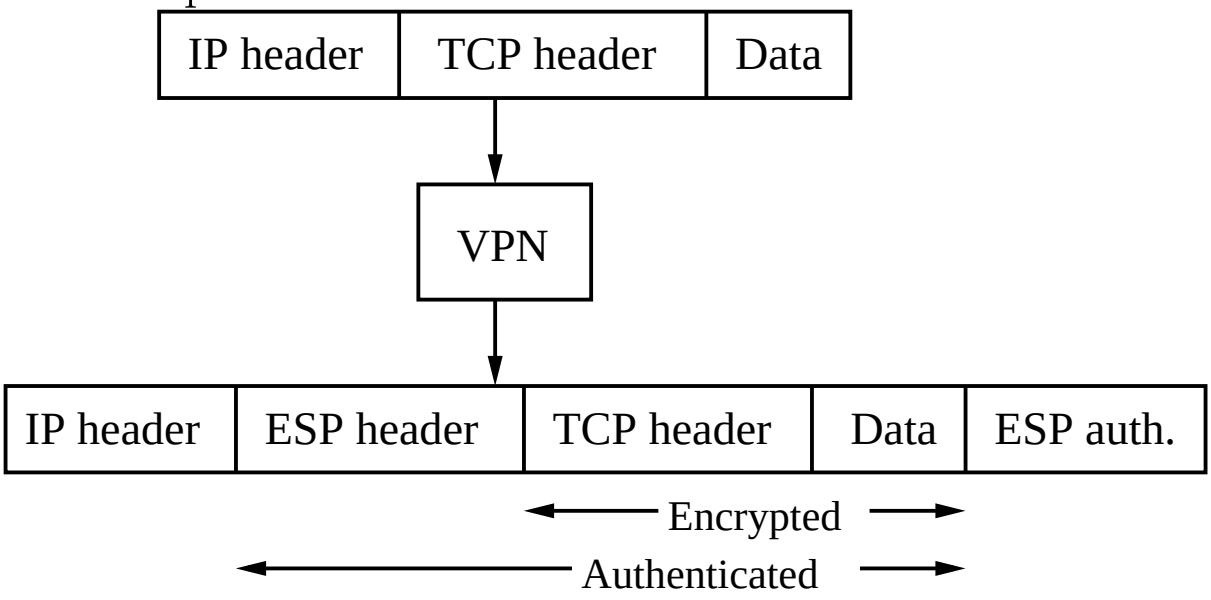
IPsec glava za overjanje (AH)

– Podpira MACs: HMAC-MD5-96, HMAC-SHA-1-96.
– Transportni način:



IPec ESP glava

- Encapsulating Security Payload.
- Podprti šifrirni algoritmi: 3-DES, RC5, IDEA, ...
- Transportni način:

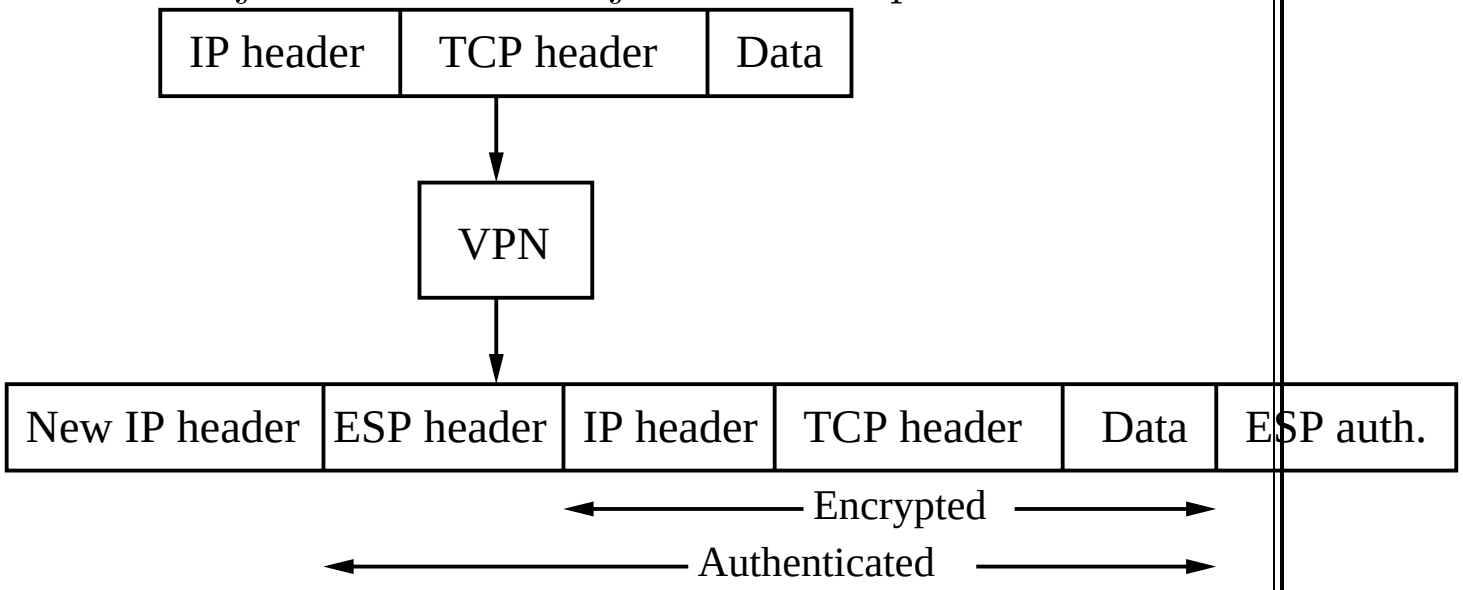


- Opomba: analiza prometa je še vedno možna (ker IP glave niso šifrirane).

ESP v tunelskem načinu

– Požarni zid vključi novo IP glavo (IP naslov pošiljateljevega
požarnega zidu in IP naslov prejemnikovega požarnega zidu).

– Možna je samo zelo omejena analiza prometa.



Secure Sockets Layer (SSL)

- SSL je naredil Netscape.
- TLS (Transport Layer Security) je IETF-ova verzija SSL-a.
- SSL uporabljamo v brskalnikih kot so Netscape za zaščito mrežnih transakcij.
- Osnovne komponente SSL/TLS:
  - Handshake protocol: Dopusti strežniki in klientu da se overita in dogovorita za kriptografske ključe.
  - Record protocol: Uporabljan za šifriranje in overjanje prenašanih podatkov.

Upravljanje z javnimi ključi v SSL/TLS

- Korenski CA ključ je vnaprej inštaliran v brskalnik.
  - Klik na "Security" in nato na "Signers", da najdete seznam ključev korenskih CA v Netscape-u.
- Mrežnim strežnikom certificirajo javne ključe z enim izmed korenskih CA-jev (seveda brezplačno).
  - Verisign-ov certification business za mrežne strežnike www.verisign.com/server/index.html

- Klienti (uporabniki) lahko pridobijo svoje certifikate. A večina uporabnikov trenutno nima svojih lastnih certifikatov.
  - Če klienti nimajo svojih certifikatov, potem je overjanje samo enostransko (strežnik se avtenticira klientu).
  - Obiščite varno internetno stran kot npr. webbroker1.tdwaterhouse.ca in kliknite na "padlock" v Netscapu, da si ogledate informacijo o strežnikovem certifikatu.

SSL/TLS handshake protocol

Na voljo so naslednji kriptografski algoritmi:

- MAC: HMAC-SHA-1, HMAC-MD5.
- šifriranje s simetričnimi ključi: IDEA, RC2-40, DES-40, DES, Triple-DES, RC4-40, RC4-128.
- Osnovne sheme za dogovor o ključu so:

– RSA transport ključev: deljeno skrivnost izbere klient in jo zašifrirana s strežnikovim javnim RSA ključem.

– Fixed Diffie-Hellman: strežnikov Diffie-Hellman-ov javni ključ g^x je v njegovem certifikatu.
Klient ima lahko g^y v svojem certifikatu, ali generira enkratno vrednost g^y.

– Ephemeral Diffie-Hellman: Strežnik izbere 1-kratni Diffie-Hellman-ov javni ključ g^x in ga podpiše s svojim RSA ali DSA ključim za podpise. Klient izbere 1-kratni g^y in ga podpiše če in samo če ima certifikat.

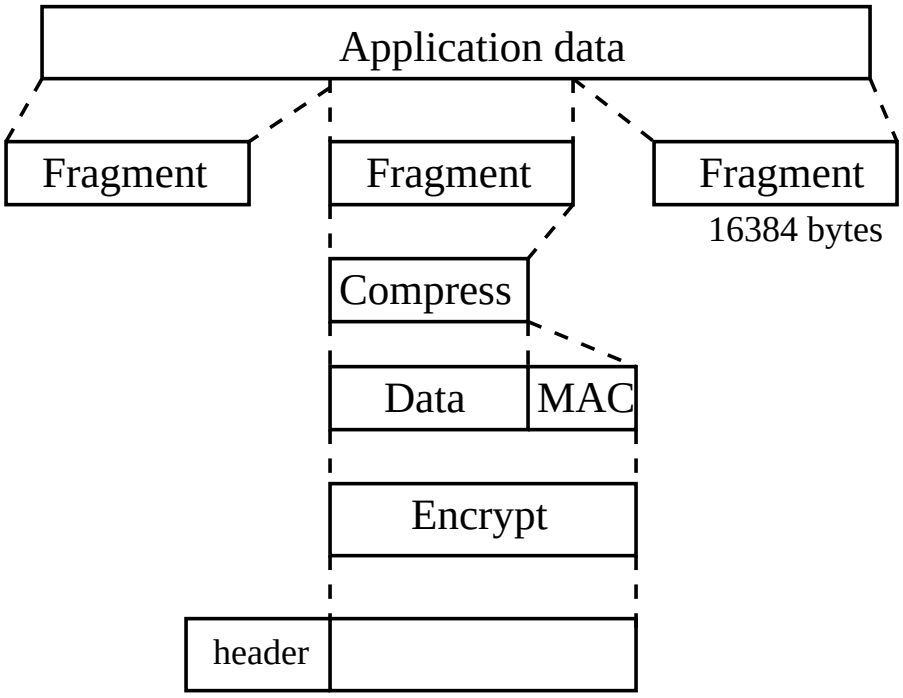
– MAC in šifrirni ključi so izpeljani iz skupne skrivnosti.

SSL/TLS handshake protokol (2)

1. faza: Določi varnostne zmožnosti.
   - Verzija protokola, način kompresije, kriptografski algoritmi,...
2. faza: Strežnikovo overjanje in izmenjava ključev.
   - Strežnik pošlje svoj certifikate, in (morda še) parametre za izmenjavo ključev.
3. faza: Klientovo overjanje in izmenjava ključeve.
   - Klient pošlje svoj certifikat (če ga ima) in parametre za izmenjavo ključev.
4. faza: Zaključek.

SSL/TLS record protocol

Predpostavimo, da klient in strežnik delita MAC tajnega ključa in sejni šifrirni ključ:



11. poglavje

Sheme za deljenje skrivnosti

(angl. **Secret sharing schemes**)

- Uvod
- Stopenjske sheme za deljenje skrivnosti
- Strukture dovoljenj
- Vizualne sheme za deljenje skrivnosti
- Formalne definicije
- Informacijska stopnja
- Ekvivalenca stopenjske sheme in OA

Deljenje skrivnosti

Kombinatorni problem:

n znanstvenikov dela na tajnem projektu, katerega materiali so spravljeni v trezorju z več ključavnicami.

Dostop do materialov je dovoljen, le kadar je prisotna večina znanstvenikov (tj. več kot polovica).

Vsak znanstvenik dobi enako število ključev.

Najmanj koliko ključavnic potrebujemo in koliko ključev mora dobiti vsak znanstvenik?

Rešitev: Naj bo $k = \lfloor (n+2)/2 \rfloor$ in $s = \binom{n}{k}$.
Potem imamo s različnih k-elementnih množic znanstvenikov: $G_1, G_2, \ldots, G_s$.

Osebe izven skupine G_i nimajo vseh ključev. Naj bo K_i množica, ključev, ki jim manjkajo. $K_i \neq \emptyset$, $i \in [1..s]$.

Skupaj s katerimkoli članom skupine G_i pa imajo vse ključe, torej ima vsaka oseba iz G_i vse ključe iz K_i.

Naj bo $i \neq j$. V množici G_i obstaja oseba, ki ni v G_j. Ta oseba nima nobenega izmed ključev iz K_j, torej je

$$K_i \cap K_j = \emptyset \quad \text{in zato} \quad \#\text{ključev} \geq s.$$

Pokažimo, da je $s = \binom{n}{k}$ ključev, tj, $k_1, \ldots, k_s$, dovolj za rešitev tega problema.

Ključe razdelimo tako, da dobijo ključ k_i le osebe iz skupine G_i. Torej dobi vsak znanstvenik $\binom{n-1}{k-1}$ ključev.

Le večinska skupina znanstvenikov ima neprazen presek z vsemi skupinami G_i, tako da lahko le taka skupina odpre trezor. ■

Problem: *V banki morajo trije direktorji odpreti trezor vsak dan, vendar pa ne želijo zaupati kombinacijo nobenemu posamezniku. Zato bi radi imeli sistem, po katerem lahko odpreta trezor poljubna dva med njimi.*

Ta problem lahko rešimo z $(2, 3)$-stopenjsko shemo.

Stopenjske sheme za deljenje skrivnosti sta leta 1979 neodvisno odkrila **Blakley in Shamir**.

V splošnem je **(t, n)-stopenjska shema** za deljenje skrivnosti K med n oseb (množica $\mathcal{P}$), $2 \leq t \leq n$, metoda, za katero velja

- poljubnih t oseb lahko izračuna vrednost K,
- nobena skupina s $t - 1$ osebami (ali manj) ne more izračunati prav nobene informacije o vrednosti K.

Varnost te sheme mora biti *brezpogojna*, tj. neodvisna od kakšnega računsko zahtevnega problema, kot je na primer faktorizacija v primeru RSA.

Uporaba:

- varno večstrankarsko računanje
  (*npr. kriptografske volilne sheme*)
- stopenjska kriptografija, večnivojske kontrole
  (*npr. skupinski podpisi*)
- upravljanje in delitev ključev
  (*npr. key escrow and keyrecovery schemes*)
- finance in bančništvo (*npr. elektronski denar*)

Revija Time (4. maj, 1992, str. 13)

V Rusiji imajo $(2, 3)$-stopenjsko shemo za kontrolo **nuklearnega orožja**:

- predsednik,
- obrambni minister,
- obrambno ministrstvo.

(2,2)-stopenjska shema

1. *Naj bo* $K = k_1 k_2 \ldots k_n, \quad k_i \in \mathbb{Z}_2$ (**skrivnost**).
2. *Delivec izbere naključna števila*
$$a_i \in \mathbb{Z}_2, \quad 1 \leq i \leq n$$
in izračuna $b_i = a_i + k_i \bmod 2, \quad 1 \leq i \leq n.$
3. *Anita in Bojan dobita zaporedoma* **dela**
$A = a_1 a_2 \ldots a_n$ *in* $B = b_1 b_2 \ldots b_n$ *za skrivnost* K.

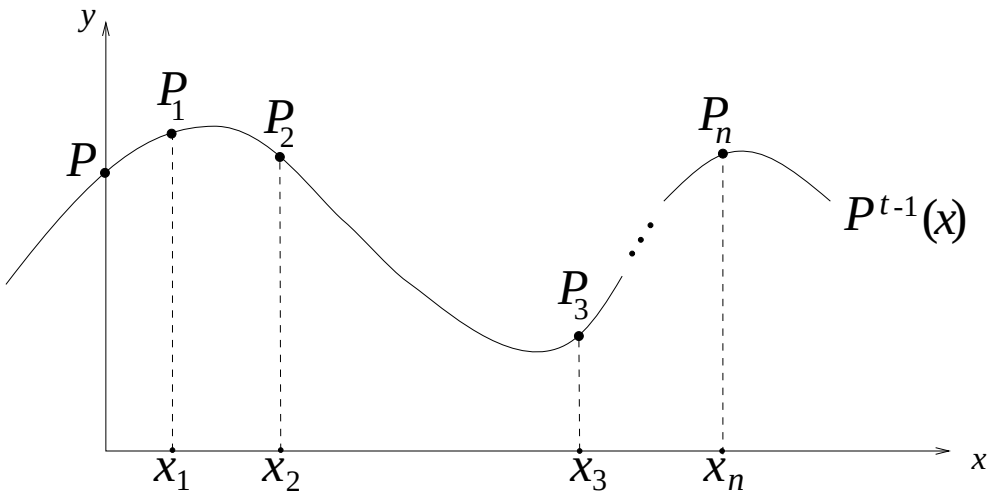
Ne Anita ne Bojan ne moreta vsak zase odkriti nobene informacije o skrivnosti, skupaj pa njuna dela A in B omogočata izračun ključa: $K = A + B \bmod 2$.

(t,t)-stopenjska shema

1. *Naj bo* $K \in \mathbb{Z}_p$ (**skrivnost**).
2. *Delivec* $D \notin \mathcal{P}$ *izbere neodvisno naključna števila* $y_1, y_2, \ldots, y_{t-1} \in \mathbb{Z}_m$, $m \geq r+1$, *in izračuna*
$$y_t = K - \sum_{i=1}^{t-1} y_i \bmod m.$$
3. *Oseba* P_i *dobi* **del** y_i, $1 \leq i \leq t$.

Osebe $P_1, \ldots, P_{i-1}, P_{i+1}, \ldots, P_t$ lahko izračunajo samo $K - y_i$, kar pa jim nič ne pomaga, saj je bilo število y_i naključno izbrano.

Shamir je skonstruiral tudi splošno (t, n)-stopenjsko shemo, za poljubna naravna števila t in n, $2 \leq t \leq n$:



1. *Delivec* $D \notin \mathcal{P}$ *izbere* n *različnih elementov* $x_1, x_2, \ldots, x_n \in \mathbb{Z}_p^*$, $p \geq n+1$, *in da* x_i *osebi* $P_i \in \mathcal{P}$ *(vrednosti* x_i *so javne).*

---

2. *Za delitev ključa* K *delivec* D *izbere naključno (neodvisno)* $t-1$ *elementov* $a_1, \ldots, a_{t-1} \in \mathbb{Z}_p$ *ter izračuna* $y_i = a(x_i)$ *in ga da osebi* P_i, $1 \leq i \leq n$,

$$\text{kjer je} \qquad a(x) = K + \sum_{j=1}^{t-1} a_j x^j \bmod p.$$

Osebe $P_1, P_2, \ldots, P_t$ določijo ključ K iz:

$$y_i = a(x_i) = a_0 + a_1 x_i + \cdots + a_t x_i^t, \quad \text{za } 1 \leq i \leq t$$

oziroma če zapišemo sistem enačb v matrični obliki

$$\begin{pmatrix} 1 & x_1 & x_1^2 & \ldots & x_1^{t-1} \\ 1 & x_2 & x_2^2 & \ldots & x_2^{t-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_t & x_t^2 & \ldots & x_t^{t-1} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_{t-1} \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_{t-1} \end{pmatrix}.$$

Koeficienti tvorijo Vandermondovo matriko z determinanto

$$\det A = \prod_{1 \leq i < j \leq t} (x_i - x_j) \bmod p \neq 0,$$

zato ima sistem enolično rešitev v $\mathbb{Z}_p$.

$t-1$ oseb ima $t-1$ enačb in t neznank.
Za poljuben $a_0 \in \mathbb{Z}_p$ dodamo še enačbo $a_0 = a(0)$
in zopet dobimo sistem z Vandermondovo matriko,
katere determinanta je različna od nič.

Torej ne morejo izključiti nobenega ključa K in to je res (t, n)-stopenjska shema za deljenje skrivnosti.

Do enakega zaključka bi lahko prišli tudi z
Lagrangovo interpolacijsko formulo za polinome:

$$a(x) = \sum_{j=1}^{t} y_i \prod_{1 \le j \le t, j \ne i} \frac{x - x_j}{x_i - x_j}.$$

Pravzaprav potrebujemo samo:

$$K = \sum_{j=1}^{t} y_i \prod_{1 \le j \le t, j \ne i} \frac{x_j}{x_j - x_i}.$$

Za $1 \le i \le t$ definirajmo

$$b_i = \prod_{1 \le j \le t, j \ne i} \frac{x_j}{x_j - x_i}.$$

Potem je ključ linearna kombinacija delov y_i:

$$K = \sum_{j=1}^{t} b_i y_i.$$

Strukture dovoljenj

L.1987 so **Ito, Saito** in **Nishizeki** vpeljali idejo shem za deljenje skrivnosti za poljubno strukturo dovoljenj.

Naj bo $\mathcal{P} = \{P_1, \ldots, P_n\}$ množica oseb, med katere želimo razdeliti skrivnost K. V splošnem si lahko želimo predpisati, katere podmnožice oseb iz $\mathcal{P}$ lahko izračunajo ključ in katere ga ne morejo.

Če so podmnožice iz družine $\Gamma \subseteq 2^{\mathcal{P}}$ natanko tiste množice oseb iz $\mathcal{P}$, ki lahko izračunajo ključ, potem množico Γ imenujemo **struktura dovoljenj**, njene elemente pa **pooblaščene** množice.

Popolna shema za deljenje skrivnosti, ki ustreza strukturi dovoljenj Γ, je metoda za deljenje ključa K na n oseb ($\mathcal{P}$) tako, da velja:

1. vsaka pooblaščena množica $B \subseteq \mathcal{P}$ lahko določi ključ K,
2. vsaka nepooblaščena množica $B \subseteq \mathcal{P}$ ne more odkriti čisto nič o ključu K.

Shamirjeva (t, n)-stopenljska shema je popolna, saj realizira strukturo dovoljenj

$$\{B \subseteq \mathcal{P} \,|\, t \leq |B|\}.$$

Študirali bomo brezpogojno varnost shem za deljenje skrivnosti (nepooblaščene množice imajo na voljo neomejeno računsko moč).

Monotonost: supermnožica pooblaščene množica je tudi pooblaščena.

Zanimale nas bodo samo monotone sheme za deljenje skrivnosti.

$B \in \Gamma$ je **minimalna** pooblaščena množica, če je $A \notin \Gamma$ za vsako podmnožico $A \subset B$.

Γ_0 je množica minimalnih pooblaščenih množic, **baza** za strukturo dovoljenj Γ. Množica

$$\Gamma = \{C \subseteq \mathcal{P} \mid B \subseteq C,\ B \in \Gamma_0\}$$

je potem zaprtje množice Γ_0 in jo bomo označili tudi z $\mathrm{cl}(\Gamma_0)$.

Konstrukcija z monotonim vezjem

Elegantna konstrukcija Benaloha in Leichera nas prepriča, da za vsako (monotono) strukturo dovoljenj obstaja popolna shema za deljenje skrivnosti.

Najprej bomo zgradili vezje, ki "prepozna" strukturo dovoljenj, potem pa iz njegovega opisa še shemo za deljenje skivnosti.

Naj bo $\mathcal{C}$ (booleansko) vezje z vhodi $x_1, \ldots, x_n$ (ki ustrezajo osebam $P_1, \ldots, P_n$) ter "OR" in "AND" vrati, tj. vrata "NOT" niso dovoljena (vsaka vrata imajo lahko poljubno število vhodov in le en izhod).

Takemu vezju $\mathcal{C}$ bomo rekli **monotono** vezje.

Za $B(x_1, \ldots, x_n) := \{P_i \,|\, x_i = 1\}$ je struktura dovoljenj

$$\Gamma(\mathcal{C}) = \{B(x_1, \ldots, x_n) \,|\, \mathcal{C}(x_1, \ldots, x_n) = 1\}$$

monotona (to sledi iz monotonosti vezja $\mathcal{C}$).

Ni se težko prepričati, da obstaja bijektivna korespondenca med monotonimi vezji in boolenskimi formulami z operatojema $\wedge$ ("AND"), $\vee$ ("OR") in brez negacije.

Naj bo Γ_0 baza za strukturo dovoljenj $\Gamma(\mathcal{C})$ in

$$\bigvee_{B \in \Gamma_0} \Big(\bigwedge_{P_i \in B} P_i \Big)$$

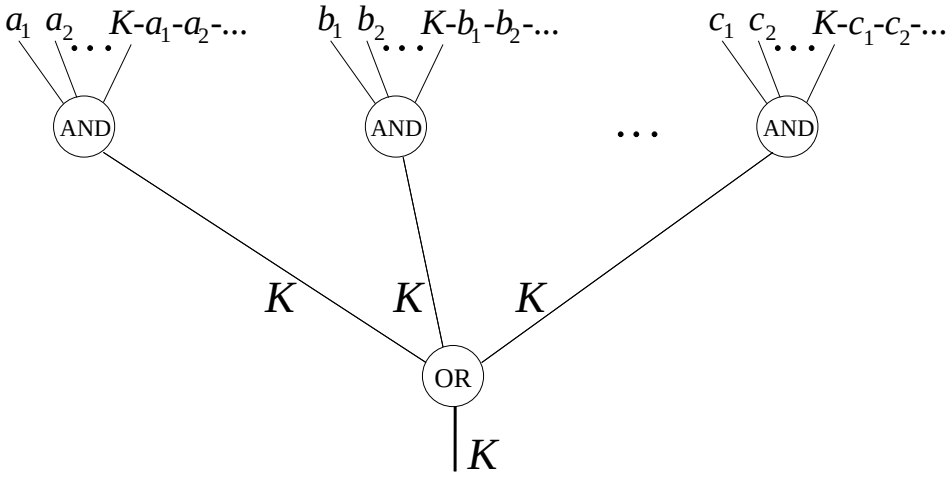
disjunktna normalna forma.

Primer: Za
$\Gamma_0 = \{\{P_1, P_2, P_4\}, \{P_1, P_3, P_4\}, \{P_2, P_3\}\}$ dobimo

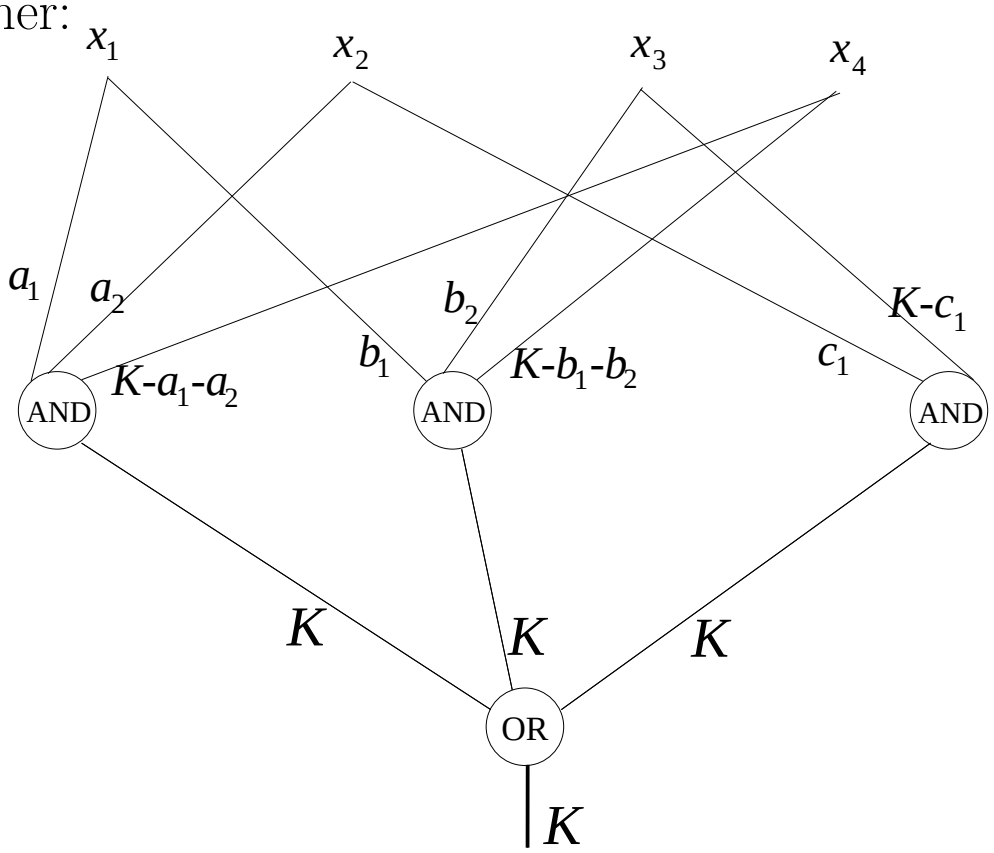
$$(P_1 \wedge P_2 \wedge P_4) \vee (P_1 \wedge P_3 \wedge P_4) \vee (P_2 \wedge P_3).$$

Skupno število vrat v zgornjem vezju je $|\Gamma_0| + 1$.

Sedaj pa naj bo $\mathcal{C}$ poljubno monotono vezje za strukturo dovoljenj Γ (ne nujno zgornje vezje) in $\mathcal{K} = \mathbb{Z}_m$, $m \in \mathbb{N}$. Uporabimo (t, t)-stopenjsko shemo.



Primer:



Drugačen pristop pa nam da konjunktivna normalna forma:

$$(P_1 \vee P_2) \wedge (P_1 \vee P_3) \wedge (P_2 \vee P_3) \wedge (P_2 \vee P_4) \wedge (P_3 \vee P_4)$$

- P_1 dobi a_1 in a_2,
- P_2 dobi a_1, a_3 in a_4,
- P_3 dobi a_2, a_3 in $K - a_1 - a_2 - a_3 - a_4$,
- P_4 dobi a_4 in $K - a_1 - a_2 - a_3 - a_4$.

Izrek 1. *Če je $\mathcal{C}$ monotono vezje, potem nam konstrukcija z monotonim vezjem da popolno shemo za deljenje skrivnosti, ki realizira strukturo dovoljenj $\Gamma(\mathcal{C})$.*

Dokaz: Popolna indukcija po številu vrat vezja $\mathcal{C}$.

Če imamo samo ena vrata, potem je trditev očitna. Sedaj pa naj bo $j > 1$ število vrat.

Zadnja vrata so "OR": $\Gamma(C) = \bigcup_{i=1}^{t} \Gamma(C_i)$.

Zadnja vrata so "AND": $\Gamma(C) = \bigcap_{i=1}^{t} \Gamma(C_i)$. ■