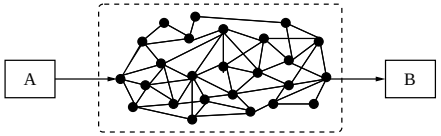
Internetne aplikacije

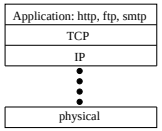


- ftp: File Transfer Protocol
- http: HyperText Transfer Protocol
- smtp: Simple Mail Transfer Protocol

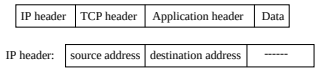
TCP – Transport Control Protocol
IP – Internet Protocol

TCP/IP

Protokolov sklad:



TCP/IP paket:



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 (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.

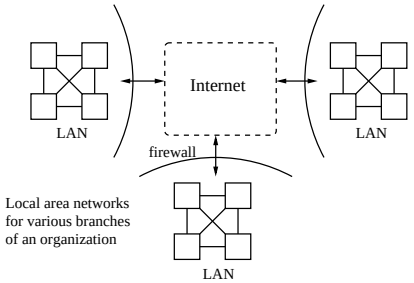
Internet Engineering Task Force (IETF)

- 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.

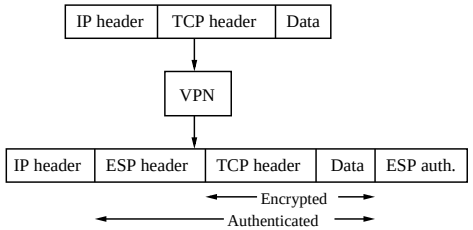


Gradniki IPsec

- 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.

IPec ESP glava

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

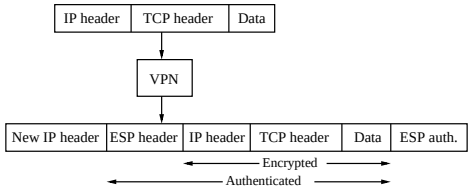


- Opomba: analiza prometa je še vedno možna (ker IP glave niso šifrirane).

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 (npr. Netscape) za zaščito mrežnih transakcij.
- Osnovne komponente SSL/TLS:

  handshake protocol: dopusti strežniku 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. 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 enkratni Diffie-Hellman-ov javni ključ g^x in ga podpiše s svojim RSA ali DSA ključem za podpise. Klient izbere enkratni g^y in ga podpiše če in samo če ima certifikat.

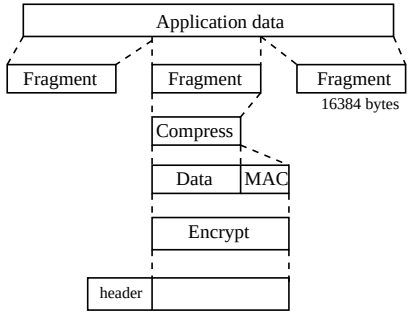
– MAC in šifrirni ključi so izpeljani iz skupne skrivnosti.

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:



9. poglavje

Identifikacijske sheme

oziroma **sheme za predstavljanje**:

- Uporaba in cilji identifikacijskih shem
- Protokol z izzivom in odgovorom
- Schnorrova identifikacijska shema
- Okomotova identifikacijska shema
- Guillou-Quisquater identifikacijska shema
- Pretvarjanje identifikacijske sheme v shemo za digitalni podpis

Pogosto hočemo dokazati svojo identiteto, npr.:

- **dvig denarja**
  (na bankomatu rabimo kartico in PIN)
- **nakup/plačilo**
  (prek telefona, potrebujemo kartico in rok veljave)
- **telefonska kartica** (telefonska številka in PIN)
- **prijava na svojo šifro na računalniku**
  (uporabniško ime in geslo)

Cilji identifikacijskih shem

- priča Anitine predstavitve Bojanu se ne more kasneje lažno predstaviti za Anito,
- tudi Bojan se ne more po Anitini predstavitvi lažno predstaviti za Anito,
- enostavnost (npr. za pametno/čip kartico)

Anita s svojo predstavitvijo ne izda informacije, ki jo identificira/predstavlja.

Kartica se predstavi sama, nepooblaščeno uporabo (kraja/izguba) pa preprečimo s PIN-om.

Protokol z **izzivom in odgovorom**:

Anita in Bojan delita tajni (skrivni) ključ K, ki ga uporabljata za šifriranje.

1. *Bojan izbere 64-bitni izziv x in ga pošlje Aniti.*
2. *Anita izračuna $y = e_K(x)$ in ga pošlje Bojanu,*
3. *Bojan izračuna $y' = e_K(x)$ in preveri $y = y'$.*

Skoraj vse sheme uporabljajo protokole z izzivom in odgovorom, vendar pa najbolj koristne ne uporabljajo skupnih ključev.

Schnorrova identifikacijska shema

Je ena od najbolj praktičnih shem in potrebuje agencijo TA.

1. *praštevilo p, za katero je DLP nedosegljiv problem (npr. $p \geq 2^{512}$),*
2. *velik delitelj q števila $p-1$ (npr. $q \geq 2^{140}$),*
3. *element $\alpha \in \mathbb{Z}_p^*$ reda q,*
4. *varnostni parameter t, za katerega je $q > 2^t$ (v praksi ponavadi vzamemo $t = 40$),*
5. *TA z algoritmoma za tajno podpisovanje $\mathrm{sig}_{\mathrm{TA}}$ in javno preverjanje $\mathrm{ver}_{\mathrm{TA}}$,*
6. *predpisana varna zgoščevalna funkcija.*

Parametri p, q in α, algoritem za preverjanje $\mathrm{ver}_{\mathrm{TA}}$ in zgoščevalna funkcija so javni.

Agencija TA izda Aniti certifikat:

1. *TA preveri Anitino identiteto po običajni poti (potni list, rojstni list, osebna izkaznica itd.) in izda* ID(Anita), *ki vsebuje identifikacijske podatke,*
2. *Anita si izbere zasebno naključno število $a \in [0, \ldots, q-1]$, izračuna $v = \alpha^{-a} \bmod p$ in ga izroči agenciji TA.*
3. *Agencija TA izračuna $s = \mathrm{sig}_{\mathrm{TA}}(\mathrm{ID}(\mathrm{Anita}), v)$ ter izroči Aniti potrdilo*

$$C(\mathrm{Anita}) = \big(\mathrm{ID}(\mathrm{Anita}), v, s\big).$$

Bojan preveri Anitino identiteto:

1. *Anita si izbere naključno število $k \in [0, \ldots, q-1]$ in izračuna $\gamma = \alpha^k \bmod p$, ki ga pošlje hkrati s svojim potrdilom C*(Anita) *Bojanu.*
2. *Bojan preveri podpis TA, izbere naključno število $r \in [1, \ldots, 2^t]$ in ga pošlje Aniti.*
3. *Anita izračuna $y = k + ar \bmod q$ in ga da Bojanu.*
4. *Bojan preveri, ali je $\gamma \equiv \alpha^y v^r \pmod p$.*

Podpis s potrdi Anitin certifikat
(tako kot pri uskladitvi ključa).

V drugem delu tajno število a deluje kot nekakšen PIN, saj prepriča Bojana, da je Anita res lastnica certifikata.

Za razliko od PIN-a Anita (oziroma bolj natančno pametna kartica) ne izda števila a, kljub temu, da "dokaže" z odgovorom na izziv z računanjem y-a v 3. koraku, da ga pozna.

Tej tehniki pravimo **dokaz brez razkritja znanja**.

Namen varnostnega parametra t je preprečiti, da bi napadalka, ki bi se hotela predstaviti za Anito, vnaprej uganila Bojanov izziv r (verjetnost $> 2^{40}$).

Če bi napadalka uganila r, bi si lahko za y izbrala poljubno število, izračunala

$$\gamma = \alpha^y v^r \bmod p$$

in ga poslala v 1. koraku Bojanu.
Ko bi prejela Bojanov izziv v drugem koraku, bi mu v 3. koraku dala že izbrani y in identiteta bi bila potrjena v 4. koraku.

Očitno Bojan ne sme uporabiti isti izziv r dvakrat.

Napadalka ne more ponarediti Anitin certifikat:

$$C'(\text{Anita}) = (\text{ID}(\text{Anita}), v', s'), \quad \text{kjer je } v \neq v',$$

saj bi v tem primeru znala ponarediti podpis s' od $(\text{ID}(\text{Anita}), v')$, ki ga v drugem koraku preveri Bojan.

(Vrednosti v' si ne moremo prosto izbirati, saj bi v tem primeru morali izračunati DLP, da bi dobili ustrezen a'.)

Napadalka ne more uporabiti niti Anitinega pravega certifikata $C(\text{Anita}) = (\text{ID}(\text{Anita}), v, s)$ (lahko bi ga spoznala pri prejšnjem preverjanju identitete), ker ne pozna a, ki ga potrebuje v 3. koraku za računanje y-a.

Izrek 1. *Če napadalka pozna število γ, za katero se zna z verjetnostjo $\varepsilon \geq 1/2^{t-1}$ predstaviti kot Anita, potem zna napadalka izračunati število a v polinomskem času.*

Dokaz: Predpostavimo, da lahko napadalka za ε od 2^t možnih izzivov r izračuna vrednost y, ki jo bo Bojan sprejel. Potem lahko zaradi $2^t\varepsilon \geq 2$ napadalka poišče taka para (y_1, r_1) in (y_2, r_2), da je

$$y_1 \not\equiv y_2 \pmod q \quad \text{in} \quad \gamma \equiv \alpha^{y_1} v^{r_1} \equiv \alpha^{y_2} v^{r_2} \pmod p.$$

Potem je

$$\alpha^{y_1-y_2} \equiv v^{r_2-r_1} \pmod p$$

in zaradi $v = \alpha^{-a}$ velja

$$y_1 - y_2 \equiv a(r_2 - r_1) \pmod q.$$

Končno je $0 < |r_2 - r_1| < 2^t$, število $q > 2^t$ pa je praštevilo, torej $D(r_2 - r_1, q) = 1$ in lahko izračunamo

$$a = (y_1 - y_2)(r_1 - r_2)^{-1} \bmod q.$$

■

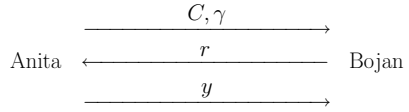
Ugotovili smo, da Anita zna potrditi svojo identiteto (*polnost*), vsak drug, ki zna to storiti z neznatno verjetnostjo (z uporabo identifikacijskega protokola) pa bodisi pozna zasebni a bodisi ga zna izračunati v polinomskem času (*uglašenost*).

To pa še ne pomeni, da je Schnorrov protokol varen, saj ima protokol, po katerem bi se Anita identitificirala enostavno tako, da bi odkrila svoj zasebni eksponent a, obe zgornji lastnosti.

Če napadalka ne izračuna nobene informacije o zasebnem eksponentu a medtem, ko je priča polinomskemu številu ponovitev Anitinega identifikacijskega protokola, potem je ta protokol **varen**.

Odprt problem: *Ali je Schnorrova shema varna?*

Naj ima ID(Anita) 512 bitov. Tudi v ima 512 bitov. Podpis s bo imel 320 bitov, če uporabimo DSS. Potem ima C(Anita) 1344 bitov. V prvem koraku mora Anita potencirati po modulu p, vendar pa lahko te vrednosti izračunamo vnaprej, če je potrebno.



Anita pošlje 1344+512=1856 bitov, nato Bojan pošlje 40 bitov in končno Anita pošlje še 140 bitov.

Okomotova identifikacijska shema

Izberimo parametra p, q tako kot v Schnorrovi shemi.

Naj imata elementa $\alpha_1, \alpha_2 \in \mathbb{Z}_p^*$ red q,
vrednost $c = \log_{\alpha_1} \alpha_2$ pa naj ne pozna niti Anita.

Kot pri Schnorrovi shemi si agencija TA izbere shemo za digitalni podpis in zgoščevalno funkcijo.

Agencija TA izda Aniti certifikat:

1. *Agencija TA preveri Anitino identiteto in ji izda* ID(Anita),
2. *Anita si izbere zasebni naključni števili* $a_1, a_2 \in [0, \ldots, q-1]$, *izračuna* $v = \alpha_1^{-a_1}\alpha_2^{-a_2} \bmod p$ *in ga izroči agenciji TA.*
3. *TA izračuna* $s = \mathrm{sig}_{\mathrm{TA}}(\mathrm{ID(Anita)}, v)$ *ter izroči Aniti potrdilo*

$$C(\mathrm{Anita}) = \big(\mathrm{ID(Anita)}, v, s\big).$$

Bojan preveri Anitino identiteto:

1. *Anita si izbere naključni števili* $k_1, k_2 \in [0,\ldots,q-1]$ *in izračuna* $\gamma = \alpha_1^{k_1}\alpha_2^{k_2} \bmod p$, *ki ga pošlje hkrati s svojim potrdilom* C(Anita) *Bojanu.*
2. *Bojan preveri podpis TA, izbere naključno število* $r \in [1, \ldots, 2^t]$ *in ga da Aniti.*
3. *Anita izračuna* $y_i = k_i + a_i r \bmod q$, *za* $i = 1, 2$ *in ju da Bojanu.*
4. *Bojan preveri, ali je* $\gamma \equiv \alpha_1^{y_1}\alpha_2^{y_2} v^r \pmod p$.

Okomotova shema je *polna*, za razliko od Schnorrove shema pa zanjo znamo pokazati, da je *varna*, kakor hitro je diskretni logaritem $\log_{\alpha_1} \alpha_2$ prezahteven.

Predspostavimo, da se je Anita identificirala tako, da je ponovila dani protokol polinomsko število krat in da je napadalka uspela priti do informacije o tajnih eksponentih a_1 in a_2. Pokazali bomo, da v tem primeru znamo izračunati c v polinomskem času, kar je seveda v protislovju s predpostavko.