Kerberos

Doslej smo spoznali sisteme, kjer vsak par uporabnikov izračuna fiksen ključ, ki se ne spreminja.
Zaradi tega je preveč izpostavljen nasprotnikom.

Zato bomo vpeljali tako imenovan sejni ključ, ki se oblikuje brž, ko se pojavita dva, ki želita komunicirati.

Tak sistem, ki uporablja simetrične sisteme, je Kerberos. Slabost tega sistema pa je zahteva po sinhronizaciji ur uporabnikov omrežja.

Določena časovna variacija je dovoljena.

Predpostavimo, da vsak uporabnik deli z agencijo TA tajni DES ključ K_U. Tako kot prej imejmo tudi ID(U).

Ko dobi agencija TA zahtevo po novem sejnem ključu, si TA izbere naključni ključ K, zabeleži časovno oznako T (timestamp), določi življenjsko dobo L (lifetime) za ključ K ter vse skupaj pošlje uporabnikoma U in V.

Prenos sejnega ključa z uporabo Kerberosa

- *Uporabnik U zahteva od agencije TA sejni ključ za komunikacijo z uporabnikom V.*
- *Agancija TA izbere naključni sejni ključ K, časovno oznako T in življenjsko dobo L.*
- *TA izračuna* $m_1 = e_{K_U}(K, \mathrm{ID}(V), T, L)$ *in* $m_2 = e_{K_V}(K, \mathrm{ID}(U), T, L)$ *ter ju pošlje uporabniku U.*
- *U uporabi odšifrirno funkcijo d_{K_U}, da dobi iz m_1 K, T, L in $\mathrm{ID}(V)$. Potem izračuna $m_3 = e_K(\mathrm{ID}(U), T)$ in ga pošlje osebi V skupaj s sporočilom m_2, ki ga je dobil od agencije TA.*

- *V uporabi odšifrirno funkcijo d_{K_V}, da dobi iz m_2 K, T, L in $\mathrm{ID}(U)$. Potem uporabi d_K, da dobi T in $\mathrm{ID}(U)$ iz m_3. Preveri, da sta tako dobljeni vrednosti za T in $\mathrm{ID}(U)$ enaki prejšnjim. Če je tako, potem izračuna še*

  $$m_4 = e_K(T+1)$$

  in ga pošlje uporabniku U.
- *U odšifrira m_4 z uporabo e_K in preveri, ali je rezultat enak $T+1$.*

V tem protokolu se prenašajo različne funkcije sporočil.

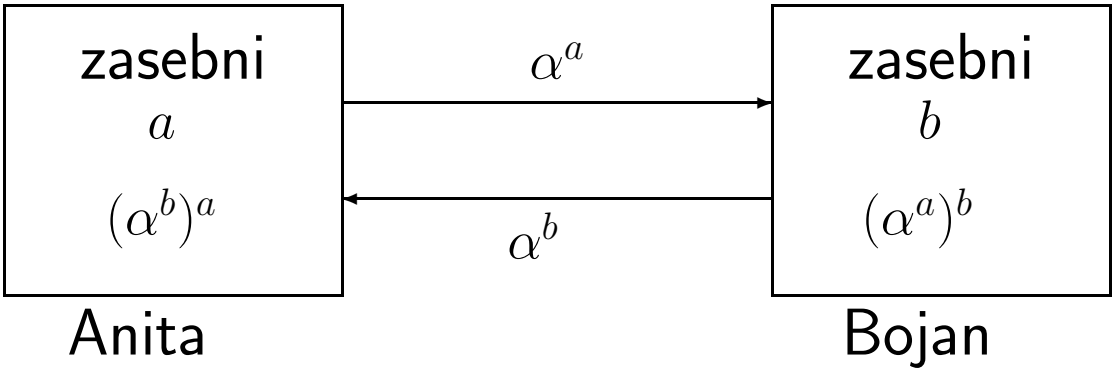
Sporočili m_1 in m_2 poskrbita za tajnost pri prenosu sejnega ključa K.

Sporočili m_3 in m_4 se uporabljata kot potrdilo sejnega ključa K tako, da se U in V prepričata, da imata res isti sejni ključ K.

Diffie-Hellmanova uskladitev ključev

Naj bo p praštevilo in α generator multiplikativne grupe $\mathbb{Z}_p^*$. Naj bosta oba javno poznana
(ali pa naj ju oseba U sporoči osebi V).

1. *Oseba* U *izbere naključen* a_U, $0 \leq a_U \leq p-2$, *izračuna* $\alpha^{a_U} \bmod p$ *in ga pošlje osebi* V.
2. *Oseba* V *izbere naključen* a_V, $0 \leq a_V \leq p-2$, *izračuna* $\alpha^{a_V} \bmod p$ *in ga pošlje osebi* U.
3. *Osebi* U *in* V *izračunata zaporedoma* $K = (\alpha^{a_V})^{a_U} \bmod p$ *in* $K = (\alpha^{a_U})^{a_V} \bmod p$.



Anita in Bojan si delita skupni element grupe:

$$(\alpha^a)^b = (\alpha^b)^a = \alpha^{ab}.$$

Edina razlika med tem protokolom in pa Diffie-Hellmanovim protokolom za distribucijo ključev je, da si izberemo nova eksponenta a_U in a_V uporabnikov U in V zaporedoma vsakič, ko poženemo ta protokol.

Varnost Diffie-Hellmanovega protokola

Protokol ni varen pred aktivnim sovražnikom,
ki prestreže sporočila in jih nadomesti s svojimi.
Ta napad bomo imenovali **napad srednjega moža**.

$$U \quad \underset{\alpha^{a'_V}}{\overset{\alpha^{a_U}}{\rightleftarrows}} \quad W \quad \underset{\alpha^{a_V}}{\overset{\alpha^{a'_U}}{\rightleftarrows}} \quad V$$

Na koncu sta osebi U in V vzpostavili z napadalcem W zaporedoma ključa $\alpha^{a_U a'_V}$ in $\alpha^{a'_U a_V}$.

Tako bo zašifrirano sporočilo osebe U odšifriral napadalec W ne pa oseba V.

Uporabnika U in V bi bila rada prepričana, da ni prišlo namesto medsebojne izmenjave sporočil do izmenjave z napadalcem W.

Potrebujeta protokol za medsebojno identifikacijo (predstavitev).

Dobro bi bilo, če bi potekala identifikacija istočasno z uskladitvijo ključev, saj bi s tem onemogočili aktivnega sovražnika.

Overjena uskladitev ključev

Diffie, Van Oorschot in Wiener so predlagali protokol **uporabnik-uporabniku** (station-to-station - STS), ki je protokol za *overjeno uskladitev kjuča* in je modifikacija Diffie-Hellmanove uskladitve ključev.

Vsak uporabnik ima **certifikat (potrdilo)**

$$C(U) = \Big(\mathrm{ID}(U), \mathrm{ver}_U, \mathrm{sig}_{\mathrm{TA}}\big(\mathrm{ID}(U), \mathrm{ver}_U\big)\Big),$$

kjer je shranjena njegova identifikacija $\mathrm{ID}(U)$.

Poenostavljen protokol uporabnik-uporabniku

1. *Oseba U izbere naključen $a_U \in \{0, \ldots, p-2\}$, izračuna $\alpha^{a_U} \bmod p$ in pošlje osebi V.*

2. *Oseba V izbere naključen $a_V \in \{0, \ldots, p-2\}$, izračuna $\alpha^{a_V} \bmod p$,*
   $K = (\alpha^{a_U})^{a_V} \bmod p$ in $y_V = \mathrm{sig}_V(\alpha^{a_V}, \alpha^{a_U})$,
   ter pošlje potrdilo $(C(V), \alpha^{a_V}, y_V)$ osebi U.

3. *Oseba U izračuna* $K = (\alpha^{a_V})^{a_U} \bmod p$ *ter preveri podpis* y_V *z uporabo* ver_V *in potrdilo* $C(V)$ *z* $\mathrm{ver}_{\mathrm{TA}}$.

   Nato izračuna $y_U = \mathrm{sig}_U(\alpha^{a_U}, \alpha^{a_V})$ *in pošlje potrdilo* $(C(U), y_U)$ *osebi* V.

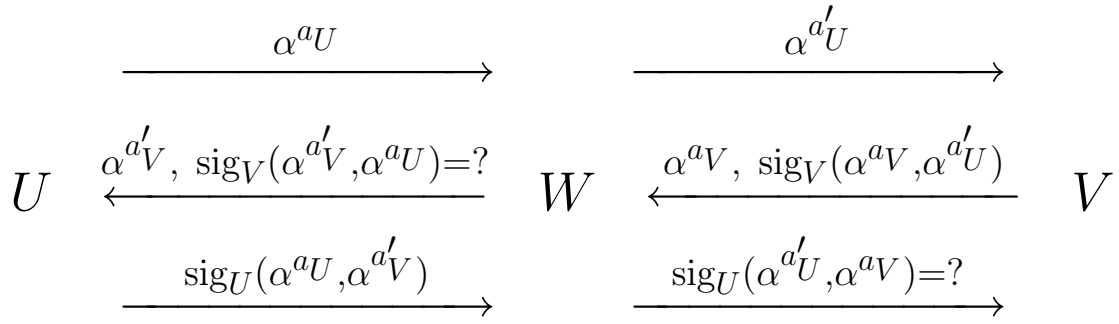
4. *Oseba V preveri podpis* y_U *z uporabo* ver_U *in potrdilo* $C(U)$ *z uporabo* $\mathrm{ver}_{\mathrm{TA}}$.

Varnost protokola STS

Uporabnika U in V si izmenjata naslednje informacije (izpustimo potrdila):

$$
\begin{array}{ccc}
 & \xrightarrow{\qquad\qquad \alpha^{a_U} \qquad\qquad} & \\
U & \xleftarrow{\qquad \alpha^{a_V},\ \mathrm{sig}_V(\alpha^{a_V}, \alpha^{a_U}) \qquad} & V \\
 & \xrightarrow{\qquad\qquad \mathrm{sig}_U(\alpha^{a_U}, \alpha^{a_V}) \qquad\qquad} &
\end{array}
$$

Kaj lahko naredi napadalec W (mož na sredini):



Poenostavljeni STS protokol je torej varen pred napadom srednjega moža.

Tako oblikovan protokol ne vsebuje potrditve ključa, kakor je slučaj v Kerberosovi shemi.

Protokol, v katerem je vključena potrditev ključa:

$$y_V = e_K(\mathrm{sig}_V(\alpha^{a_V}, \alpha^{a_U})), \quad y_U = e_K(\mathrm{sig}_U(\alpha^{a_U}, \alpha^{a_V}))$$

se imenuje STS protokol.

MTI protokoli

Matsumoto, Takashima, Imai so modificirali Diffie-Hellmanovo uskladitev ključev, tako da uporabniki U in V ne potrebujejo podpisov.

Kadar moramo izmenjati dve pošiljki, pravimo, da gre za **protokole z dvema izmenjavama**.

Predstavili bomo en njihov protokol.

Osnovne predpostavke so enake kot pri Diffie-Hellmanovi uskladitvi ključev: praštevilo p in generator α multiplikativne grupe $\mathbb{Z}_p^*$ sta javna.

Vsak uporabnik U ima svoj *zasebni* eksponent a_U $(0 \leq a_U \leq p-2)$ in *javno* vrednost $b_U = \alpha^{a_U} \bmod p$.

Agencija TA ima shemo za digitalni podpis, z *javnim* algoritmom $\mathrm{ver}_{\mathrm{TA}}$ in *tajnim* algoritmom $\mathrm{sig}_{\mathrm{TA}}$.

Vsak uporabnik U ima svoj certifikat:

$$C(U) = (\mathrm{ID}(U), b_U, \mathrm{sig}_{\mathrm{TA}}(\mathrm{ID}(U), b_U)).$$

1. *Oseba U izbere naključen $r_U \in \{0, \ldots, p-2\}$, izračuna* $s_U = \alpha^{r_U} \bmod p$ *in pošlje osebi V* $(C(U), s_U)$.

2. *Oseba V izbere naključen $r_V \in \{0, \ldots, p-2\}$, izračuna* $s_V = \alpha^{r_V} \bmod p$ *in pošlje osebi U* $(C(V), s_V)$.

3. *Osebi U in V izračunata zaporedoma*

$$K = s_V^{a_U} b_V^{r_U} \bmod p \quad \text{in} \quad K = s_U^{a_V} b_U^{r_V} \bmod p,$$

kjer sta b_V in b_U zaporedoma iz $C(V)$ in $C(U)$.

Varnost protokola MTI

Ta MTI protokol je enako varen pred pasivnimi sovražniki kot Diffie-Hellmanov protokol.

Varnost pred aktivnimi sovražniki je bolj vprašljiva. Brez uporabe podpisnega algoritma nismo varni pred napadom srednjega moža.

$$U \quad \begin{array}{c} \xrightarrow{\qquad C(U), \alpha^{r_U} \bmod p \qquad} \\ \xleftarrow{\qquad C(V),\ \alpha^{a_V} \bmod p \qquad} \end{array} \quad V$$

Ključ uporabnikov, ki komunicirata, je težko izračunati, ker je v ozadju težko izračunljiv diskretni logaritem.

Tej lastnosti pravimo **implicitna overitev ključev**.

Uskladitev ključev s ključi, ki se sami overijo

Giraultova shema ne potrebuje certifikatov, saj uporabnike razlikujejo že njihovi javni ključi in identifikacije.

Vsebuje lastnosti RSA sheme in diskretnega logaritma.

Uporabnik naj ima identifikacijo ID(U).
Javni ključ za osebno overitev dobi od agencije TA.

Naj bo $n = p\,q,$ kjer je $p = 2p_1 + 1,\ q = 2q_1 + 1$, in so p, q, p_1, q_1 velika praštevila. Potem je

$$(\mathbb{Z}_n^*, \cdot) \sim (\mathbb{Z}_p^* \times \mathbb{Z}_q^*, \cdot).$$

Največji red poljubnega elementa v $\mathbb{Z}_n^*$ je najmanjši skupni večkratnik elementov $p - 1$ in $q - 1$ oziroma $2p_1q_1$.

Naj bo α generator ciklične podgrupe v $\mathbb{Z}_p^*$ reda $2p_1q_1$, problem diskretnega logaritma v tej podgrupi pa naj bo računsko prezahteven za napadalca.

Javni ključ za osebno overitev

Naj bosta števili n, α *javni*,
števila p, q, p_1, q_1 *pa naj pozna samo agencija TA.*

Število e *je javni RSA šifrirni eksponent in ga izbere agancija TA,* $d = e^{-1} \bmod \varphi(n)$ *pa je tajni odšifrirni eksponent.*

1. *Oseba* U *izbere tajni eksponent* a_U,
   izračuna $b_U = \alpha^{a_U} \bmod n$ *in*
   izroči a_U *ter* b_U *agenciji TA.*
2. *Agencija TA izračuna*
   $p_U = (b_U – \mathrm{ID}(U))^d \bmod n$ *ter ga izroči osebi* U.

Giraultov protokol za uskladitev ključev

1. *Oseba U izbere naključen zasebni r_U, izračuna*
$$s_U = \alpha^{r_U} \bmod n$$
ter pošlje $\mathrm{ID}(U)$, p_U *in* s_U *osebi* V.
2. *Oseba V izbere naključen zasebni r_V, izračuna*
$$s_V = \alpha^{r_V} \bmod n$$
ter pošlje $\mathrm{ID}(V)$, p_V *in* s_V *osebi* U.
3. *Osebi U in V izračunata ključ K zaporedoma z*

$$s_V^{a_U}(p_V^e + \mathrm{ID}(V))^{r_U} \bmod n, \quad s_U^{a_V}(p_U^e + \mathrm{ID}(U))^{r_V} \bmod n.$$

Varnost Giraultovega protokola

Ključ za osebno overitev varuje pred sovražniki.

Protokol implicitno overi ključe, zato napad srednjega moža ni možen.

Agencija TA je prepričana, da uporabnik pozna vrednost števila a predno izračuna ključ za osebno overitev.

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
- 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 (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* $C(\text{Anita})$ *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 drugen 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:

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

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

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

■

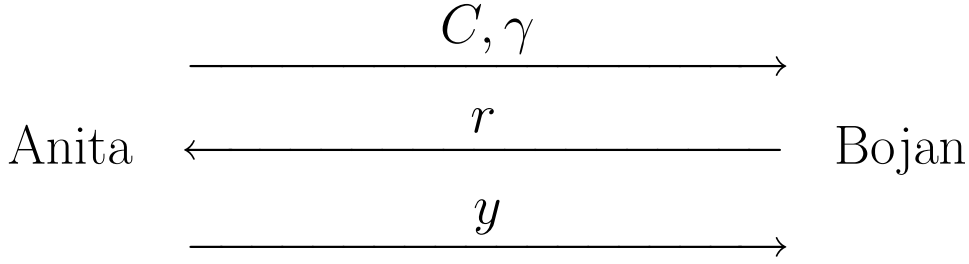
Ugotovili smo, da Anita zna potrditi svojo identiteto (*polnost*), vsak drug, ki zna to storiti z neznatno verjetnostjo (z uporabo identifikacijskega protokola) pa bodisi pozna zasebni a bodisi ga zna izračunati v polinomskem času (*uglašenost*).

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* $s = \mathrm{sig}_{\mathrm{TA}}(\mathrm{ID}(\mathrm{Anita}), v)$ *ter izroči Aniti potrdilo*

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

Bojan preveri Anitino identiteto:

1. *Anita si izbere naključni števili* $k_1, k_2 \in [0,..., q–1]$ *in izračuna* $\gamma = \alpha_1^{k_1}\,\alpha_2^{k_2} \bmod p$, *ki ga pošlje hkrati s svojim potrdilom* $C(\text{Anita})$ *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.

Izrek 2. *Če napadalka pozna število γ, za katero se zna z verjetnostjo $\varepsilon \geq 1/2^{t-1}$ predstaviti kot Anita, potem zna napadalka v polinomskem času izračunati taki števili b_1 in b_2, da je* $v \equiv \alpha_1^{-b_1}\,\alpha_2^{-b_2} \pmod{p}$.

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, y_2, r) in (z_1, z_2, s), da je $r \neq s$ in

$$\gamma \equiv \alpha_1^{y_1}\,\alpha_2^{y_2}\,v^r \equiv \alpha_1^{z_1}\,\alpha_2^{z_2}\,v^s \pmod{p}.$$

Definirajmo $b_i \equiv (y_i - z_i)(r-s)^{-1} \bmod q$ za $i = 1, 2$ in preverimo

$$v \equiv \alpha_1^{-b_1}\, \alpha_2^{-b_2}\, v^r \pmod{p}.$$

■

Izrek 3. *Če napadalka pozna število γ, za katero se zna z verjetnostjo $\varepsilon \geq 1/2^{t-1}$ predstaviti kot Anita, potem znata z verjetnostjo $1 - 1/q$ Anita in napadalka v polinomskem času izračunati* $\log_{\alpha_1} \alpha_2$.

Dokaz: Iz prejšnjega izreka sledi, da zna napadalka priti do števil b_1 in b_2, za kateri velja:

$$v \equiv \alpha_1^{-b_1} \alpha_2^{-b_2} \pmod{p}.$$

Anita izda vrednosti a_1 in a_2 tako, da imamo

$$v \equiv \alpha_1^{-a_1} \alpha_2^{-a_2} \pmod{p}$$

in od tod

$$\alpha_1^{a_1-b_1} \equiv \alpha_2^{b_2-a_2} \pmod{p}.$$

Če je $(a_1, a_2) \neq (b_1, b_2)$, potem obstaja
$(a_2 - b_2)^{-1} \bmod q$ in je

$$c = \log_{\alpha_1} \alpha_2 = (a_1 - b_1)(a_2 - b_2)^{-1} \bmod q.$$

Naj bo sedaj $(a_1, a_2) = (b_1, b_2)$. Pokazali bomo, da se lahko to zgodi le z zelo majhno verjetnostjo $1/q$, kar pomeni, da Anita in napadalka lahko skoraj vedno izračunata c.

Definirajmo množico vseh urejenih parov, ki bi bili lahko Anitini tajni eksponenti:

$$\mathcal{A} = \{(a_1', a_2') \in \mathbb{Z}_q \times \mathbb{Z}_q \mid \alpha_1^{-a_1'} \alpha_2^{-a_2'} \equiv \alpha_1^{-a_1} \alpha_2^{-a_2} \pmod{p}\}.$$

Potem ima množica $\mathcal{A}$ natanko q elementov, saj je

$$\mathcal{A} = \{(a_1 - c\theta, a_2 + \theta) \mid \theta \in \mathbb{Z}_q\}.$$

Po Okomotovemu protokolu si izbere Anita γ, napadalka si izbere r, Anita pa izračuna

$$\gamma \equiv \alpha_1^{y_1}\, \alpha_2^{y_2}\, v^r \pmod{p}$$

iz

$$y_i = k_i + a_i r \bmod q, \quad \text{za} \quad i = 1, 2,$$

kjer je

$$\gamma \equiv \alpha_1^{k_1}\, \alpha_2^{k_2} \pmod{p}$$

in ne izda k_1, k_2 (ne a_1 in a_2).

Ena četverica (γ, r, y_1, y_2) je navidez odvisna od urejenega para (a_1, a_2). Pokažimo, da bi lahko ta četverica bila generirana od poljubnega drugega para $(a'_1, a'_2) \in \mathcal{A}$, tj. $a'_1 = a_1 - c\theta$ in $a'_2 = a_2 + \theta$, $\theta \in [0..q-1]$:

$$y_1 = k_1 + a_1 r = k_1 + (a'_1 + c\theta)r = (k'_1 + rc\theta) + a'_1 r,$$

in

$$y_2 = k_2 + a_2 r = k_2 + (a'_2 - \theta)r = (k'_2 - r\theta) + a'_2 r,$$

Torej če bi začeli z (a'_1, a'_2) in bi si lahko izbrali $k'_1 = k_1 + r\,c\,\theta$ in $k'_2 = k_2 - r\,\theta$, bi dobili isti γ. ■

Guillou-Quisquaterjeva identifikacijska shema

Ta shema je zasnovana na sistemu RSA.

Agencija TA si izbere dve praštevili p in q ter izračuna $n = pq$. Slednje število je javno, medtem ko sta vrednosti p in q zasebni in izbrani tako, da je problem faktorizacije prezahteven.

TA si izbere še shemo za digitalni podpis, zgoščevalno funkcijo ter 40-bitno praštevilo b, ki bo služilo kot varnostni parameter in šifrirni eksponent.

Izdaja certifikata poteka na naslednji način:

1. *Agencija TA preveri Anitino identiteto in izda* $\mathrm{ID}(\mathrm{Anita})$.
2. *Anita si izbere zasebno naključno število* $u \in [0, \ldots, n-1]$, *izračuna* $v = u^{-b} \bmod n$ *in ga izroči agenciji TA.*
3. *Agencija TA izračuna* $s = \mathrm{sig}_{\mathrm{TA}}(\mathrm{ID}(\mathrm{Anita}), v)$ *ter izroči Aniti potrdilo*

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

GQ-identifikacija (Bojan preveri Anitino identiteto):

1. *Anita si izbere naključno število* $k \in [0,\ldots,n-1]$ *in izračuna* $\gamma = k^b \bmod n$*, ki ga da hkrati s svojim certifikatom* C(Anita) *Bojanu.*
2. *Bojan preveri podpis agencije TA, izbere naključno število* $r \in [1,\ldots,b-1]$ *in ga da Aniti.*
3. *Anita izračuna* $y = ku^r \bmod n$ *in ga da Bojanu.*
4. *Bojan preveri, ali je* $\gamma \equiv y^b v^r \pmod{n}$.

Prepričali se bomo, da je ta shema polna in uglašena, nihče pa ni uspel dokazati, da je tudi varna (tudi če bi privzel, da je krpitosistem RSA varen).

Polnost je očitna

$$v^r y^b \equiv (u^{-b})^r (ku^r)^b \equiv k^b \equiv \gamma \pmod{n},$$

za uglašenost pa privzamemo, da ni mogoče izračunati števila u iz v (le-tega smo dobili iz u z RSA šifriranjem).

Izrek. *Če napadalka pozna število γ, za katerega se zna z verjetnostjo $\varepsilon \geq 1/b$ predstaviti kot Anita, potem zna napadalka izračunati število u v polinomskem času.*

Dokaz: Za nek γ izračuna napadalka take vrednosti y_1, y_2, r_1, r_2, da je $r_1 > r_2$ in

$$\gamma \equiv v^{r_1} y_1^b \equiv v^{r_2} y_2^b \pmod{n}.$$

Potem velja

$$(y_2/y_1)^b \equiv v^{r_1 - r_2} \pmod{n}$$

in je $r_1 - r_2 < b$, tako da lahko izračunamo $t = (r_1 - r_2)^{-1} \bmod b$ z razširjenim Evklidovim algoritmom. Od tod dobimo tak s, da je $(r_1 - r_2)t = sb + 1$ in

$$(y_2/y_1)^{bt} \equiv v^{(r_1 - r_2)t} \equiv v^{sb+1} \pmod{n}, \quad \text{ali}$$

$$v \equiv (y_2/y_1)^{bt} v^{-sb} \pmod{n}.$$

Obe strani zgornje kongruence potenciramo na b^{-1} mod $\phi(n)$ ter ju nato invertiramo po modulu n

$$u \equiv (y_1/y_2)^t v^s \pmod{n} \quad \blacksquare$$

Popularne identifikacijske sheme so še Brickel in McCurleyjeva shema, Feige-Fiat-Shamirjeva shema in Shamirjeva shema s permutiranim jedrom. Zanjo je Shamir dokazal, da je varna s pomočjo metod za dokazovanje brez razkrivanja znanja.