# OPIS IN ANALIZA KRIPTOSISTEMA XTR

*Keršnik Jelena, podiplomski študij, FRI*

Mentor: Aleksandar Jurišič

CILJ: Opisati in analizirati kriptosistem XTR.

XTR je kriptosistem ki sodi v področje kriptografije z javnimi ključi. Razvila sta ga Arjen Lenstra in Eric Verheul. XTR oziroma ECSTR je kratica za Efficient and Compact Subgroup Trace Representation. Pri tej metodi predstavimo elemente podgrupe končnega obsega z uporabo sledi in vse izračune opravimo na podlagi tega zapisa. Z uporabo XTR lahko izvedemo vse algoritme, ki temeljijo na podlagi diskretnega logaritma. Varnost kriptosistema zagotavlja zahtevnost problema izračuna diskretnega logaritma v multiplikativni grupi končnega obsega. Prednosti so hitra izbira parametrov in ključa (hitreje kot pri RSA in kot pri ECC) , krajši ključ (glede na RSA in primerljiv z ECC), in hitrost (tretjina časa običajnih postopkov) ob enaki stopnji varnosti.

## 1. Uvod

Hitrost, učinkovitost in varnost so zagotovo lastnosti, ki jih mora imeti vsak dober kriptosistem. S spremembami zapisa (predstavitve) in izračunov je moč vplivati na vse te lastnosti in tako obstoječ kriptosistem izboljšati.

Pri številnih kriptosistemih uporabljamo generator multiplikativne grupe končnega obsega. XTR ni prvi sistem, ki namesto tega uporablja generator relativno majhne podgrupe, katere red je zadosti veliko praštevilo $q$. Njegovi »predhodniki« niso prinesli bistvenih prihrankov prostora in časa, ali pa so bili zapleteni in zato neprimerni za uporabo.

Kriptosistem XTR sta leta 2000 (Advances in Cryptology - CRYPTO 2000) predstavila Arjen Lenstra in Eric Varheul. XTR oziroma ECSTR je kratica za Efficient and Compact Subgroup Trace Representation. Gre za sistem kriptografije z javnimi ključi, ki temelji na multiplikativni podgrupi končnega obsega $\mathrm{GF}(p^6)$. Elementi iz $\mathrm{GF}(p^6)$ so predstavljeni (zapisani) s pomočjo sledi nad $\mathrm{GF}(p^2)$ in vsi izračuni so opravljeni v tem zapisu. Na ta način skrčimo zapis na približno tretjino bitov prvotne velikosti. Poleg tega dosežemo tudi hitrejše izračune.

XTR lahko implementiramo pri vseh kriptografskih algoritmih (kriptografija z javnim ključem), ki temeljijo na problemu diskretnega logaritma. Prednosti XTR so hitra izbira parametrov, krajši ključ ter prihranek prostora in časa pri izračunih. Po mnenju avtorjev bi lahko XTR postal alternativa algoritmom, ki temeljijo na eliptičnih krivuljah ali RSA. Možno ga je implementirati v aplikacijah kot so SSL/TLS (Secure Socket Layer, Transport Layer Security), pametne kartice, WAP/WTLS (Wireless Application Protocol, , Wireless Transport

Layer Security), IPSEC/IKE (Internet Protocol Security, Internet Key Exchange) in SET (Secure Electronic Transaction).

V drugem poglavju predstavimo matematično podlago XTR, definiramo sled, ter ocenimo zahtevnost računskih operacij v tem zapisu. V tretjem poglavju spregovorimo o izbiri parametrov (ključa), v četrtem predstavimo uporabo XTR v kriptografskih postopkih (Diffie-Hellmanov dogovor o ključi, ElGamal šifriranje/odšifriranje in digitalno podpisovanje). O varnosti XTR in primerjavi z drugimi metodami govori peto poglavje. Predstavljena je primerjava varnosti XTR s supersingularno eliptično krivuljo. Učinkovitost XTR primerjamo z RSA in eliptičnimi krivuljami. Šesto poglavje govori o napadih na XTR ter ukrepih za njihovo preprečevanje.

# 2. Matematična podlaga XTR

V tem poglavju opišemo, kako predstaviti poljubno potenco števila $g$ iz $\mathrm{GF}(p^6)$ z elementom iz $\mathrm{GF}(\mathrm{p}^2)$ in utemeljimo, da je ta zapis smiseln in učinkovit. Te potence lahko učinkovito izračunamo z aritmetiko v $\mathrm{GF}(\mathrm{p}^2)$. Na ta način zmanjšamo velikost zapisa na tretjino prvotnega, ker vsi izračunu potekajo v $\mathrm{GF}(\mathrm{p}^2)$, pa smo pridobili tudi na hitrosti. Predstavljena je aritmetika, ki je bila prvotno osnova XTR. Izboljšave najdemo v članku [3].

## 2.1. XTR supergurpa in XTR (pod)grupa

Naj bo $p$ takšno praštevilo, da je $p \equiv 2 \bmod 3$ in da ima $p^2 - p + 1$ prafaktor $q > 6$. Moč multiplikativne grupe $\mathrm{GF}(\mathrm{p}^6)^*$ je $p^6 - 1$. Ker $p^2 - p + 1$ deli $p^6 - 1$, obstaja njena podgrupa z močjo $p^2 - p + 1$. To je XTR supergrupa. Ker pa $q$ deli $p^2 - p + 1$, obstaja podgrupa XTR supergrupe z močjo $q$, generirana z elementom $g \in \mathrm{GF}(p^6)$. To je XTR (pod)grupa. Ker $q$ deli $p^6 - 1$, in ne deli $p^s - 1$, za $s = 1, 2, 3$, je $\mathrm{GF}(p^6)$ najmanjši obseg, ki vsebuje XTR grupo. Ob ustrezni izbiri parametrov $p$ in $q$ je torej problem diskretnega logaritma v XTR grupi enako zahteven kot v $\mathrm{GF}(p^6)$.

## 2.2. Definicija sledi

Generator XTR grupe $g$, je element $\mathrm{GF}(p^6)$. Njemu konjugirani elementi nad $\mathrm{GF}(p^2)$ so $g$, $g^{p^2}$ in $g^{p^4}$. Sled elementa $g$ je definirana kot vsota njemu konjugiranih elementov:

$$Tr(g) = g + g^{p^2} + g^{p^4}$$

Ker red elementa g deli moč $\mathrm{GF}(p^6)^*$, je $p^6 \equiv 1 (\bmod q)$. Od tod $(Tr(g))^{p^2} = Tr(g)$ in zato $Tr(g) \in \mathrm{GF}(p^2)$.

Spomnimo se, da red elementa $g$ deli $p^2-p+1$. Ker je $p^2 \equiv p-1\left(\mathrm{mod}\left(p^2-p+1\right)\right)$ in je $p^4 \equiv (p-1)^2 \equiv p^2-2p+1 \equiv p-1-2p+1 \equiv -p\left(\mathrm{mod}\left(p^2-p+1\right)\right)$ je $g^{p^2}=g^{p-1}$ in $g^{p^4}=g^{-p}$. Sled lahko zato zapišemo drugače:

$$Tr(g)=g+g^{p-1}+g^{-p}$$

Upoštevajmo, da je $p \cdot p^{p-1} \cdot p^{-p}=1$ in primerjajmo koeficiente polinomov $(x-g)\left(x-g^{p-1}\right)\left(x-g^{-p}\right)$ in $x^3-Tr(g)x^2+\left(Tr(g)\right)^p x-1$ iz $\mathrm{GF}\left(p^2\right)[x]$. Ugotovimo, da sta enaka. Gre za minimalni polinom elementa $g$ nad $\mathrm{GF}\left(p^2\right)$. Polinom, oziroma konjugirani elementi so natanko določeni s sledjo $Tr(g)$. Ker pa je minimalni polinom enolično določen s $g$, je z $g$ enolično določena tudi sled.

Enaka ugotovitev velja za poljubno potenco $g^n$, kjer je $n \in \mathbb{N}$. Konjugirani elementi h $g^n$ so ravno ničle polinoma $x^3-Tr\left(g^n\right)x^2+\left(Tr\left(g^n\right)\right)^p x-1$, ki pa je enolično določen s sledjo $Tr\left(g^n\right)$, ki je element $\mathrm{GF}\left(p^2\right)$. V poglavju 2.5 omenimo algoritem, s katerim na podlagi $Tr(g)$ hitro izračunamo $Tr\left(g^n\right)$. Ta izračun je hitrejši od računanja potence $g^n$.

Predstavitev s sledmi nam prihrani prostor in čas, odrečemo pa se razlikovanju med konjugiranimi elementi. Generatorja $g$ nam ni potrebno iskati, potrebujemo le njegovo sled ter sledi njegovih potenc. Ker so sledi iz $\mathrm{GF}\left(p^2\right)$, potekajo vsi izračuni v tem obsegu.

## 2.3. Aritmetične operacije v $\mathrm{GF}\left(\mathrm{p}^2\right)$ in njihova zahtevnost

V tem podpoglavju predstavimo zapis elementov iz $\mathrm{GF}\left(\mathrm{p}^2\right)$ ter utemeljimo zahtevnost računskih operacij v tem zapisu.

Ker je $p \equiv 2 \bmod 3$, je $p \bmod 3$ generator $\mathrm{GF}(3)^*$. Ničli $\alpha$ in $\alpha^p$ polinoma $\left(x^3-1\right)/(x-1)=x^2+x+1$ , ki je nerazcepen nad $\mathrm{GF}(p)$, predstavljata optimalno normalno bazo za $\mathrm{GF}\left(\mathrm{p}^2\right)$ nad $\mathrm{GF}(p)$. Ker je $\alpha^i=a^{i \bmod 3}$, lahko element $x$ iz $\mathrm{GF}\left(\mathrm{p}^2\right)$ zapišemo $x=x_1\alpha+x_2\alpha^p=x_1\alpha+x_2\alpha^2$, kjer sta $x_1, x_2 \in \mathrm{GF}(p)$.

**Lema 2.31:** Naj bodo $x, y, z \in \mathrm{GF}\left(p^2\right)$, kjer je $p$ praštevilo in $p \equiv 2 \bmod 3$. Če ne upoštevamo seštevanja in odštevanja v $\mathrm{GF}(p)$, je

i. potenciranje $x^p$ zastonj
ii. kvadriraje $x^2$ opravljeno za ceno dveh množenj v $\mathrm{GF}(p)$
iii. množenje $xy$ opravljeno za ceno treh množen v $\mathrm{GF}(p)$

iv. izračun $xz - yz^p$ opravljen za ceno štirih množenj v $\mathrm{GF}(p)$

**Dokaz:** Opremo se na zapis elementov, ki je opredeljen v prvem odstavku tega poglavja. $x^p = \left(x_1\alpha + x_2\alpha^2\right)^p = x_1{}^p\alpha^p + x_2{}^p\alpha^{2p}$. Zadnja enakost velja, ker vsi členi v razvoju binoma $(a+b)^p$, razen $a^p$ in $b^p$, vsebujejo binomski koeficient $\binom{p}{r} = \frac{p!}{r!(p-r)!}$, ki pa je deljiv s $p$ in zato enak 0 po modulu $p$. Upoštevamo, da je $2p \equiv 1 (\mathrm{mod}\, 3)$ ter da $x_1, x_2 \in \mathrm{GF}(p)$ in dobimo, da je $x^p = x_1\alpha^2 + x_2\alpha = x_2\alpha + x_1\alpha^2$. Iz tega zapisa je razvidno število množenj v $\mathrm{GF}(p)$. $x^2 = \left(x_1\alpha + x_2\alpha^2\right)^2 = x_1{}^2\alpha^2 + 2x_1x_2\alpha^3 + x_2{}^2\alpha^4$ Upoštevamo, da je $\alpha^i = a^{i \bmod 3}$ in dobimo $x^2 = x_1{}^2\alpha^2 + 2x_1x_2 + x_2{}^2\alpha$. Ker je $\alpha^2 + \alpha + 1 = 0$, je $1 = -\alpha^2 - \alpha$ in zato $x^2 = x_1{}^2\alpha^2 + 2x_1x_2\left(-\alpha^2 - \alpha\right) + x_2{}^2\alpha = x_2\left(x_2 - 2x_1\right)\alpha + x_1\left(x_1 - 2x_2\right)\alpha^2$. Iz tega zapisa je razvidno število množenj v $\mathrm{GF}(p)$. Za produkt zapišemo $xy = \left(x_2y_2 - x_1y_2 - x_2y_1\right)\alpha + \left(x_1y_1 - x_1y_2 - x_2y_1\right)\alpha^2$. Sedaj upoštevamo še, da je $x_1y_2 + x_2y_1 = \left(x_1 + x_2\right)\left(y_1 + y_2\right) - x_1y_1 - x_2y_2$. Izračunati moramo le produkte $\left(x_1 + x_2\right)\left(y_1 + y_2\right)$, $x_1y_1$ in $x_2y_2$. Podoben pristop uporabimo pri dokazu zadnje alineje in zapišemo

$$xz - yz^p = \left(z_1\left(y_1 - x_2 - y_2\right) + z_2\left(x_2 - x_1 + y_2\right)\right)\alpha + \left(z_1\left(x_1 - x_2 + y_1\right) + z_2\left(y_2 - x_1 - y_1\right)\right)\alpha^2$$

## 2.4. Zahtevnost aritmetičnih operacij v $\mathrm{GF}\left(\mathrm{p}^6\right)$

**Lema 2.4.1:** Naj bodo $x, y, z \in \mathrm{GF}\left(p^6\right)$, kjer je $p$ praštevilo in $p \equiv 2 \bmod 3$. Če ne upoštevamo seštevanj in odštevanj, je

i. kvadriranje $x^2$ opravljeno za ceno 14,4 množenj v $\mathrm{GF}(p)$,
ii. računanje $x^a$ za ceno $23{,}4\log_2(a)$ množenj v $\mathrm{GF}(p)$,
iii. množenje $xy$ za ceno 18 množenj v $\mathrm{GF}(p)$,
iv. izračun $x^a y^b$ pa za ceno $27{,}9 \cdot \log_2(\max(a,b))$ množenj v $\mathrm{GF}(p)$.

## 2.5. Računanje sledi potenc

Trojico $S_n\left(Tr(g)\right) = \left(Tr\left(g^{n-1}\right), Tr\left(g^n\right), Tr\left(g^{n+1}\right)\right)$ lahko pri dani $Tr(g)$ izračunamo z $8\log_2(n \bmod q)$ množenji v $\mathrm{GF}(p)$ ([1], algoritem 2.3.7). Računanje $g^n$ pri danem g zahteva predvidoma $23{,}4 \cdot \log_2(q)$ množenj . Računanje $Tr\left(g^n\right)$ je skoraj trikrat hitrejše od računanja $g^n$.

Računanje $Tr\left(g^a g^{kb}\right)$ je 1,75 krat hitrejše od računanja $g^a g^{kb}$. Z algoritmom 2.4.8 v [1] lahko $Tr\left(g^a g^{kb}\right)$ izračunamo za ceno $16 \cdot \log_2(q)$ množenj v $\mathrm{GF}(p)$, pri navadnem zapisu pa računanje $g^a g^{kb}$ zahteva $27{,}9 \cdot \log_2(q)$ množenj.

# 3. Izbira parametrov in ključa

Javni ključ pri kriptosistemu XTR vsebuje vsaj $p$, $q$ in $Tr(g)$, po potrebi pa tudi eno, dva ali vse tri od števil $Tr\left(g^k\right)$, $Tr\left(g^{k-1}\right)$ in $Tr\left(g^{k+1}\right)$ kjer je $k$ skriven. Obstajajo algoritmi, ki izračunajo enega od trojice $Tr\left(g^{k-1}\right)$, $Tr(g)$, $Tr\left(g^{k+1}\right)$ iz preostalih dveh, ali celo dva iz enega. V tem poglavju opišemo, kako izbrati parametre ključa in na kaj moramo biti pri tem pozorni.

## 3.1. Izbira p in q

Parametra $p$ in $q$ morata ustrezati pogojem, navedenim v poglavju 2.1: $p \equiv 2 \pmod 3$ in $q \mid p^2 - p + 1$. Parameter $p$ mora biti izbran tako, da varianta sita numeričnih polj z uporabo diskretnega logaritma ne bo učinkovita na $\mathrm{GF}\left(p^6\right)$, parameter $q$ pa mora biti izbran tako, da Pollardova $\rho$ metoda ne bo učinkovita na XTR podgrupi. Če izberemo $p$ velikosti reda 170 bitov, $q$ pa 160 bitov, dosežemo varnost primerljivo s 1024 bitnim RSA. Poleg tega ni priporočljivo, da bi bil parameter $p$ krajši kot $q$.

Obstaja več metod (algoritmov) za določitev $p$ in $q$. Pri izboru metode moramo pretehtati razmerje med hitrostjo metode in ranljivostjo sistema. Algoritmi za izbor parametrov so opisan v [2] poglavje 3 in v [7] poglavje 4. V [2] najdemo štiri algoritme, ki se razlikujejo po enostavnosti oziroma hitrosti, s parametri, ki jih vrnejo, pa dosežemo različne stopnje varnosti. Od izbora parametrov je odvisna tudi hitrost in enostavnost aritmetike, ki jo uporabljamo v XTR. Parametre je vsekakor treba izbrati tako, da najdemo kompromis med hitrostjo in varnostjo.

## 3.2. Določitev $Tr(g)$

Ko izberemo $p$ in $q$, je potrebno določiti še $Tr(g)$. Tudi za to je na razpolago več algoritmov, ki se razlikujejo predvsem v hitrost (glej [2] poglavje 3)

Ali je dani element $c$ iz $\mathrm{GF}\left(p^2\right)$ sled generatorja XTR grupe preverimo tako, da preverimo nerazcepnost polinoma $x^3 - cx^2 + c^p x - 1$. Za naključno izbrani element c iz $\mathrm{GF}\left(p^2\right)$ je verjetnost, da je polinom nerazcepen, približno $1/3$.

Potrebnih je $\frac{q}{q-1}\left(2,7\log_2 p+8\log_2\left(\frac{p^2-p+1}{q}\right)\right)$ množenj, da izberemo element iz $\mathrm{GF}(p^2)$, ki je sled generatorja XTR grupe. Generatorja dejansko ne potrebujemo. Če je moč XTR grupe blizu moči XTR supergrupe, je ugotavljanje pripadnosti XTR grupi učinkovito, to pa je povezano s preprečevanjem napadov (glej poglavje 6).

# 4. Uporaba XTR

XTR lahko uporabimo pri Diffie Hellmanovem dogovoruo gljuču, Elgamalovem šifriranju/odšifriranju, preprečevanju tajenja, digitalnem podpisovanju....V tem poglavju prikažemo XTR Diffie-Hellmanov dogovor o ključu, XTR ElGamal šifriranje/odšifriranje ter uporabo XTR pri digitalnem podpisovanju.

## *4.1. XTR Diffie-Hellman dogovor o ključu*

Anita in Bojan poznata javni ključ (p, q in $Tr(g)$), dogovoriti pa se želita o tajnem ključu K. Dogovarjanje poteka po nezavarovanem kanalu.

1. Anita si izbere naključno celo število $a$ $(1<a<q-2)$ ter izračuna trojico $S_a(Tr(g))=\left(Tr(g^{a-1}),Tr(g^a),Tr(g^{a+1})\right)$.
2. Anita pošlje Bojanu $Tr(g^a)$.
3. Bojan si izbere naključno celo število $b$ $(1<b<q-2)$ ter izračuna trojico $S_b(Tr(g))=\left(Tr(g^{b-1}),Tr(g^b),Tr(g^{b+1})\right)$
4. Bojan pošlje Aniti $Tr(g^b)$
5. Anita izračuna $S_a\left(Tr(g^b)\right)=\left(Tr(g^{(a-1)b}),Tr(g^{ab}),Tr(g^{(a+1)b})\right)$ in določi $K$ na podlagi $Tr(g^{ab})$
6. Bojan izračuna $S_b\left(Tr(g^a)\right)=\left(Tr(g^{(b-1)a}),Tr(g^{ab}),Tr(g^{(b+1)a})\right)$ in določi $K$ na podlagi $Tr(g^{ab})$

XTR Diffie-Hellmanov dogovor o ključu je trikrat hitrejši od običajnega, porabi pa tretjino običajnega prostora.

## *4.2. XTR ElGamal šifriranje/odšifriranje*

Anita in Bojan poznata javni ključ (p, q in $Tr(g)$).

1. Anita si izbere naključno celo število $k$ ter objavi $Tr(g^k)$.

2. Bojan si izbere naključno celo število $b$ ($2 \le b \le q-3$) ter izračuna trojici
   $S_b(Tr(g)) = (Tr(g^{b-1}), Tr(g^b), Tr(g^{b+1}))$ ter
   $S_b(Tr(g^k)) = (Tr(g^{(b-1)k}), Tr(g^{bk}), Tr(g^{(b+1)k}))$
3. Bojan na podlagi $Tr(g^{bk})$ določi simetrični ključ $K$ za enkripcijo, ter z dogovorjenim simetričnim algoritmom šifrira sporočilo $M$ v tajnopis $C$
4. Aniti pošlje par $(Tr(g^b), E)$
5. Anita izračuna $S_k(Tr(g^b)) = (Tr(g^{(k-1)b}), Tr(g^{bk}), Tr(g^{(k+1)b}))$ in določi simetrični ključ $K$ na podlagi $Tr(g^{bk})$
6. Anita z dogovorjenim simetričnim algoritmom in ključem $K$ odšifrira tajnopis $C$ nazaj v $M$

Prostorska in časovna zahtevnost XTR ElGamal šifriranja/odšifriranja je tretjina zahtevnosti običajnega postopka.

## 4.3. XTR DSA podpisovanje

Dolžina $q$ naj bo 160 bitov, toliko kot je dolžina zgostitve pri SHA-1. Anitin javni ključ je (p, q, $Tr(g)$, $Tr(g^k)$), kjer je $k$ skrivno število, ki ga pozna le Anita. Predpostavljamo, da za overjanje podpisa poznamo tudi $Tr(g^{k-1})$ in $Tr(g^{k+1})$. Oboje lahko vključimo v ključ in povečamo obseg prenosa podatkov, ali pa ju ob overjanju izračunamo iz $Tr(g^k)$, kar zahteva nekaj dodatnega računanja in ustrezno izbiro ključa $k$ (glej podpoglavje 5.5 v [2] ).

V resnici so na razpolago tri možnosti XTR javnega ključa. Ključ lahko vsebuje eno, dve ali vse tri vrednosti $Tr(g^k)$, $Tr(g^{k-1})$ in $Tr(g^{k+1})$. V nekaterih primerih, kot je na primer izdajanje certifikata s strani certifikatne agencije (CA), se lahko zahteva, da tretja stran (CA) lahko preveri pravilnosti teh komponent. Metoda za overjanje je opisana v [2], poglavje 6, in v članku [7] poglavje 5.

### Generiranje podpisa

1. Anita si izbere naključno celo število $u$ ($2 \le u \le q-3$) ter izračuna $S_u(Tr(g)) = (Tr(g^{u-1}), Tr(g^u), Tr(g^{u+1}))$.
2. Anita zapiše $Tr(g^u) = x_1\alpha + x_2\alpha^2$ ter izračuna $r = (x_1 + p \cdot x_2) \bmod g$. Če je $r = 0$ se vrne na 1.
3. Anita izračuna $u^{-1} \bmod q$
4. Anita izračuna zgostitev (SHA-1) $h$ zapisa $M$
5. Anita izračuna $s = u^{-1}(h + k \cdot r) \bmod q$. Če je $s = 0$ se vrne na 1.
6. Anitin podpis za $M$ je par $(r, s)$

### Overjanje podpisa

Predpostavljamo, da Anitin javni ključ vsebuje trojico $S_k\left(Tr(g)\right)$

1. Bojan preveri, če je $1 \le r, s \le q-1$
2. Bojan izračuna $w = s^{-1} \pmod q$
3. Bojan izračuna zgostitev $h$ zapisa $M$
4. Bojan izračuna $u_1 = w \cdot h \pmod q$ in $u_2 = r \cdot w \pmod q$
5. Bojan izračuna $v_0 = Tr\left(g^{u_1} \cdot g^{k \cdot u_2}\right)$, kar je enako $Tr\left(g^u\right)$
6. Bojan zapiše $v_0 = z_1\alpha + z_2\alpha^2$ ter izračuna $v = \left(z_1 + p \cdot z_2\right) \bmod q$
7. Bojan sprejme podpis če in samo če je $v = r$

Če je $(r,s)$ veljaven podpis, sta veljavna tudi podpisa $\left(r, s \cdot p^2 \pmod q\right)$ in $\left(r, s \cdot p^4 \pmod q\right)$. Temu se izognemo z izbiro podpisa, kjer je število $s \cdot p^{2i} \pmod q$, $i = 0,1,2$, najmanjše. Pri overjanju pa moramo preveriti tudi ta pogoj.

Pri algoritmih za overjanje podpisa računamo $Tr\left(g^a g^{bk}\right)$ kar je možno opraviti 1,75 krat hitreje kot izračunati produkt $g^a g^{bk}$.

## 5. Varnost XTR in primerjava z drugimi metodami

V tem poglavju spregovorimo o varnosti XTR, varnost XTR primerjamo s supersingularno eliptično krivuljo in sklenemo s primerjavo učinkovitosti XTR z RSA in ECC.

Ker parametre izberemo tako, da je $\mathrm{GF}\left(p^6\right)$ najmanjši obseg, ki vsebuje XTR grupo, je problem diskretnega logaritma v grupi XTR enako zahteven kot problem diskretnega logaritma v $\mathrm{GF}\left(p^6\right)$. Enako velja za Diffie-Hellmanov problem in odločitveni Diffie Hellmanov problem. Pri prvem gre za izračun $Tr\left(g^{xy}\right)$ pri danih $Tr\left(g^x\right)$ in $Tr\left(g^y\right)$, pri drugem pa za ugotovitev, če so tri števila $a$, $b$ in $c$ v zvezi $a = Tr\left(g^x\right)$, $b = Tr\left(g^y\right)$ in $c = Tr\left(g^{xy}\right)$.

Ker lahko algoritme za problem diskretnega logaritma (ali Diffie Hellmanov problem ali odločitveni Diffie Hellmanov problem) pretvorimo v algoritem za ustrezni XTR postopek in obratno, je odkrivanje (majhnega) ključa pri XTR Diffie Hellmanovem algoritmu prav tako zahtevno kot odkrivanje ključa pri običajnem Diffie Hellmanovem algoritmu.

XTR je vsaj tako varen kot RSA oziroma kriptosistem c eliptičnimi krivuljami.

## 5.1. XTR in supersingularna eliptična krivulja

Število točk nad $GF(p^2)$ na supersingularni eliptični krivulji (skupaj s točko neskončno) nad $GF(p^2)$ je $p^2 - p + 1$, kar je ravno moč XTR supergrupe. Obstajajo injektivni homomorfizmi takšnih krivulj na XTR supergrupo, ki so izračunljivi v polinomskem času. Znani so pod imenom MOV vložitve.

Takoj po predstavitvi XTR je bila postavljena domneva, da bi bili lahko inverzi teh homomorfizmov prav tako izračunljivi v polinomskem času. To bi pomenilo, da je XTR podgrupa le primer podgrupe supersingularne eliptične krivulje in bi napad na supersingularno eliptično krivuljo pomenil napad na XTR . Varnost XTR tako ne bi bila boljša od supersingularne ECC.

Resničnost te domneve bi pomenila še, da sta tudi Diffie-Hellmanov problem v XTR podgrupi in Diffie-Hellmanov problem v grupi točk reda $q$ na supersingularni eliptični krivulji učinkovito izračunljiva. Da zadnje ni res, je Verhuel potrdil v članku [6] V istem članku je dokazal, da je odločitveni Diffie Hellmanov problem na supersingularni eliptični krivulji učinkovit, na XTR grupi pa ne.

## 5.2. Primerjava XTR z ECC in RSA

XTR naj bi bil boljši od drugih kriptosistemov tudi iz drugih razlogov. Odlikujeta ga hitrejša izbira parametrov in ključa kot pri RSA in ECC . Zaradi enostavnega generiranje ključa, imajo vsi uporabniki lahko svoj javni ključ in ga ne delijo z ostalimi uporabniki sistema. Ključ je krajši kot pri RSA in primerljiv z ECC. Dolžina zapisa $p$ približno 170 bitov, da varnost ekvivalentno 1024 bitnem RSA (dolžina zapisa q naj bo približno 160 bitov).

Zaradi zapisov v $GF(p^2)$ so izračuni trikrat hitrejši od običajnih, stopnja varnosti pa je enaka. Odkrivanje ključa v XTR- Diffie Hellmanovem postopku je enako zahtevno kot odkrivanje ključa v (običajnem) Diffie Hellmanovem postopku.

Tabeli prikazujeta primerjavo med RSA in XTR. Povzeti sta iz [1]. Prva tabela prikazuje rezultate meritev časov potrebnih za izbiro ključa, šifriranje in odšifriranje pri RSA in XTR

| | Izbira ključa | Šifriranje (overjanje) | Odšifriranje (podpisovanje) |
|---|---|---|---|
| 1020 bit RSA | 1224 ms | 5 ms | 40 ms |
| 170 bit XTR | 73 ms | 23 ms | 11ms |

Druga tabela prikazuje teoretične ugotovitve o številu množenj, potrebnih za posamezne operacije.

| | Šifriranje | Odšifriranje | Podpisovanje | Overjanje | DH hitrost | DH velikost |
|---|---|---|---|---|---|---|
| ECC | 3400 | 1921 | 1700 | 3400 množenj | 3842 | 171 bitov |
| XTR | 2720 | 1360 | 1360 | 2720 množenj | 2720 | 340 bitov |

# 6. Napadi na XTR

Poglavje govori o možnih napadih na XTR ter ukrepih za njihovo preprečevanje.

Med predvidenimi napadi na XTR zasledimo Pollardovo $\rho$ metodo, napad na multiplikativno grupo, napad na podgrupo, različne oblike napada z uporabo stranskega kanala (Side channel Attack) kot na primer napad s pomočjo enostavne analize moči - SPA (Simple Power Analysis), napad s trki (Collision attack)...

Pri napadu na multiplikativno grupo gre za varianto sita numeričnih polj z uporabo disktretnega logaritma v $\mathrm{GF}\left(p^6\right)^*$.

Varnost kriptografskih protokolov se zmanjša, če so izmenjani elementi določenih grup, prejemniki pa ne preverijo pripadnosti elementov tem grupam. Na tem temelji tudi napad na XTR podgrupo. Napad na podgrupo je neučinkovit, če je $\frac{p^2-p+1}{q}$ majhno število, ali pa $\frac{p^2-p+1}{q}$ majhen večkratnik praštevila enakega velikostnega reda kot $q$. Kljub temu pa je potrebno preverjati pripadnost podgrupi. Pripadnost supergrupi se ugotovi z nekaj več kot $1,8\log_2 p$ množenji ali celo manj (a z več izmenjave podatkov). Preverjanje pripadnosti podgrupi pa je dražje, saj terja $8\log_2 q$ množenj v $\mathrm{GF}(p)$. Izkaže se, da je smiselno izbrati moč podgrupe velikostnega reda blizu velikostnega reda moči supergrupe.

Možni napadi z uporabo stranskega kanala so data bit attack, address bit attack, doubling attack , simple power analysis (SPA) ... Potrebnih je $U^{1,25}$ poskusov z SPA, kjer je $U=\max(a,b)$, da odkrijemo oba eksponenta a in b. Obstaja zveza med krivuljo napetosti in vrstnim redom računskih operacij. Napad otežimo s premišljeno izbiro vrstnega reda operacij. Več o napadih je zapisano v članku [4] , o SPA pa v članku [5].

O napadu s trki govori članek [8]

# 7. Sklep

XTR je relativno nov kriptosistem, ki še ni standardiziran. Glede na to, da potrebuje vsaka novost v kriptografskem svetu svoj čas, da se preveri njena učinkovitost in zanesljivost, to ni presenetljivo. Bistvo XTR je, da prinese le drugačen zapis, zaradi česar se olajšajo izračuni ter pridobi na prostoru in času ob enaki stopnji varnosti.

# 8. Literatura

[1] A.K. Lenstra, E.R. Verheul, The XTR public key system,
Advances in Cryptology - Crypto 2000 Proceedings, volume 1880 of LNCS, strani 1–19.
Springer-Verlag, 2000.

[2] A.K. Lenstra, E.R. Verheul, An overview of the XTR public key system, Proceedings of the Warsaw Conference on Public-Key cryptography and computational number theory, 2000.
[3]M. Stam, A. Lenstra, Speeding Up XTR, Advances in Cryptology (ASIACRYPT 2001), Springer LNCS 2248, 125-143, 2001.
[4] D. Han, J. Lim, K. Sakurai, On Security of XTR Public Key Cryptosystems Against Side Channel Attacks, ACISP 2004: 454-465
[5] J. Chung, A. Hasan, Security Analysis of XTR Exponentiation Algorithms Against Simple Power Analysis Attack, SecUbiq 2005
[6] E. R. Verheul, Evidence that XTR is more secure than supersingular elliptic curve cryptosystems, EUROCRYPT, vol. 2045.
[7] A.K. Lenstra, E.R. Verheul, Key Improvements to XTR, ASIACRYPT 2000: 220-233
[8] D.Han, T. Takagi, T. Kim, H. Kim, K. Chung, Collision Attack on XTR and a Countermeasure with a Fixed Pattern. EUC Workshops 2005: 864-873