Signal – kryptografia

Martin Stanek

2025

KI FMFI UK Bratislava

Úvod

- Signal – 70 miliónov používateľov v 2024
  - bezpečný prenos správ, súborov, video hovory a pod.
  - verejne známe protokoly a konštrukcie, využívané aj v iných aplikáciách
- špecifické požiadavky a obmedzenia, napr. asynchrónna komunikácia

Cieľ:

- predstaviť niektoré kryptografické konštrukcie a ich vlastnosti
  - protokol na dohodnutie spoločných kľúčov (X3DH)
  - využitie postkvantovej kryptografie (PQXDH)
  - algoritmus odvádzania kľúčov pre šifrovanie správ (dvojitá račňa)

Extended Triple Diffie–Hellman (X3DH)

X3DH

- Rozšírený trojitý Diffieho–Hellmanov protokol
  - dohodnutie spoločného kľúča medzi účastníkmi A a B, s využitím servera S
  - vzájomná autentizácia na základe publikovaných verejných kľúčov
  - asynchrónna komunikácia, teda B môže byť offline (ale S má jeho verejný kľúč)
  - *forward secrecy*, teda ochrana minulej komunikácie
  - minimálna dôvera v server S
  - hodnoverné popretie vzájomnej komunikácie, resp. jej obsahu
- kryptografické primitíva:
  - $\mathrm{DH}(\mathrm{pk}_1, \mathrm{pk}_2)$ – výsledok ECDH s verejnými parametrami (kľúčmi) pk_1 a pk_2
  - $\mathrm{Sig}(\mathrm{pk}, M)$ – XEdDSA podpis správy M, overiteľný verejným kľúčom pk
  - $\mathrm{KDF}(x)$ – funkcia na odvodenie kľúčov zo vstupu x (HKDF konštrukcia)

Role

- ***A***
  - chce poslať úvodnú správu *B* a vytvoriť spoločný kľúč pre vzájomnú komunikáciu
- ***B***
  - chce umožniť iným účastníkom vzájomnú komunikáciu s ním
  - môže byť offline, využíva služby servera
- **server *S***
  - odkladá správy pre *B*, ktorý si ich neskôr vyzdvihne
  - umožní *B* zverejniť informácie, ktoré je možné poskytnúť *A* a iným účastníkom

Kľúče v X3DH

- IK_A, IK_B – dlhodobý verejný kľúč zviazaný s identitou účastníka (*identity key*)
- EK_A – efemérny kľúč, generovaný A nanovo pre každý beh protokolu
- SPK_B – podpísaný „predkľúč“, periodicky menený (aktualizovaný)
- OPK_B – jednorazový „predkľúč“
- dlhodobé kľúče aj predkľúče sú publikované na S

Poznámka: Ku každému verejnému kľúču existuje zodpovedajúci súkromný kľúč.

1. *B* zverejní na *S* sadu verejných kľúčov
2. *A* získa od *S* sadu predkľúčov a pošle úvodnú správu *B*
3. *B* prijme a spracuje správu od *A*

Publikované kľúče (*B*)

- identita: IK_B
  - nemení sa
- podpísaný predkľúč: SPK_B, $Sig(IK_B, SPK_B)$
  - nové hodnoty napr. raz za týždeň, mesiac
- sada jednorazových predkľúčov: $OPK_B^1, OPK_B^2, \ldots$
  - môžu byť kedykoľvek doplnené novými kľúčmi

Iniciálna správa

1. *A* získa od *S* sadu kľúčov:
   - IK_B
   - SPK_B, $\mathrm{Sig}(\mathrm{IK}_B, \mathrm{SPK}_B)$
     - *A* overí podpis
   - voliteľne jeden z OPK_B
     - ak je k dispozícii
     - *S* tento kľúč následne vymaže
2. *A* vygeneruje efemérny pár kľúčov
   - EK_A je príslušný verejný kľúč
   - zmazaný po výpočte SK
3. Výpočet spoločného kľúča SK:

$$\mathrm{DH}_1 = \mathrm{DH}(\mathrm{IK}_A, \mathrm{SPK}_B)$$

$$\mathrm{DH}_2 = \mathrm{DH}(\mathrm{EK}_A, \mathrm{IK}_B)$$

$$\mathrm{DH}_3 = \mathrm{DH}(\mathrm{EK}_A, \mathrm{SPK}_B)$$

$$\mathrm{SK} = \mathrm{KDF}(\mathrm{DH}_1 \| \mathrm{DH}_2 \| \mathrm{DH}_3)$$

- ak je OPK_B k dispozícii:

$$\mathrm{DH}_4 = \mathrm{DH}(\mathrm{EK}_A, \mathrm{OPK}_B)$$

$$\mathrm{SK} = \mathrm{KDF}(\mathrm{DH}_1 \| \mathrm{DH}_2 \| \mathrm{DH}_3 \| \mathrm{DH}_4)$$

Iniciálna správa (pokr.)

4. Konštrukcia iniciálnej správy:
   - asociované dáta $\mathrm{AD} = \mathrm{IK}_A \parallel \mathrm{IK}_B$
   - obsah: IK_A, EK_A, identifikátory použitých predkľúčov, šifrový text
   - šifrový text:
     - AEAD s kľúčom SK, príp. odvodeným z SK; asociované dáta AD
     - šifrovaná správa je prvá správa v post-X3DH protokole

5. Príjem iniciálnej správy
   - *B* načíta príslušné súkromné kľúče
   - *B* vypočíta SK, dešifruje správu a overí AD
     - úspešne: vymaže použitý súkromný kľúč pre OPK_B
     - neúspešne: ukončí protokol a vymaže SK

Poznámky k bezpečnosti X3DH

Autentizácia

- DH_1 a DH_2 zodpovedné za autentizáciu
- overenie autentickosti IK_A a IK_B
  - rieši aplikácia, mimo záberu X3DH
  - Signal: „safety number“
  - v opačnom prípade: MITM útok

Význam OPK_B (ak sa nepoužije)

- *B* akceptuje opakovanú iniciálnu správu
- *B* odvodí rovnaké SK pre iný beh protokolu
- post-X3DH protokol **musí** znáhodniť šifrovací kľúč predtým, ako *B* pošle šifrované dáta

Význam podpisu SPK_B

- Čo ak SPK_B nie je podpísaný, nestačia DH_1 a DH_2 samotné?
- útočník podvrhne vlastnú sadu kľúčov
  - IK_B bude jediný autentický
  - neskôr, ak dôjde ku kompromitácii IK_B, dokáže vypočítať SK

Nahradiť DH_1 a DH_2 podpismi?

- sťažuje popretie vzájomnej komunikácie
- zhrošuje situáciu pri kompromitácii súkromných efemérnych kľúčov a predkľúčov

Poznámky k bezpečnosti X3DH (pokr.)

Kompromitácia súkromných kľúčov

- s.k. pre IK_X: falšovanie identity *X*
- použitie OPK_B; s.k. pre IK_B a predkľúče:
  - minulé SK nie sú kompromitované (s.k. pre OPK_B vymazaný)
- bez použitia OPK_B; s.k. pre IK_B a predkľúče kompromitované:
  - kompromitácia starých aj nových SK
  - častá zmena SPK_B, príp. račňa
- s.k. pre predkľúče *B*, útočník dokáže
  - predstierať inú identitu pred kompromitovaným *B*
  - pasívne rekonštruovať SK

Previazanie identity

- X3DH samotné nezväzuje identitu s IK_X
- *C* prezentuje identitu *B* ako svoju
  - iniciálnu správu od *A*, určenú jemu, prepošle *B*
- mimo záberu X3DH, riešenie v aplikácii:
  - obohatenie AD alebo odtlačku kľúča o dodatočné informácie
  - telefónne číslo, používateľské meno …

Post-Quantum Extended Diffie-Hellman (PQXDH)

PQXDH

- PQXDH: kombinácia X3DH s post-kvantovým KEM (Key Encapsulation Mechanism)
  - cieľ: ochrana pred útokmi typu „*pozbieraj teraz, dešifruj neskôr*“
  - zachovanie obvyklých vlastností: asynchrónna komunikácia, offline popretie komunikácie, …
  - autentickosť sa opiera o DH, nie je post-kvantová
  - neposkytuje ochranu pred aktívnym „kvantovým“ útočníkom
- nové kryptografické primitíva:
  - $(\mathrm{CT}, \mathrm{SS}) = \mathrm{PQKEM\text{-}ENC}(\mathrm{pk})$ – výsledok KEM s verejným kľúčom pk; šifrový text a spoločné tajomstvo/kľúč
  - $\mathrm{SS} = \mathrm{PQKEM\text{-}DEC}(\mathrm{pk}, \mathrm{CT})$ – „otvorenie“ CT s využitím súkromného kľúča prislúchajúcemu k pk

Publikované kľúče

- *B* publikuje na serveri kľúče ako predtým:
  - IK_B; SPK_B aj s podpisom; OPK_B^1, OPK_B^2, …
- nové verejné kľúče pre PQKEM:
  - PQSPK_B – podpísaný predkľúč „poslednej inštancie“
  - PQOPK_B^1, PQOPK_B^1, … – podpísané jednorazové predkľúče
  - podpisuje sa pomocou súkromného kľúča pre IK_B
- všetky kľúče majú jednoznačný identifikátor

Iniciálna správa v PQXDH

1. *A* získa od *S* sadu kľúčov:
   - všetky ako v X3DH protokole
   - jeden z PQOPK_B aj s podpisom
     - ak je k dispozícii
     - *S* tento kľúč následne vymaže
     - ak žiadny PQOPK_B nezostal, *S* pošle PQSPK_B
     - označme tento kľúč PQPK_B
   - *A* overí všetky podpisy
2. *A* vygeneruje
   - efemérny pár kľúčov ako v X3DH
   - $(\mathrm{CT}, \mathrm{SS}) = \mathrm{PQKEM\text{-}ENC}(\mathrm{PQPK}_B)$
3. Výpočet spoločného kľúča SK:
   - výpočet DH_1, DH_2, DH_3, príp. DH_4 ako predtým
   - ak OPK_B nebolo k dispozícii:

     $$\mathrm{SK} = \mathrm{KDF}(\mathrm{DH}_1 \| \mathrm{DH}_2 \| \mathrm{DH}_3 \| \mathrm{SS})$$

   - inak:

     $$\mathrm{SK} = \mathrm{KDF}(\mathrm{DH}_1 \| \mathrm{DH}_2 \| \mathrm{DH}_3 \| \mathrm{DH}_4 \| \mathrm{SS})$$

   - *A* vymaže súkromný efemérny kľúč, SS a výsledky DH výpočtov

Iniciálna správa v PQXDH (pokr.)

4. Konštrukcia iniciálnej správy:
   - len rozdiely oproti X3DH
   - asociované dáta AD zahŕňajú aj PQKP_B, ak tento nie je súčasťou CT
   - iniciálna správa obsahuje aj CT (*A* ho po odoslaní zmaže)
5. Príjem iniciálnej správy
   - *B* postupuje ako v X3DH, pričom vykoná aj príslušné KEM výpočty
     - $\mathrm{SS} = \mathrm{PQKEM\text{-}DEC}(\mathrm{PQPK}_B, \mathrm{CT})$
   - *B* vypočíta SK, dešifruje správu a overí AD
     - úspešne: vymaže použitý súkromný kľúč pre OPK_B a pre použitý PQOPK_B
     - neúspešne: ukončí protokol a vymaže SK

Poznámky k bezpečnosti PQXDH

- väčšina vlastností analogicky ako pri X3DH
- existuje formálna analýza PQXDH
  - symbolický model (ProVerif), výpočtový model (CrypoVerif)
- špecifikum pre generické KEM: KEM Re-encapsulation útok
  - ak je SS nezávislé na verejnom kľúči
  - kompromitácia PQPK, následne:
    - útočník vnúti PQPK iniciátorovi a získa SS
    - „prebalí“ SS pre *B* podľa skutočného, aktuálneho PQPK
    - z pohľadu *B* je všetko v poriadku (SK zachraňuje len X3DH časť protokolu)
  - ochrana:
    - Kyber KEM zahŕňa do výpočtu SS aj PQPK
    - vložiť PQPK do AD

Double Ratchet

KDF reťaz

- KDF je funkcia na odvodenie symetrických kľúčov, napr. HMAC, HKDF:
  - parametre: KDF kľúč, vstup
  - výstup: reťazec bitov požadovanej dĺžky
- KDF reťaz
  - výstup sú dva symetrické kľúče: nový kľúč pre KDF, výstupný kľúč
  - postup sa opakuje s novým vstupom a novým KDF kľúčom
  - výsledkom je postupnosť výstupných kľúčov

$$\mathrm{KDF}(k_1, \mathrm{input}_1) \Rightarrow k_2, \mathrm{output\ key}_1$$

$$\mathrm{KDF}(k_2, \mathrm{input}_2) \Rightarrow k_3, \mathrm{output\ key}_2 \quad \ldots$$

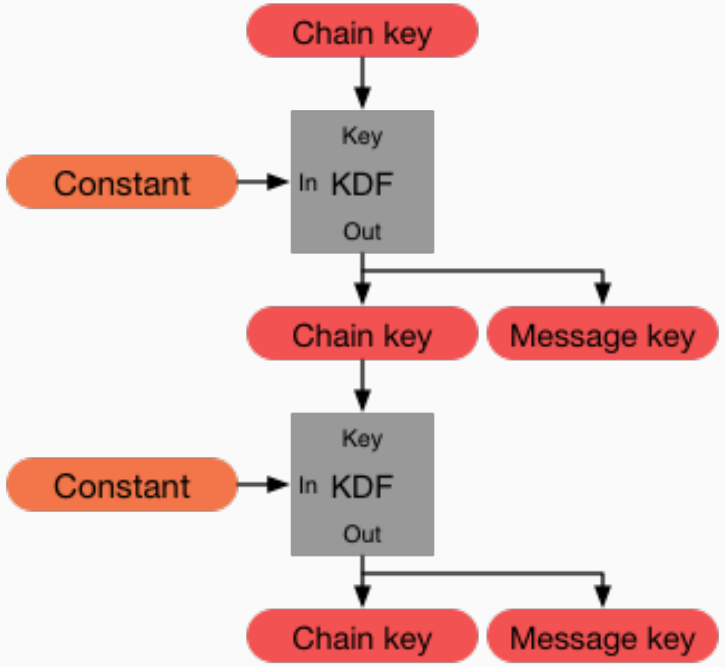
Vlastnosti KDF reťaze

- **odolnosť** - bez znalosti KDF kľúčov vyzerajú výstupné kľúče náhodne, aj keď útočník môže voliť vstup
- **dopredná bezpečnosť** – minulé výstupné kľúče vyzerajú náhodne, aj keď sa útočník dozvie aktuálny KDF kľúč
- **zotavenie po kompromitácii** – budúce výstupné kľúče vyzerajú náhodne, aj keď sa útočník dozvie KDF kľúč, pokiaľ majú vstupy dostatočnú entropiu

Hlavná myšlienka dvojitej račne

Račňa pre symetrické kľúče

- každá poslaná alebo prijatá správa má svoj vlastný symetrický kľúč
- kľúč pre správu je výstupným kľúčom KDF reťaze
- vstupy pre reťaz sú konštanty
- samostatná reťaz pre posielané správy a pre prijímané správy

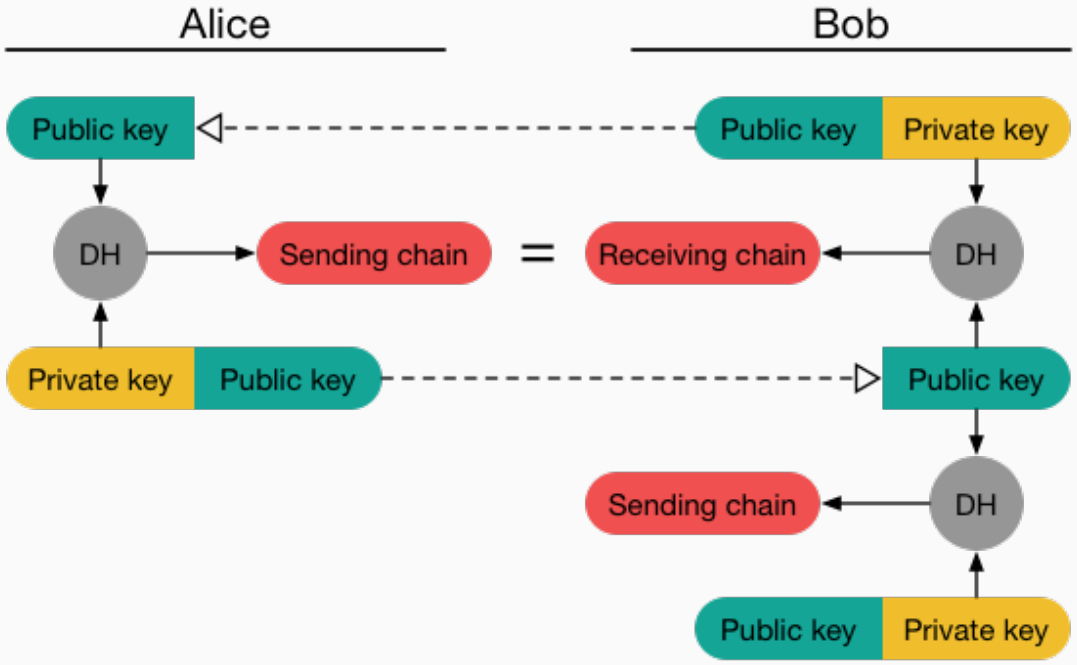


https://signal.org/docs/specifications/doubleratchet

Hlavná myšlienka dvojitej račne (pokr.)

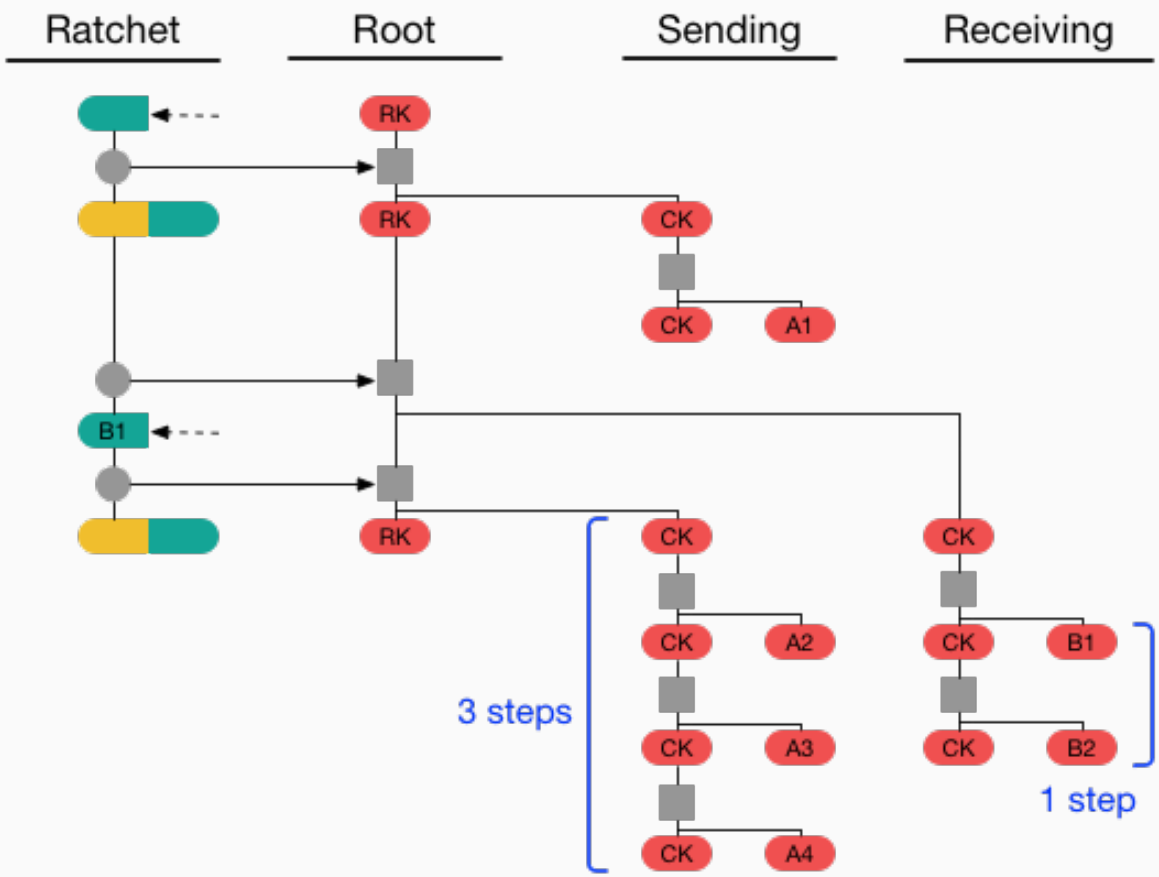
DH račňa (koreňová KDF reťaz)

- inicializovaná SK ako koreňovým kľúčom
- priebežné DH výmeny (aktuálny verejný kľúč súčasť hlavičky správy)
- spoločné tajomstvá ako vstupy do koreňovej reťaze
- výstupné kľúče sú KDF kľúčmi račnı́ pre symetrické kľúče



https://signal.org/docs/specifications/doubleratchet

Dvojitá račňa dokopy (pohľad jedného účastníka)



https://signal.org/docs/specifications/doubleratchet

Poznámky

- dvojitá račňa je v úlohe „post-X3DH“ protokolu
- podpísaný predkľúč SPK_B z X3DH slúži ako iniciálny kľúč B pre DH račňu
- potreba spravovať vnútorný stav
  - aktuálne hodnoty pre KDF reťaze
  - DH Kľúče
  - aktuálne čísla správ pre posielanie a príjem
  - počet správ v predchádzajúcej reťazi pre posielanie správ, …

Ako sa na Signal útočí v praxi

- veľmi ťažké: kryptoanalýza kryptografických algoritmov
- ťažké: útok na zariadenie samotné (telefón)
- realistické: využitie funkcií samotnej aplikácie + sociálne inžinierstvo:
  - QR kód s linkou na pridanie do skupiny (group chat)
  - QR kód na pripojenie ďalšieho zariadenia k telefónu (umožňuje súčasné používanie viacerých zariadení s jedným účtom)
  - škodlivý QR kód pripojí zariadenie útočníka
  - Signal aktualizovaný s cieľom sťažiť takéto útoky v budúcnosti
  - zdroj: Google Threat Intelligence Group (2025)

Úlohy

1. Popíšte dopad, keď útočník bude v X3DH cyklicky žiadať úvodnú sadu kľúčov účastníka B a navrhnite vhodné opatrenie na ošetrenie súvisiaceho rizika.
2. Ukážte, ako dokáže útočník so schopnosťou efektívne počítať diskrétny logaritmus na eliptických krivkách, predstierať identitu A v PQXDH protokole.
3. Popíšte dopad kompromitácie kľúča na šifrovanie správy, ktorý je odvodený v dvojitej račni.