


Säkra protokoll och procedurer

Daniel Bosk¹

Avdelningen för informations- och kommunikationssystem (IKS),
Mittuniversitetet, Sundsvall.

1st May 2017

¹Detta verk är tillgängliggjort under licensen Creative Commons Erkännande-DelaLika 2.5 Sverige (CC BY-SA 2.5 SE). För att se en sammanfattning och kopia av licenstexten besök URL

<http://creativecommons.org/licenses/by-sa/2.5/se/> 

1 Introduktion

- Vad är ett protokoll?
- Ibland blir det fel
- Autentisering

2 Formell notation

- Protokoll
- BAN-logik
- Automated Verification

3 Protokoll och attacker

- Enkel autentisering
- Challenge–response
- Miljöbyte
- Internetbanken och betalkort



Definition (Protokoll)

- Ett system består av en uppsättning principals.
- Ett protokoll är en uppsättning regler som styr hur dessa kommunicerar.

Example (Tentamen MIUN)

- 1 Tentamensvakten öppnar salen och ger varje tentand ett nummer.
- 2 Tentanden går in och sätter sig vid sin tilldelade plats.
- 3 Efter att tentan börjat jämför tentamensvakten tentandens legitimation och nummer.
- 4 Vid inlämning av skrivning jämförs legitimationen och numret.





Definition (Protokoll)

- Ett system består av en uppsättning principals.
- Ett protokoll är en uppsättning regler som styr hur dessa kommunicerar.

Example (Tentamen MIUN)

- 1 Tentamensvakten öppnar salen och ger varje tentand ett nummer.
- 2 Tentanden går in och sätter sig vid sin tilldelade plats.
- 3 Efter att tentan börjat jämför tentamensvakten tentandens legitimation och nummer.
- 4 Vid inlämning av skrivning jämförs legitimationen och numret.





Vad är ett protokoll?

Att tänka på

- Bör vara designade för att motstå attacker.
- Både oavsiktligt och avsiktligt brott mot protokollet.



Example (Beställa vin)

- 1 **Hovmästaren** visar vinlistan för värden.
- 2 **Värden** väljer vin, hovmästaren hämtar.
- 3 **Värden** provsmakar vin, det serveras till gästerna.

Egenskaper

Konfidentialitet Gästerna får ej veta priset.

Riktighet Hovmästaren kan inte byta ut vinet.

Oavvislighet Värden kan inte falskt klaga på vinet i efterhand.



Example (Beställa vin)

- 1 **Hovmästaren** visar vinlistan för värden.
- 2 **Värden** väljer vin, hovmästaren hämtar.
- 3 **Värden** provsmakar vin, det serveras till gästerna.

Egenskaper

Konfidentialitet Gästerna får ej veta priset.

Riktighet Hovmästaren kan inte byta ut vinet.

Oavvislighet Värden kan inte falskt klaga på vinet i efterhand.

Example (Tentamen MIUN)

- 1 Tentamensvakten öppnar salen och ger varje tentand ett nummer.
- 2 Tentanden går in och sätter sig vid sin tilldelade plats.
- 3 Efter att tentan börjat jämför tentamensvakten tentandens legitimation och nummer.
- 4 Vid inlämning av skrivning jämförs legitimationen och numret.

Egenskaper

Anonymitet Varje tentand förblir anonym.

Autenticitet Skrivningen är garanterat associerad med tentanden.

Example (Tentamen MIUN)

- 1 Tentamensvakten öppnar salen och ger varje tentand ett nummer.
- 2 Tentanden går in och sätter sig vid sin tilldelade plats.
- 3 Efter att tentan börjat jämför tentamensvakten tentandens legitimation och nummer.
- 4 Vid inlämning av skrivning jämförs legitimationen och numret.

Egenskaper

Anonymitet Varje tentand förblir anonym.

Autenticitet Skrivningen är garanterat associerad med tentanden.



Att tänka på

- Konstrueras utifrån grundläggande antaganden.
 - Exempelvis att kortägaren kan mata in PIN-koden direkt i terminalen.
- Analysera om hoten är rimliga.
- Analysera om protokollet hanterar dem.

Example (Tentamen KTH)

- 1 Tentamensvakten öppnar salen.
- 2 Tentanden går in och sätter sig.
- 3 Efter att tentan börjat undersöker tentamensvakten tentandens legitimation och antecknar plats i salen.
- 4 Tentanden lämnar in skrivningen.

Egenskaper

- Autentiseringen brister, tentanden kan skriva vilket namn och personnummer som helst på inlämnad tentamen.
- Ingen anonymitet.

Example (Tentamen KTH)

- 1 Tentamensvakten öppnar salen.
- 2 Tentanden går in och sätter sig.
- 3 Efter att tentan börjat undersöker tentamensvakten tentandens legitimation och antecknar plats i salen.
- 4 Tentanden lämnar in skrivningen.

Egenskaper

- Autentiseringen brister, tentanden kan skriva vilket namn och personnummer som helst på inlämnad tentamen.
- Ingen anonymitet.

Example (Autentisera uttag)

- Banker lagrade kontonummer på magnetremsan.
- PIN-koden skickades till centrala systemet för verifiering.

'Förbättring'

Kryptera PIN-koden och lagra den på magnetremsan så att uttagsautomaten kan verifiera den när den inte får kontakt med centrala systemet.

Exercise

Några problem?

Example (Autentisera uttag)

- Banker lagrade kontonummer på magnetremsan.
- PIN-koden skickades till centrala systemet för verifiering.

'Förbättring'

Kryptera PIN-koden och lagra den på magnetremsan så att uttagsautomaten kan verifiera den när den inte får kontakt med centrala systemet.

Exercise

Några problem?

Example (Autentisera uttag)

- Banker lagrade kontonummer på magnetremsan.
- PIN-koden skickades till centrala systemet för verifiering.

'Förbättring'

Kryptera PIN-koden och lagra den på magnetremsan så att uttagsautomaten kan verifiera den när den inte får kontakt med centrala systemet.

Exercise

Några problem?

Problem

- Jag kan byta ut kontonumret men inte ändra koden.
- Ändra kontonummer och använd min egen kod för att ta ut från annans konto.



- Lösenord och PIN-koder är fundamentala metoder för autentisering.
- Exempelvis 90-talets fjärrlås till bilen:
 - Nyckeln skickade serienumret.
 - Bilen kontrollerade allt den mottog och jämförde med sitt serienummer.
- Kunde angripas med en inspelningsattack.
 - Spela in allt som sänds (serienummer).
 - Spela upp serienummer för att låsa upp.



- Lösenord och PIN-koder är fundamentala metoder för autentisering.
- Exempelvis 90-talets fjärrlås till bilen:
 - Nyckeln skickade serienumret.
 - Bilen kontrollerade allt den mottog och jämförde med sitt serienummer.
- Kunde angripas med en inspelningsattack.
 - Spela in allt som sänds (serienummer).
 - Spela upp serienummer för att låsa upp.



- Lösenord och PIN-koder är fundamentala metoder för autentisering.
- Exempelvis 90-talets fjärrlås till bilen:
 - Nyckeln skickade serienumret.
 - Bilen kontrollerade allt den mottog och jämförde med sitt serienummer.
- Kunde angripas med en inspelningsattack.
 - Spela in allt som sänds (serienummer).
 - Spela upp serienummer för att låsa upp.



- Ibland kan ett enkelt lösenord vara rätt väg att gå.
- Matkuponger på lunchställen:
 - Papperslapp med serienummer.
 - Kan framställas i kopian.
- Hotet är inte värt högre säkerhet.



- Ibland kan ett enkelt lösenord vara rätt väg att gå.
- Matkuponger på lunchställen:
 - Papperslapp med serienummer.
 - Kan framställas i kopian.
- Hotet är inte värt högre säkerhet.

1 Introduktion

- Vad är ett protokoll?
- Ibland blir det fel
- Autentisering

2 Formell notation

- Protokoll
- BAN-logik
- Automated Verification

3 Protokoll och attacker

- Enkel autentisering
- Challenge–response
- Miljöbyte
- Internetbanken och betalkort

Example (Protokollbeskrivning)

Två principals P, P' ska kommunicera.

- 1 P skickar sitt namn till P' .
- 2 P' svarar med ett token t_P för vidare användning, detta är krypterat med P 's kryptonyckel k_P .

Example (Formell beskrivning)

Principals P, P' , token t_P , P 's kryptonyckel k_P .

$$P \rightarrow P' : P$$

$$P' \rightarrow P : \{t_P\}_{k_P}$$

Example (Protokollbeskrivning)

Två principals P, P' ska kommunicera.

- 1 P skickar sitt namn till P' .
- 2 P' svarar med ett token t_P för vidare användning, detta är krypterat med P 's kryptonyckel k_P .

Example (Formell beskrivning)

Principals P, P' , token t_P , P 's kryptonyckel k_P .

$$P \rightarrow P' : P$$

$$P' \rightarrow P : \{t_P\}_{k_P}$$

Example (Tentamen MIUN)

Låt T vara tentanden, V tentamensvakten, n_T det unika numret för T och S skrivningen. Vidare låt k vara en kryptonyckel delad mellan legitimationsutfärdaren och tentamensvakten (legitimation).

$$V \rightarrow T: n_T$$

$$T \rightarrow V: \{T\}_k, n_T$$

$$T \rightarrow V: \{T\}_k, n_T, S$$

Example (Autentisering KTH)

Låt T vara tentanden, V tentamensvakten och S skrivningen.
Vidare låt k vara en kryptonyckel delad mellan
legitimationsutfärdaren och tentamensvakten (legitimation).

$$T \rightarrow V: T, \{T\}_k$$

$$T \rightarrow V: T, S$$

Example (Needham-Schröder Shared-Key)

Låt A, B, S vara principals, S en server, n_A, n_B vara nonces².

$$A \rightarrow S: A, B, n_A$$

$$S \rightarrow A: \{n_A, B, k_{AB}, \{k_{AB}, A\}_{k_{BS}}\}_{k_{AS}}$$

$$A \rightarrow B: \{k_{AB}, A\}_{k_{BS}}$$

$$B \rightarrow A: \{n_B\}_{k_{AB}}$$

$$A \rightarrow B: \{n_B - 1\}_{k_{AB}}$$

²Number used once.



- Namn efter upphovsmän: Burrows, Abadi och Needham.
- Presenterades i en artikel [BAN90] från 1989.
- Används för formell analys av autentiseringsprotokoll.
- Har väsentligen utökats [SC01].

Definitioner I

- $A \models X$ A tror på X , och kan agera som att X vore sann.
- $A \sim X$ A sade X , eller något som innehöll X , vid något tidigare tillfälle. A trodde då på X och förstod att X skickades.
- $A \triangleleft X$ A ser X , eller har mottagit ett meddelande innehållandes X . Detta kan innebära att avkryptera.
- $A \Rightarrow X$ A är auktoritet över X . A går att lita på gällandes X .
- $\#(X)$ X är färsk, det vill säga inte en återuppspelning från tidigare protokollsession.
- $A \stackrel{k}{\leftrightarrow} B$ A och B delar nyckeln k . k är giltig för kommunikation mellan A och B .

Definitioner II

- $\xrightarrow{k} A$ A har den publika nyckeln k . Den motsvarande privata nyckeln k^{-1} hålls hemlig.
- $\{X\}_k$ X är krypterad med nyckeln k . Principals känner igen egna meddelanden och krypterade meddelanden kan då användas för att säkert identifiera avsändaren.



Har följande sex postulat:

- Message meaning,
- Nonce verification,
- Jurisdiction,
- Belief conjuncatenation,
- Freshness conjuncatenation,
- Seeing is receiving.

Postulat (Message meaning)

Låt A och B vara principals, k är en delad nyckel och X är ett uttalande. Då gäller

$$\frac{A \models A \stackrel{k}{\leftrightarrow} B, A \triangleleft \{X\}_k}{A \models B \sim X}.$$

Om A tror att k är giltig nyckel för att kommunicera med B och A har mottagit X krypterat med k , då kan A tro att B har sagt X .

Postulat (Nonce verification)

Låt A och B vara principals, X är ett uttalande. Då gäller

$$\frac{A \models \#(X), A \models B \vdash X}{A \models B \models X}.$$

Om A tror att X är färsk och A tror att B har sagt X , då kan A tro att B tror på X .

Postulat (Jurisdiction)

Låt A och B vara principals, K är en delad nyckel och X är ett uttalande. Då gäller

$$\frac{A \models B \Rightarrow X, A \models B \models X}{A \models X}.$$

Om A tror att B kontrollerar X och A tror att B tror på X , då kan A tro på X .



- Måste idealisera protkollet, skriva det i termer av BAN.
- Måste identifiera antaganden.
- Annotera protokollet.
- Använd logiken för att härleda vad principals tror på.

För detaljer och exempel, se [And08] avsnitt 3.8 och [SC01] avsnitt 2.3.

- Med BAN-logik kunde man visa att Needham-Schröder-protokollet krävde antagandet $B \models \#(A \stackrel{k_{AB}}{\leftrightarrow} B)$.
- Ledde till Denning-Sacco-attacken.

Denning-Sacco

Låt E_A beteckna angriparen som låtsas vara A .

$$E_A \rightarrow B: \{k_{AB}, A\}_{k_{BS}}$$

$$B \rightarrow E_A: \{n'_B\}_{k_{AB}}$$

$$E_A \rightarrow B: \{n'_B - 1\}_{k_{AB}}$$



Begränsningar

- Våra externa antaganden är ett problem: antag att nyckeln inte är tillgänglig för obehöriga.
- Kan bli problem vid idealiseringen av protokollet.
- Är tämligen komplext.

Automated Tools³

- There exists numerous tools, e.g. ProVerif [Bla].
- These are generally based on two things:
 - an equational theory,
 - an inference system.
- Then the equations are used to verify that the security properties hold.
- We generate a proof that our properties cannot be broken under the assumptions of the equational theory.

³V. Cortier and S. Kremer. 'Formal Models and Techniques for Analyzing Security Protocols: A Tutorial'. In: *Foundations and Trends in Programming Languages* 1.3 (2014), pp. 151–264. URL: <http://www.loria.fr/~skremer/fosad14/notes.pdf>; Hubert Comon-Lundh and Stéphanie Delaune. 'Formal Security Proofs'. In: *Software Safety and Security*. Ed. by T. H. Nielsen, G. G. Balasubramanian, and P. Li. Hoboken, NJ: Wiley, 2014.

Definition (Inference System)

- An *inference rule* (or *axiom*) is a rule of the form $\frac{u_1, \dots, u_n}{u}$, where u_1, \dots, u_n are terms in the formal system.
- u is inferred from u_1, \dots, u_n .
- An *inference system* is a set of inference rules.

Example (BAN logic)

The axioms of BAN logic is an example of an inference system.

Definition (Inference System)

- An *inference rule* (or *axiom*) is a rule of the form $\frac{u_1, \dots, u_n}{u}$, where u_1, \dots, u_n are terms in the formal system.
- u is inferred from u_1, \dots, u_n .
- An *inference system* is a set of inference rules.

Example (BAN logic)

The axioms of BAN logic is an example of an inference system.

Example (Dolev-Yao inference system)

- Can deduce concatenation from items: $\frac{x,y}{x||y}$
- Can deduce items from concatenation: $\frac{x||y}{x}$ and $\frac{x||y}{y}$
- Can encrypt with plaintext and key: $\frac{x,y}{\{x\}_y}$
- Can decrypt with ciphertext and key: $\frac{\{x\}_y,y}{x}$
- ...

Definition (Equational theory)

Is simply a set of equations using terms from a formal system.

Example

- $x \oplus (y \oplus z) = (x \oplus y) \oplus z$
- $x \oplus y = y \oplus x$
- $x \oplus x = 0$
- $x \oplus 0 = x$

Putting it together

- This type of property cannot be expressed with an inference system.
- We need a complementing equational theory.



Example (Public-key crypto with ProVerif)

```

1 type skey.
2 type pkey.
3 type seed.
4 type block.
5 type encblock.
6
7 (* Probabilistic public-key encryption *)
8
9 fun pk(skey): pkey.
10 fun enc(block, pkey, seed): encblock.
11 fun dec(encblock, skey): block.
12 equation forall x: block, y: skey, z: seed; dec(
    enc(x, pk(y), z), y) = x.

```



- We also have to defined the protocol interactions for ProVerif.
- The idea is similar to that of BAN-logic.
- ProVerif does the analysis automatically.

Remember

This only complements the rigorous analysis made by experts!

- We also have to defined the protocol interactions for ProVerif.
- The idea is similar to that of BAN-logic.
- ProVerif does the analysis automatically.

Remember

This only complements the rigorous analysis made by experts!

1 Introduktion

- Vad är ett protokoll?
- Ibland blir det fel
- Autentisering

2 Formell notation

- Protokoll
- BAN-logik
- Automated Verification

3 Protokoll och attacker

- Enkel autentisering
- Challenge–response
- Miljöbyte
- Internetbanken och betalkort

Example (Bättre fjärrlås)

Låt A, B vara principals, n nonce, k_A en nyckel unik för A .

$$A \rightarrow B: A, \{A, n\}_{k_A}$$

Egenskaper

- Nonce n för färskhet.
- Krypteringen för identifiering.

Nyckelhantering

- Måste hantera nycklarna k_i för alla enheter i .
- *Nyckeldiversifiering*: huvudnyckel k_M och generera $k_i = \{i\}_{k_M}$.
- Måste tänka efter:
 - 128-bitar nyckel krypterar 16-bitar ID, mindre lämpligt för diversifiering.
 - Svagt chiffer ger också dåligt resultat.
 - $k_i = i \oplus k_M$?



Kolla nonces

Kolla nonces långt tillbaka i tiden.

- Jämför med senaste nonce.
- Spela in två och spela upp dem varannan gång.
- Förbetalda elmätare, köp två laddningar och använd dem om vartannat.



Betjäntattacken

- Hur genereras nonces?
- En person som har tillfällig åtkomst att generera tokens.
- Generera ett antal, använd dem senare.
- Exempelvis engångskoder för att logga in hos internetbanken.
- Attacken fungerar om nonces är (pseudo-) slumpstal.



Kontra betjäntattacken

Förbättring

- Använd en räknare c som successivt ökas på.
- $A \rightarrow B: A, \{A, c + 1\}_{k_A}, c = c + 1.$
- Inget $c' \leq c$ accepteras.

Problem

- Får inte ha jämförelsen $c' = c$, ger synkroniseringsproblem.
- $c \notin \mathbb{Z}_+$ utan $c \in \mathbb{Z}_{2^x}$, för något $x \in \mathbb{N}$: vid något tillfälle blir då $c + 1 < c \pmod{2^x}$.



Andra tillämpningar

- Tillbehörskontroll: skrivare ändrar inställning från 1200 dpi till 300 dpi om icke-originalbläckpatroner används.
- 'Använd alltid godkända originaldelar'.
- Inte hålla angripare ute, utan hålla användare inne.
- Läs kapitel 7 *Economics* i [And08] för vidare diskussion.

Grundläggande princip

Två principals A, B med gemensam nyckel k och nonce n .

$$A \rightarrow B: n$$

$$B \rightarrow A: \{B, n\}_k$$

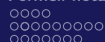
Problem

- Dåliga (pseudo-) slumpalsgeneratorer, ger förutsägbara n .



Tvåfaktorautentisering

- Ha användarnamn och lösenord.
- Komplettera med extern kod; exempelvis genererad av koddosa, SMS till mobiltelefonen.
- Finns många varianter, kombinera två:
 - Något du vet (lösenord),
 - något du har (koddosa, mobiltelefon),
 - något du är (biometrik).



Tvåfaktorautentisering

Protokoll (tvåfaktorautentisering med koddosa)

Låt A, B, D vara principals, D är koddosa, k är nyckel delad mellan B, D och p är A 's PIN-kod.

$$A \rightarrow B: A$$

$$B \rightarrow A: n$$

$$A \rightarrow D: n, p$$

$$D \rightarrow A: \{n\}_k$$

$$A \rightarrow B: \{n\}_k$$

Tvåkanalsautentisering

Protokoll (tvåkanalsautentisering med mobiltelefon)

Låt A, B, M vara principals, M är mobiltelefon och p är A :s lösenord.

$$A \rightarrow B: A, p$$

$$B \rightarrow M: n$$

$$M \rightarrow A: n$$

$$A \rightarrow B: n$$

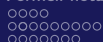


- Betalkortsystemet designades för en pålitlig miljö.
- Kraftigt reglerad miljö inbyggd i bankens fasad.
- Tillämpas i den mindre pålitliga miljön i samtliga affärer.
- Skimming.



Personen i mitten

- 'Det är enkelt att spela oavgjort mot en schackstormästare i postschack: spela bara mot två stormästare samtidigt, en som vit och en som svart, och skicka deras brev mellan varandra.'
(John Convey)
- Problem med pålitliga användargränssnitt: hur vet du att inte kortterminalen ljuger?



Olika former av bankdosa

Swedbank

- Individuell dosa, förkonfigurerad av banken.
- Kan generera engångskod.
- Kan hantera challenge–response.

Nordea

- Oberoende smartkortläsare, använder individuellt betalkort.
- Kan generera engångskod.
- Kan hantera challenge–response.



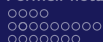
Olika former av bankdosor

Swedbank

- Individuell dosa, förkonfigurerad av banken.
- Kan generera engångskod.
- Kan hantera challenge–response.

Nordea

- Oberoende smartkortläsare, använder individuellt betalkort.
- Kan generera engångskod.
- Kan hantera challenge–response.



Problem som kan uppstå

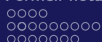
Problem

- Om bankkort och dosa förvaras tillsammans kan PIN-koden utläsas från de slitna knapparna på bankdosan.
- Om kortet används i en dålig terminal har angriparna allt som behövs för att logga in till ditt bankkonto.

Förbättringar

- Använd inte samma säkerhetsmekanism i flera sammanhang.
- Ha separata oberoende mekanismer.
- Ha ett pålitligt användargränssnitt.





Problem som kan uppstå

Problem

- Om bankkort och dosa förvaras tillsammans kan PIN-koden utläsas från de slitna knapparna på bankdosan.
- Om kortet används i en dålig terminal har angriparna allt som behövs för att logga in till ditt bankkonto.

Förbättringar

- Använd inte samma säkerhetsmekanism i flera sammanhang.
- Ha separata oberoende mekanismer.
- Ha ett pålitligt användargränssnitt.



Referenser

- [And08] Ross J. Anderson. *Security Engineering. A guide to building dependable distributed systems*. 2nd ed. Indianapolis, IN: Wiley, 2008. ISBN: 978-0-470-06852-6 (hbk.) URL: <http://www.cl.cam.ac.uk/~rja14/book.html>.
- [BAN90] Michael Burrows, Martin Abadi and Roger Needham. 'A logic of authentication'. In: *ACM Transactions on Computer Systems (TOCS)* 8.1 (Feb. 1990). URL: <https://dl.acm.org/citation.cfm?id=77649>.
- [Bla] Bruno Blanchet. *ProVerif: Cryptographic protocol verifier in the formal model*. Fetched on 25th April 2016. URL: <http://prosecco.gforge.inria.fr/personal/bblanche/proverif/>