暗号プロトコル設計演習 第一週例題

21T2166D 渡辺大樹

2023 年 7 月 5 日

演習内容

ProVerif を用いて設計したメッセージ認証とデジタル署名のプロトコルは以下のソースコード 1,2 となった。

ソースコード 1　MAC.pv

```
1  free c:channel.
2  free s:bitstring[private]. (*秘密鍵*)
3  free m:bitstring. (*メッセージ*)
4
5  (* fun MAC(bitstring, bitstring): bitstring.
6  fun VER(bitstring, bitstring, bitstring): bool.
7  reduc forall x: bitstring, s: bitstring;
8   VER(x,MAC(x,s),s) = true. *)
9
10 fun hash(bitstring): bitstring.
11 fun con(bitstring,bitstring): bitstring.
12 reduc forall x:bitstring, s:bitstring;
13  h(x,s) = hash(con(x,s)). (*メッセージと秘密鍵をくっつけてハッシュ化*)
14
15
16 event SEND.
17 event ACC1.
18
19 let AuthS (PSK:bitstring)
20  = in (c, (mr:bitstring,d:bitstring));
21  if( h(mr,PSK) = d)
22     then event ACC1. (*メッセージと秘密鍵のハッシュ値が送られてきたやつと一緒な
           らACC*)
23
24 query event(ACC1) ==> event(SEND).
```

```
25
26 process
27 ( event SEND;
28     out (c, (m,h(m,s)) ) (*メッセージと秘密鍵をくっつけてハッシュ化したやつとメ
       ッセージを送信*)
29 )
30 | AuthS (s)
```

ソースコード 2　SIG.pv

```
1  free c:channel.
2  free s:bitstring[private].
3  free m:bitstring.
4
5  fun gPK(bitstring): bitstring.
6  fun SIG(bitstring, bitstring): bitstring.
7  (* fun VERS(bitstring, bitstring, bitstring): bool. *)
8  reduc forall x: bitstring, s: bitstring;
9  VERS(x,SIG(x,s),gPK(s)) = true.
10
11 event SEND.
12 event ACC.
13
14 let AuthS (spk:bitstring) (*公開鍵を知っている認証サーバ*)
15  = in (c,(mr:bitstring, sg:bitstring)); (*メッセージとデジタル署名を受け取る*)
16  if(VERS(mr,sg,spk))
17     then event ACC. (*公開鍵を用いて認証、あっていればACC イベントを実行*)
18
19 query event(ACC) ==> event(SEND).
20
21 process
22 ( event SEND;
23     out (c, (m, SIG(m,s))) (*メッセージとデジタル署名を送信*)
24 )
25 | AuthS (gPK(s))
```

コードの内容はソースコード内のコメントで記している。