

暗号プロトコル設計演習 第2週例題

信州大学工学部
電子情報システム工学科

実験日: 2023/07/05

実験場所: W1 棟 115 教室

実験者: 21T2166D 渡辺 大樹

共同実験者: 21T2167B 渡邊 大翔

21T2804J 伊藤 星斗

1 演習内容

第二週の暗号プロトコル設計演習では前回学習した暗号技術を用いて、課題として課されたプロトコルを設計する。

今回実装した暗号プロトコルはハイブリッド暗号で、そのプロトコルと実装したコードを示していく。

2 暗号プロトコル

2.1 ハイブリッド暗号

以下にハイブリッド暗号を実装したコードを示す。

ソースコード 1 hyb.pv

```
1 free c:channel.
2 type skey.
3 type pkey.
4 free askey:skey [private].
5 free m:bitstring [private].
6
7 fun pk(skey):pkey.
8 fun enc(bitstring,pkey):bitstring.
9 fun dec(bitstring,skey):bitstring.
10 equation forall
11   x:bitstring, sk:skey;
12   dec(enc(x,pk(sk)),sk) = x. (*この関数は共通鍵の暗号化,復号化に使う関数*)
13
14 fun encm(bitstring,bitstring):bitstring.
15 fun decm(bitstring,bitstring):bitstring.
16 equation forall
17   x:bitstring, y:bitstring;
18   decm(encm(x,y), y) = x. (*この関数はメッセージの,共通鍵での暗号化,復号化に使う関数*)
19
20 event SEND.
21 event RECS.
22
23 query attacker(m).
24 query event (RECS).
25 query event (RECS) ==> event (SEND).
26
```

```

27 let RECP(sk:skey)
28 = in (c, (mr:bitstring, ssk:bitstring));
29 let shk = dec(ssk, sk) in(*暗号化された共通鍵を秘密鍵で復号*)
30 let dem = decm(mr, shk) in(*暗号化されたメッセージを共通鍵で復号*)
31 if (dem = m)
32   then event RECS.(*送ったメッセージと復号したメッセージがあればイベント*)
33
34
35 process
36 ( new shkey:bitstring;
37   event SEND;
38   out (c, (encm(m,shkey),enc(shkey, pk(askey)))) )
39 (*メッセージを共通鍵で暗号化したものと,共通鍵を公開鍵で暗号化したものを送信*)
40 )
41 | RECP(askey)

```

ソースコード 1 にはいくつかコメントを入力しており、その中である程度コードの構造を説明しているが以下で今一度解説していく。

ハイブリッド暗号では共通鍵暗号方式と公開鍵暗号方式を組み合わせ用いられる暗号方式である。共通鍵暗号方式は、共通鍵という鍵を送信者受信者で共有し、送信したい平文を暗号化復号化する暗号方式である。

共通鍵はその鍵を共有するのに通信路で送信するため、そのまま送信すると攻撃者に知られてしまう可能性がある。そのためここで公開鍵暗号方式を用いる。共通鍵を通信路に送るために、送信者が共通鍵を公開鍵で暗号化し、その暗号化された共通鍵を受信者が公開鍵に対応する秘密鍵で復号する。ここで復号した共通鍵を用いて送られた暗号文を復号できる。

以上がハイブリッド暗号方式の構造となる。

3 演習結果

以上の設計した暗号プロトコルを ProVerif で検証した結果、以下のような出力を得られた。

Linear part: No equation.

Convergent part:

(中略)

Verification summary:

Query not attacker(m[]) is true.

Query not event(RECS) is false.

Query event(RECS) ==> event(SEND) is true.

この結果より、このプロトコルでは攻撃者から送信したメッセージはばれておらず、また送信時のイベント、受信時のイベントも起こっていることが分かる。