暗号プロトコル評価結果

独立行政法人　情報通信研究機構

1.プロトコル名：Basic Access Control (ISO/IEC 11770-2-6)

2.関連する標準

Machine Readable Travel Documents Technical Report

http://www.csca-si.gov.si/TR-PKI_mrtds_ICC_read-only_access_v1_1.pdf

3.使用したツール：Scyther

4.評価の概要：Scyther による評価では、non-injective synchronization への攻撃の可能性が指摘されているが、認証の意味では意味のある攻撃ではない。

5.Scyther による評価

5.1.シーケンス記述

```
hashfunction kdf;
hashfunction mac;
usertype String;
const GetChallenge: String;
const Cenc, Cmac: String;
protocol BasicAccessControl(A, P)
{
  role A
  {
    // variables
    var Np: Nonce;
    fresh Na, Ka: Nonce;
    var Kp: Nonce;
    // sequence
    //    send_0(A, P,  GetChallenge);
    recv_1(P, A,  Np);
    send_2(A, P,
{Na,Np,Ka}kdf(k(A,P), Cenc),
mac({Na,Np,Ka}kdf(k(A,P), Cenc),
```

```
(Na, Np), kdf(k(A,P), Cmac)));
    recv_3(P, A,
{Np,Na,Kp}kdf(k(A,P), Cenc),
mac({Np,Na,Kp}kdf(k(A,P), Cenc), (Na, Np), kdf(k(A,P), Cmac)));
  }
  role P
  {
    // variables
    fresh Np: Nonce;
    var Na, Ka: Nonce;
    fresh Kp: Nonce;
    // sequence
    //    recv_0(A, P,  GetChallenge);
    send_1(P, A,  Np);
    recv_2(A, P,
{Na,Np,Ka}kdf(k(A,P), Cenc),
mac({Na,Np,Ka}kdf(k(A,P), Cenc), (Na, Np), kdf(k(A,P), Cmac)));
    send_3(P, A,
{Np,Na,Kp}kdf(k(A,P), Cenc),
mac({Np,Na,Kp}kdf(k(A,P), Cenc), (Na, Np), kdf(k(A,P), Cmac)));
  }
}
```

5.2.攻撃者モデル

Scyther はデフォルトで Dolev-Yao モデルの通信路を想定しているため、Scyther を利用した評価で攻撃者モデルについて記載すべき項目はない。

5.3.セキュリティプロパティの記述

```
// ロール A のセキュリティプロパティ
claim_a1(A, Secret, Ka);
claim_a2(A, Secret, Kp);
claim_a3(A, Weakagree);
```

```
claim_a4(A, Niagree);
claim_a5(A, Nisynch);
// ロールPのセキュリティプロパティ
claim_p1(P, Secret, Ka);
claim_p2(P, Secret, Kp);
claim_p3(P, Weakagree);
claim_p4(P, Niagree);
claim_p5(P, Nisynch);
```

5.4.検証結果

○評価ツールの出力

Scytherによる評価結果概要は以下のとおりである。

```
claim   BasicAccessControl,A    Secret_a1       Ka      Ok
[proof of correctness]
claim   BasicAccessControl,A    Secret_a2       Kp      Ok
[proof of correctness]
claim   BasicAccessControl,A    Nisynch_a5      -       Fail    [at
least 2 attacks]
claim   BasicAccessControl,P    Secret_p1       Ka      Ok
[proof of correctness]
claim   BasicAccessControl,P    Secret_p2       Kp      Ok
[proof of correctness]
claim   BasicAccessControl,P    Nisynch_p5      -       Fail    [at
least 2 attacks]
```

すなわち、鍵Ka及び鍵Kpの秘匿性は満たされている。その一方で、ロールA、ロールPのいずれもNisynchを満たさない。これは、暗号プロトコルの最初に固定値を送信していることに由来している。Scyther言語による記述において、send_0, recv_0を削除することで、すべての要件が満たされる。

○攻撃に関する解説

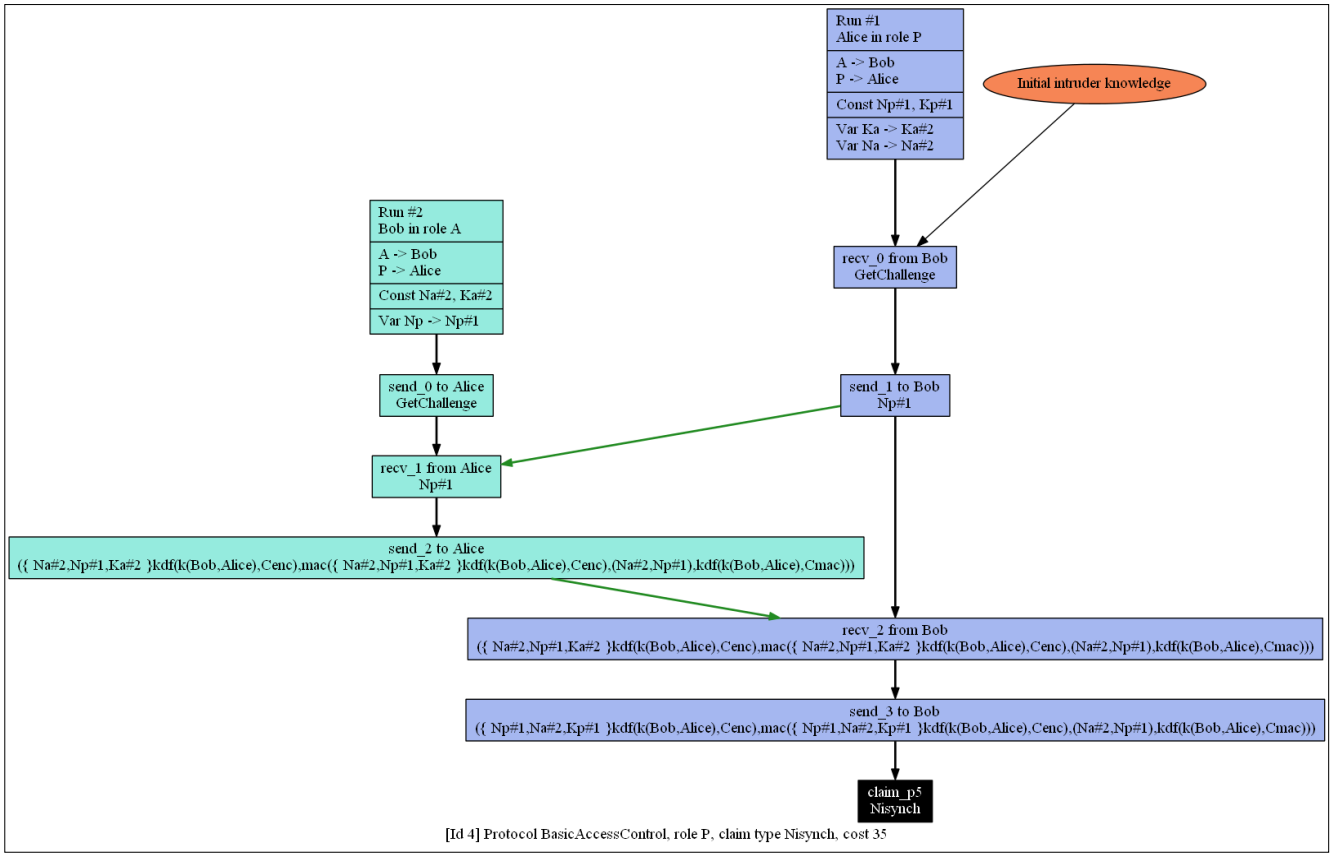

図 1. 攻撃に関する説明

図 1. は、Basic Access Control プロトコルがロール P について non-injective synchronization を満たさない事例を表している。図 1. では、ロール A が生成した GET_CHALLENGE コマンドがロール P に届いていないことが見てとれる。これは、GET_CHALLENGE コマンドが固定値であることに起因する。ロール A についても同様の攻撃が報告される。しかし、これらのいずれも本質的な脆弱性ではないと考えられる。

5.5. モデル化

○モデル化プロセス

・暗号処理関数

BAC プロトコルでは、KDF に SHA-1 を、暗号化処理に 2-key Triple-DES を、MAC に DES ベースの ISO/IEC 9797-1 Algorithm 3 を用いている。モデル化に際して、それぞれを「ハッシュ関数（理想的な一方向性関数）」、「暗号化」、「ハッシュ関数」で置き換えている。また、MRZ_info から Kseed を生成する手順は省略し、Kseed を事前共有鍵として取り扱った。

・シーケンス：本来の仕様をそのまま用いている。ただし、暗号プロトコルとしては、暗号プロトコルをキックする GET_CHALLENGE パケットは不要である。

・攻撃者：記述なし。
・セキュリティプロパティ：agreement にはいくつかのレベルがあり、システム要件がないため、どれを選ぶべきかはそれほど明確ではない。ここでは、Scyther で評価可能なもっとも強いセキュリティである non-injective agreement を選択した。

5.6.モデル化の妥当性

BAC プロトコルの安全性は、MRZ_info の秘匿性が満たされていることが最低条件である。しかし、MRZ_info はパスポート番号、生年月日などパスポート上に記載されている情報のみで構成されており、その一部は比較的流出しやすい情報である。したがって、現実において MRZ_info が十分な秘匿性を満たしているかどうかは不明である。

5.7.評価ツールとの相性

〇暗号プロトコルの記述可能性
特になし。

〇セキュリティプロパティの記述可能性
特になし。

5.8.評価ツールの性能

・評価環境
- CPU：Intel Core2Duo E8400 (3.0 GHz)
- メモリ：4 GB
・性能
- 時間：0.1 sec

5.9.備考

本文書は、**総務省「暗号・認証技術等を用いた安全な通信環境推進事業に関する実証実験の請負　成果報告書**」からの引用である