ProVerif を用いた Bluetooth の安全性解析：Passkey Entry モードの検証に向けて

塚田研究室　218K6023　河合　悠斗

1　はじめに

暗号プロトコルの自動検証ツールとして ProVerif[1]がある。一方、Bluetooth は情報端末間または情報端末とキーボードなど周辺機器を接続する際に用いられる無線通信方式である。Bluetooth において通信可能な状態に端末間で相互認証し、関連付けを行う手順をセキュアシンプルペアリング（SSP）と呼ぶ。2007 年に SSP を利用した Bluetooth2.1+EDR[2]が登場した。しかしながら、Chang らによって ProVerif を用いた安全検証で SSP のモードの 1 つである Numeric Comparison モード[3]、井上らによって Passkey Entry モード[4]に対する攻撃法を確認している。また、Numeric Comparison モードの問題に対しては Yeh ら[5]が、Passkey Entry モードの問題に対しては Sun ら[6]や Sai ら[7]が改良案を提案している。だが、横山らによって ProVerif を用いた安全検証で Yeh らの Numeric Comparison モードの攻撃法を確認し、攻撃法に対する対策をしている[8]。それに対し、改良された Passkey Entry モードについては安全検証が行われていない。

本研究では Yeh らの Numeric Comparison モードについて ProVerif を用いて再現実験を行う。さらに改良された Passkey Entry モードに向けて形式化する上での難しさや予想される攻撃法について考察する。

2　ProVerif

ProVerif は Blanchet らが開発した形式モデルで自動検証ツールであり、暗号プロトコルで要求される秘匿性や認証性などの安全性を検証可能である。ProVerif はさまざまな暗号プロトコルの検証に用いられ、多くの暗号プロトコルに対して脆弱性を発見することに成功している。

3　Bluetooth

3—1　Bluetooth の概要

Bluetooth は情報端末間または情報端末とキーボードなど周辺機器を接続する際に用いられる無線通信方式である。Bluetooth において通信可能な状態に端末間で相互認証し、関連付けを行う手順を SSP と呼ぶ。SSP には Numeric Comparison, Just Works, Out Of Band, Passkey Entry という 4 種類のモードがある。Numeric Comparison モードではペアリングする機器両方の画面に 6 桁の数字が表示され、人間の目で両方の機器の値が同じかどうか確認し、同じであれば yes を送信して認証が完了する。それに対して Passkey Entry モードではマスター側で表示された 6 桁の数字(パスキー)をデバイス側に入力して認証を行います。

SSP は 5 つのフェーズがある。フェーズ 1 では楕円曲線 Diffe Hellman(ECDH)鍵共有により公開鍵を交換し、共有鍵(DHkey)を生成する(図 1)。フェーズ 2 ではフェーズ 3,4 で用いられる値を共有する(図 2, 図 3)。フェーズ 3 では共有した全ての値が端末間で正しく共有できていることを検証する(図 4)。フェーズ 4 でリンクキーを生成し、フェーズ 5 で認証・暗号化を行う(図 5)。

フェーズ 2 には前述の 4 つのモードが規定されており、端末が持つユーザインターフェース I/O によってモードが選択される。

図 1. 公開鍵交換(フェーズ 1)

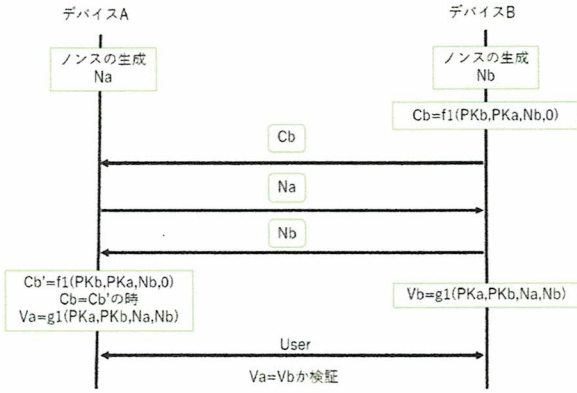

図 2. Numeric Comparison モード(フェーズ 2)

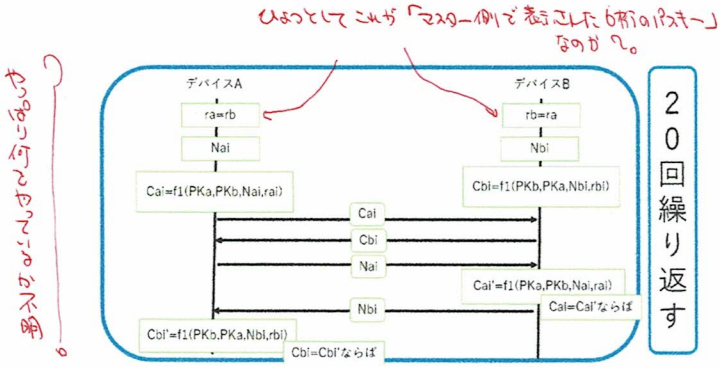

図 3. Passkey Entry モード(フェーズ 2)

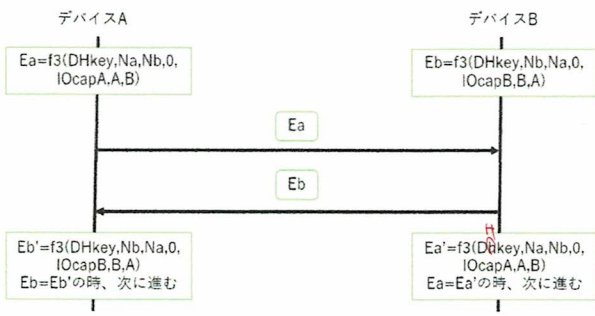

図 4. フェーズ 3

図 5. フェーズ 4, 5

3—2 Yeh らの Numeric Comparison モード

本節では Yeh らの Numeric Comparison モードについて説明する。2 つのデバイスそれぞれの画面に表示された値が同じかどうか確認し、同じならば Yes、違えば No のボタンを押す実験がノキア研究所によって行われ、5 人に 1 人、数字が違っているにもかかわらず Yes を押したという結果がある[9]。そのため、大間の目による確認は人為的ミスが起こる可能性の対策に加え、中間者攻撃に耐性を持つように改良された。Yeh らの Numeric Comparison モードはフェーズ 1 の公開鍵交換プロトコルとフェーズ 2、フェーズ 3 の認証プロトコルを組み合わせて改良している。

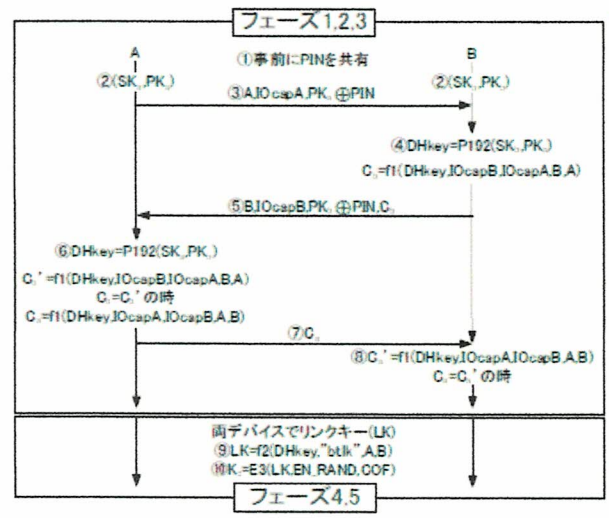

図 6. Yeh らの Numeric Comparison モード

図 6 にしたがって Yeh らの Numeric Comparison モードを説明する。デバイス A、B ともに PIN の値を入力する。次に、デバイス A は Bluetooth Device 値 A、デバイス A の I/O 情報 I/OcapA、PKa⊕PIN をデバイス B に送信する。デバイス B は受け取った値 PKa⊕PIN に対して PIN を排他的論理和し、デバイス A の公開鍵 PKa を得る。自身の秘密鍵 SKb と PKa を鍵生成関数 P192 に入力し、DHkey=P192(SKb, PKa)の生成を行う。デバイス B は DHkey、デバイス A、B の I/O 情報 IOcapA、IOcapB と Bluetooth Device 値 A、B よりハッシュ値 Cb=f1(DHkey, IOcapB, IOcapA, B, A)を計算し、Cb、B、IOcapB、PKb⊕PIN をデバイス A に送信する。デバイス A は受け取った値 PKb⊕PIN に対して排他的論理和し、デバイス B の公開鍵 PKb を得る。自身の秘密鍵 SKa と PKb を鍵生成関数 P192 に入力し、DHkey=P192(Ska, PKb)の生成を行う。デバイス A は DHkey、デバイス A、B の I/O 情報 IOcapA、IOcapB

と Bluetooth Device 値 A、B よりハッシュ値 Cb'=f1(DHkey, IOcapB, IOcapA, B, A)を計算する。デバイス A は Cb=Cb'を検証する。Cb≠Cb'ならデバイス A はペアリングを中断し、Cb=Cb'ならばデバイス A はデバイス B に Ca=f1(DHkey, IOcapA, IOcapB, A, B)を送信する。デバイス B はハッシュ値 Ca'=f1(DHkey, IOcapA, IOcapB, A, B)を計算する。Ca=Ca'を検証する。Ca≠Ca'ならばデバイス A はペアリングを中断し、Ca=Ca'ならばフェーズ 4 に進む。フェーズ 4 でリンクキーLK=f2(DHkey, "btlk", A, B)を計算する。ここで、btlk はプロトコル固有の定数である。フェーズ 5 では、この LK を用いて共有鍵 Kc=E3(LK, EN_RAND, COF)を得る。ここで、EN_RAND、COF はプロトコル固有の定数である。

4 形式化

4—1 ECDL 及び ECDH の形式化

鍵生成関数 p192 を形式化する際に ECDL 及び ECDH の議論が必要である。そのため、ProVerif 上でどのように形式化しているかを解説する。

- 楕円曲線上の離散対数問題(ECDL):既知の(P, aP)に対して、a を求めること
- 楕円曲線上の計算 DH 問題(ECDH):既知の(P, aP, bP)に対して、abP を求めること

Blanchet らは、Diffie-Hellman 鍵共有をどのように表現するかを示し、DL 及び CDH を形式化している[1]。同様の考えで ECDL 及び ECDH について以下のように定義する。なお、fun には定義したい関数の関数名、その関数の定義域及び値域を記述し、equation には fun で定義した関数がどのような関係を有しているかを記述する。

```
1. fun P192(scalar,G1):G1.
2. equation forall a:scalar, b:scalar
     P192(a,P192(b,P))=P192(b,P192(a,P)).
```

2 行目の equation は a(bP)=b(aP)を意味している。すなわち、P、aP 及び bP を知っていても a, b のいずれかを知っていないと abP は計算できないことを意味している。これは ECDH を形式化していることに他ならない。さらに、1 行目の fun の記述そのものが P192(a, P)=aP を意味していることがわかる。すなわち、P 及び aP を知っていても a を知らないと aP は計算できないことを意味している。

4—2 排他的論理和の形式化

Yeh らの Numeric Comparison モードで形式化する際に、排他的論理和(XOR)の議論が必要となる。そのため、ProVerif 上でどのように排他的論理和を形式化しているのかを解説する。

```
1. fun xor(G1,G1):G1.
2. equation forall x:G1,y:G1;
     xor(xor(x,y),y)=x.
3. equation forall x:G1;
     xor(x,xor(x,x))=x.
4. equation forall x:G1;
     xor(xor(x,x),x)=x.
5. equation forall x:G1,y:G1;
     xor(y,xor(x,x))=y.
```

2 行目の equation は((x⊕y)⊕y)=x を意味している。3、4 行目で排他的論理和は可換であることを表している。ただし、ProVerif 上では、xor(x,y)=xor(y,x)のような可換則は扱うことができない。

5 検証結果

Yeh らの Numeric Comparison モードを ProVerif により形式化し、先行研究[7]と同じ検証を行った結果、秘匿性、認証性について以下の結果となった。共通鍵 Kc が秘匿性を満たしているか検証した。認証性については(1)参加者が誰であるか(以降、Weak Authentication)、(2)参加者が誰であるかと鍵を誰が生成したか(以降、Strong Authentication)を検証した。

表 1. 秘匿性の検証結果

Property	Result
Secret A	True
Secret B	True

表 2. 認証性の検証結果

Property			Result
Strong Authentication	Injective	A-to-B	False
		B-to-A	False
	Non-injective	A-to-B	True
		B-to-A	False
Weak Authentication	Injective	A-to-B	False
		B-to-A	False
	Non-injective	A-to-B	True
		B-to-A	False

表 1 より、共有鍵 Kc が秘匿性を満たしていることを確認した。表 2 より認証性は、AtoB(デバイス A によるデバイス B の認証)の場合、Non-injective が成立している。ただし、Injective は成立していないため、再生攻撃を受ける。BtoA(デバイス B によるデバイス A の認証)の場合は Non-injective も Injective も成立していないため、なりすましが可能である。先行研究[7]と比較したところ、同じ結果を得られた。

6 まとめ

Yeh らの Numeric Comparison モードを ProVerif で形式化し、検証を行った結果、先行研究[7]と同じ結果を得られた。

Passkey Entry モードを形式化する上で考えられる難しさを説明する。フェーズ 1、3、4、5 は同じであるから変化するところはフェーズ 2 だけである。Numeric Comparison モードと違う点として入力した 10 進数の 6 桁の値、20bit を 1bit ずつ計算に用いて、図 3 の流れを 20 回繰り返す。この点が ProVerif で形式化する上で、難しいと考えられる。また、先行研究[4]より Passkey Entry モードにはなりすましの攻撃法が確認されている。よってなりすましの改良は行われていると考えられるが、Yeh らの Numeric Comparison モードのように一部だけ改善されていてまだなりすましが可能であり、それに加え、再生攻撃も可能であるのではないかと予想される。

参考文献

[1] Bruno Blanchet : “ProVerif”, https://prosecco.gforge.inria.fr/personal/bblanche/proverif/

[2] Bluetooth SIG, Bletooth2.1+EDR Core Specification, BluetoothSIG, 2007

[3] R. Chang and V. shimatikov: Formal Analysis of Authentication in Bluetooth Device Paring, Proc. Joint Workshop on Foundations of Computer Security and Automated Reasoning for Security Protocol Analysis (FCS-ARSPA 2007), pp. 45-61 (2007)

[4] 井上 博之, 荒井 研一, 金子 敏信 : “ProVerif による Bluetooth のセキュアシンプルペアリングに対する形式的な安全検証”, 日本応用数理学会 2013 年春の研究部会連合発表会「数理的技法による情報セキュリティ」(FAIS)セッション, 2013

[5] T. C. Yeh, J. R. Peng, S. S. Wang, and J. P. Hsu : “Securing Bluetooth Communications”, International Journal of Network Security, Vol. 14, No. 4, PP. 229-235, July 2012

[6] Da-Zhi Sun, Yi Mu, Willy Susilo : “Man-in-the-middle attacks on Secure Simple Pairing in Bluetooth standard V5.0 and its countermeasure”, Pers Ubiquit Comput (2018)

[7] Sai Swaroop Madugula and Ruizhong Wei : “An Enhanced Passkey Entry Protocol for Secure Simple Pairing in Bluetooth”, Lakehead University Department of Computer Science 2021

[8] 横山 雄太, 岩本 智裕, 荒井 研一, 金子 敏信 : “ProVerif を用いた Bluetooth のセキュアシンプルペアリングの形式的検証”, Computer Security Symposium 20, 21-23 October 20

[9] E. Uzum, K. Karvonen, and N. Asokan : “Usability Analysis of Secure Pairing Methods”, Nokia Research Center Technical Reports 2007 http://research.nokia.com/tr/NRC-TR-2007-002.pdf