RCIS Research Center for Information Security

情報セキュリティ研究センター
Research Center for Information Security

Certifying Assembly with Cryptographic Proofs

〈暗号学的安全性証明によるアセンブリプログラムの安全性検証〉

レナルド アフェルト, ダヴィット ノヴァック, やまだ きよし
Reynald AFFELDT, David NOWAK, and 山田 聖

背景・動機・特徴

- 暗号プログラムの組み込まれた機器の普及
- 暗号プログラムの安全性とは?
  1)暗号アルゴリズムが安全
  2)暗号プログラムは暗号アルゴリズムを正しく実装
- 本研究は数理的技法に基づき1)と2)の検証を組み合わせ暗号プログラムの安全性を証明する初めてのもの
  - 暗号プログラムの安全性検証結果は自動検証可能

検証対象(現時点)

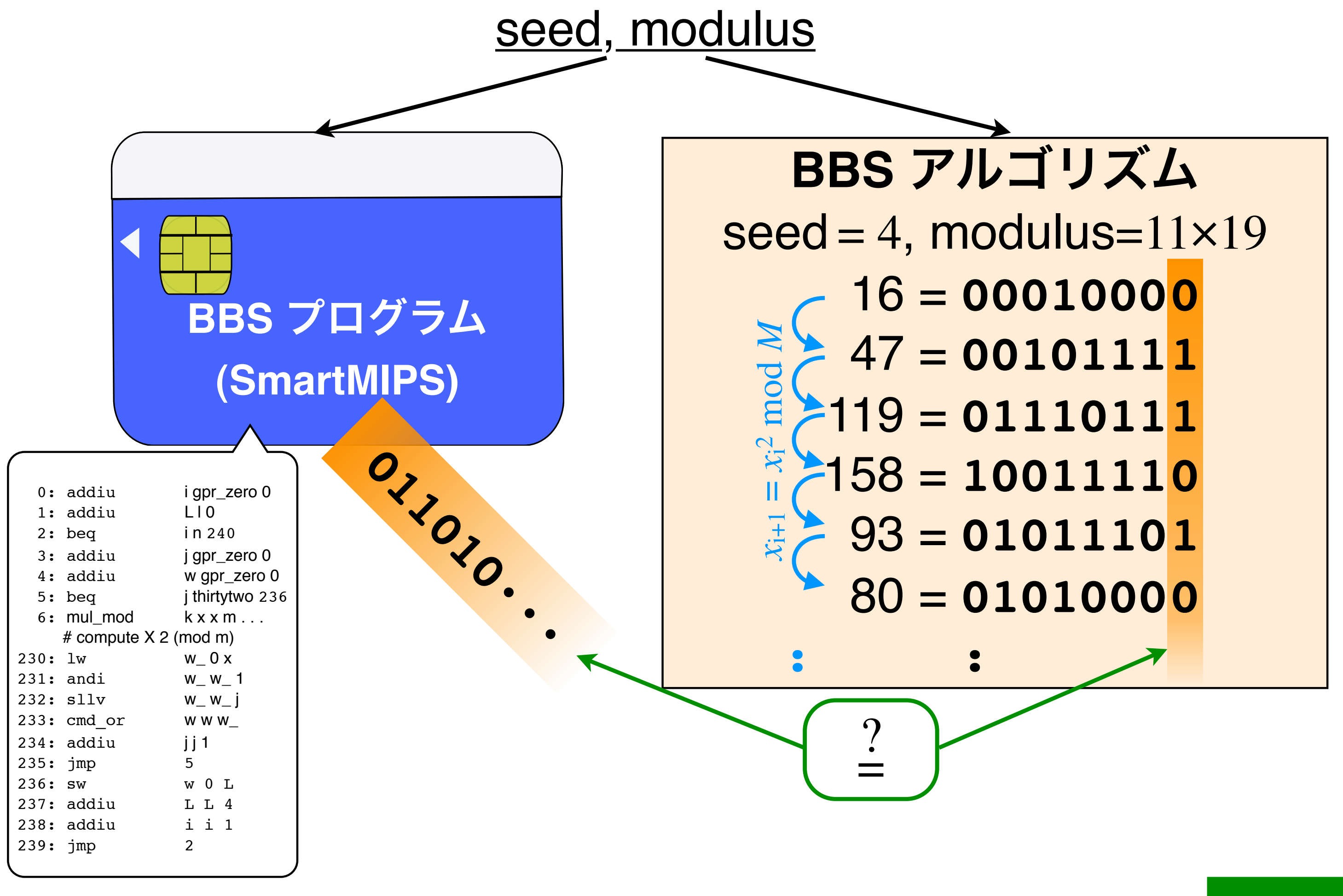
- Blum Blum Shub(BBS)疑似乱数生成アルゴリズムの実装プログラム
  - SmartMIPSアセンブリ言語による実装(約240行)
  - モンゴメリ乗算アルゴリズムを使用
  - 疑似乱数生成器は暗号システムの重要な構成要素:
    - 鍵生成, ノンスの生成等

BBS アルゴリズム
x_0 ← seed; M ← modulus
$x_{i+1} = x_i^2 \bmod M$; $b_i = \mathrm{parity}(x_i)$

暗号プログラムの正しさの検証

BBSプログラムの正しさを数理的技法により証明

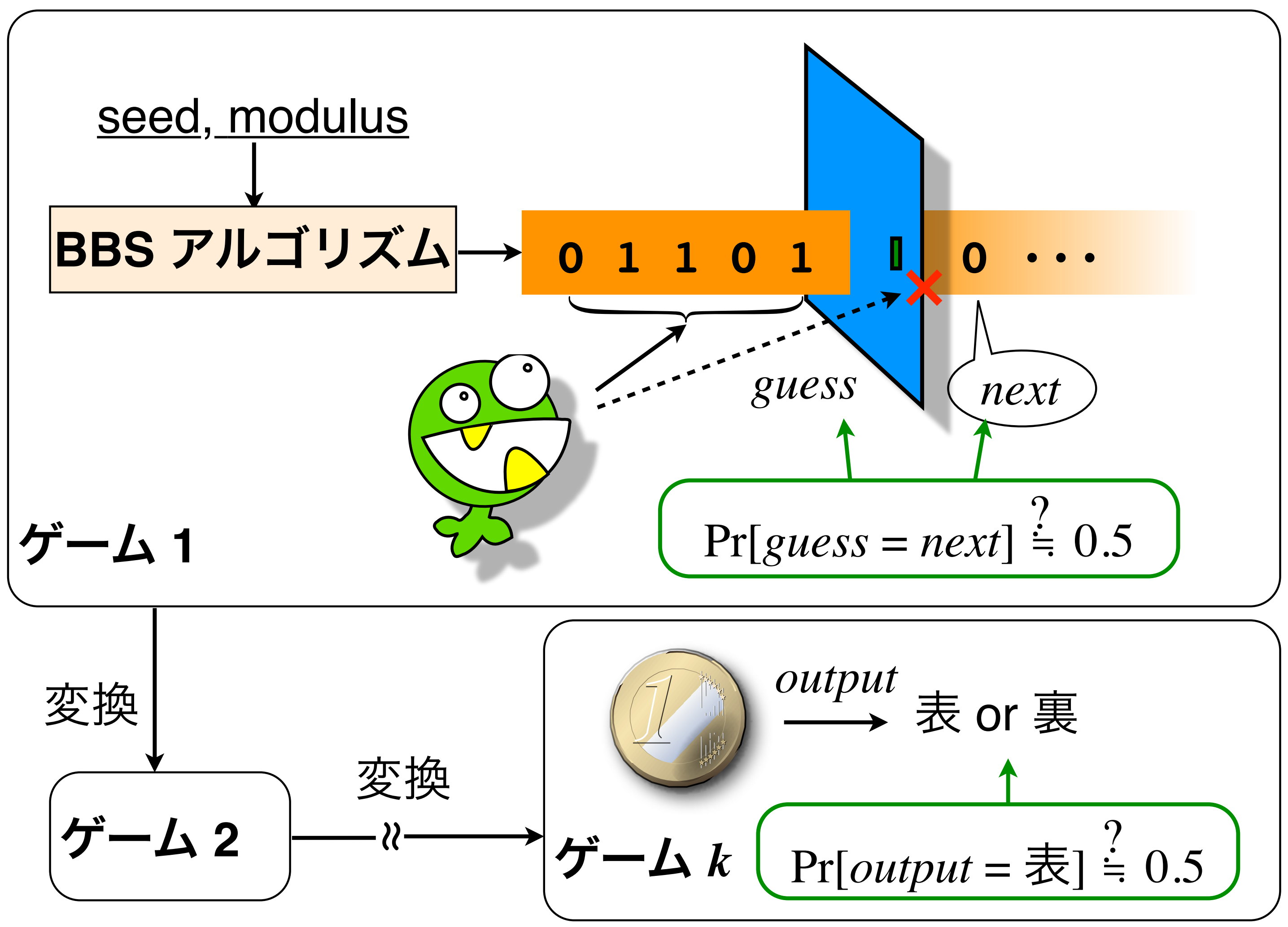
- 分離論理(ホーア論理の拡張)
- 正しさの証明されたプログラム変換器
  - ループを条件分岐命令へ



```
  0: addiu      i gpr_zero 0
  1: addiu      L l 0
  2: beq        i n 240
  3: addiu      j gpr_zero 0
  4: addiu      w gpr_zero 0
  5: beq        j thirtytwo 236
  6: mul_mod    k x x m . . .
     # compute X 2 (mod m)
230: lw         w_ 0 x
231: andi       w_ w_ 1
232: sllv       w_ w_ j
233: cmd_or     w w w_
234: addiu      j j 1
235: jmp        5
236: sw         w 0 L
237: addiu      L L 4
238: addiu      i i 1
239: jmp        2
```

暗号アルゴリズムの安全性検証

- BBSの出力の予測不能性を数理的技法により証明
  ゲーム列による安全性証明



簡単なゲームへ帰着

各ゲームでPrはほとんど変わらない

統 合

暗号プログラムの安全性検証

- ゲーム列による安全性証明の拡張:
  ゲーム変換に新ステップを追加
  - SmartMIPSアセンブリコードを直接扱う
  - アセンブリプログラムの正しさの証明に立脚
- BBSへの適用:
  自動検証可能な安全性証明を得る

これからの取り組み

- プロセッサアーキテクチャの抽象的なモデル
  プログラムの正しさの証明の簡素化
- 異なるアーキテクチャのサポート: x86-64等
- BBSのより効率的な実装を対象とした安全性証明
- 疑似乱数源にBBSを用いたElGamal暗号の安全性検証:
  ElGamal暗号が利用する乱数の疑似乱数による置き換え

参考文献

Nowak D. **On formal verification of arithmetic-based cryptographic primitives**. *11th International Conference on Information Security and Cryptology (ICISC 2008), LNCS 5461.*
Nowak D. and Yamada K. **Towards a certified implementation of a cryptographically secure pseudorandom bit generator**. *11th JSSST Workshop on Programming and Programming Languages (PPL 2009).*
Affeldt R., Nowak D. and Yamada K. **Certifying Assembly with Formal Cryptographic Proofs: the Case of BBS**. *Submitted for publication.*

http://www.aist.go.jp
National Institute of Advanced Industrial Science and Technology AIST
独立行政法人
産業技術総合研究所

National Institute of
Advanced Industrial Science
and Technology
AIST

Research Center for
Information Security

Certifying Assembly with Cryptographic Proofs

〈暗号学的安全性証明によるアセンブリプログラムの安全性検証〉

Reynald AFFELDT, David NOWAK, and 山田 聖
レナルド アフェルト, ダヴィット ノヴァック, やまだ きよし

情報セキュリティ研究センター
ソフトウェアセキュリティ研究チーム

研究の特徴

- **暗号アルゴリズムの安全性検証技術** と **プログラム検証技術** を **統合**

  - アセンブリ言語で記述された**暗号プログラムの暗号学的安全性検証** ができる

  - 定理証明支援ソフトウェアの利用により，**厳密な検証**ができる

研究の背景

暗号プログラムの安全性検証の必要性

機密情報をやりとりする
製品・サービスが普及

暗号プログラム

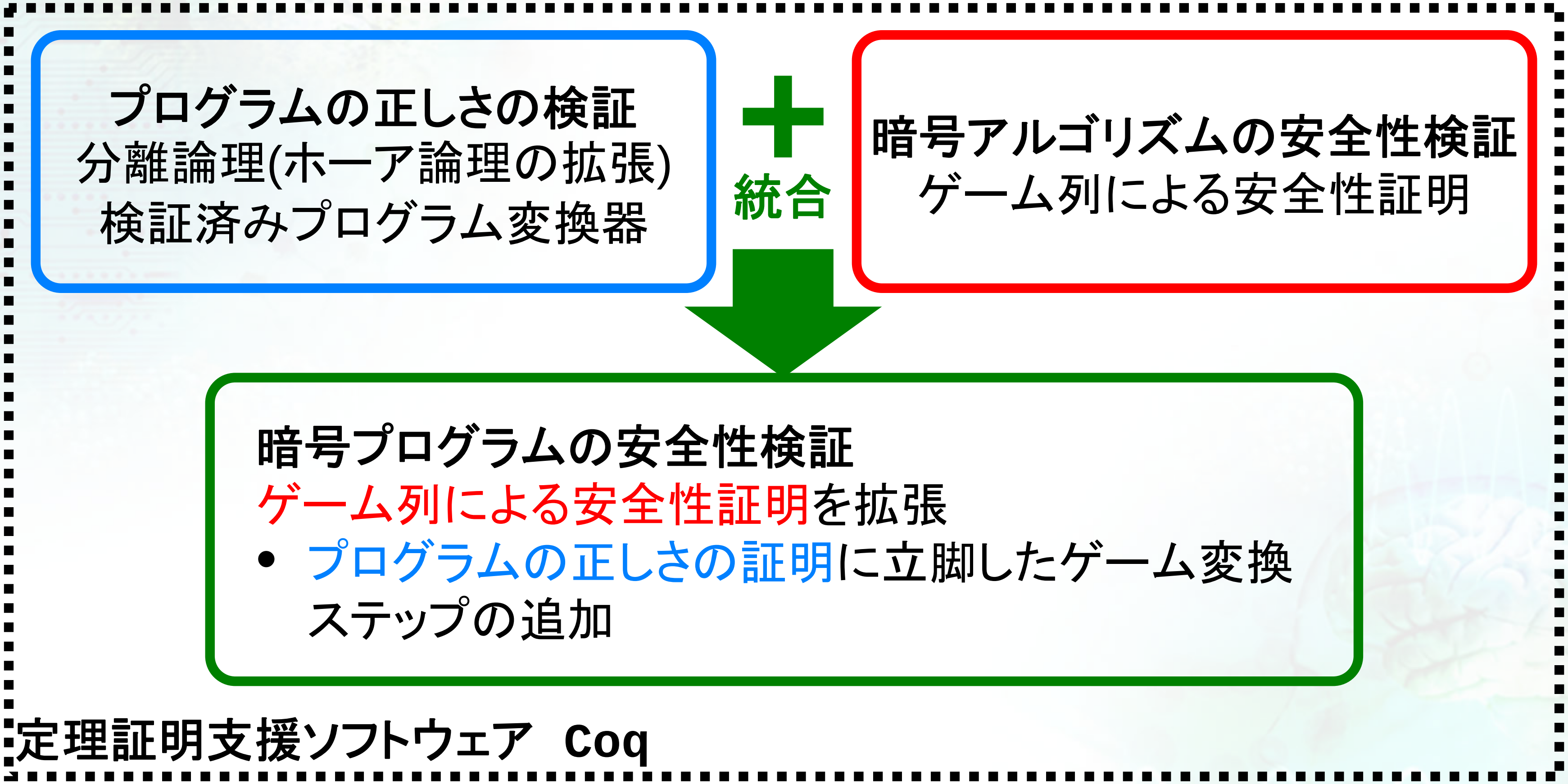
効率のためアセンブリ言語で記述されがち

- 暗号学的安全性の検証は行われていない
- 多くの製品で利用されているため，
  不具合が発見された場合の
  社会的・経済的影響が大きい

数理的技法(形式手法)による厳密な暗号学的安全性検証を
暗号プログラムに適用した前例がない！

研究の概要

■暗号アルゴリズムの安全性検証とプログラム検証の統合



プログラムの正しさの検証
分離論理(ホーア論理の拡張)
検証済みプログラム変換器

＋
統合

暗号アルゴリズムの安全性検証
ゲーム列による安全性証明

暗号プログラムの安全性検証
ゲーム列による安全性証明を拡張

- プログラムの正しさの証明に立脚したゲーム変換ステップの追加

定理証明支援ソフトウェア　Coq

事例研究

- 疑似乱数生成器Blum Blum Shub(BBS)の実装の安全性検証
  - 疑似乱数生成器:

    暗号システムの重要な構成要素 (鍵生成, ノンス生成等)

    安全性: 次ビット予測不能性

  - BBSの実装プログラム:

    SmartMIPSアセンブリで記述(約240行)

  - 結果:

    自動検証可能な安全性証明を得る

BBS アルゴリズム

$x_0 \leftarrow$ seed; $M \leftarrow$ modulus

$x_{i+1} = x_i^2 \bmod M$; $b_i =$ parity(x_i)

安全性
検証済み