評価結果の比較

ツール・評価者によって結果が異なる

※予稿の表をご覧ください

差異の原因

1. ツールの特性
   a. ツールで評価可能な安全性の定義の違い
   b. ツールが異常終了したり、停止しない
   c. 安全性の前提や数学的演算がうまく扱えない
2. 評価者のプロトコル・安全性の解釈
   a. プロトコル仕様の解釈
   b. モデル化の方法
   c. 選択する安全性

差異の原因

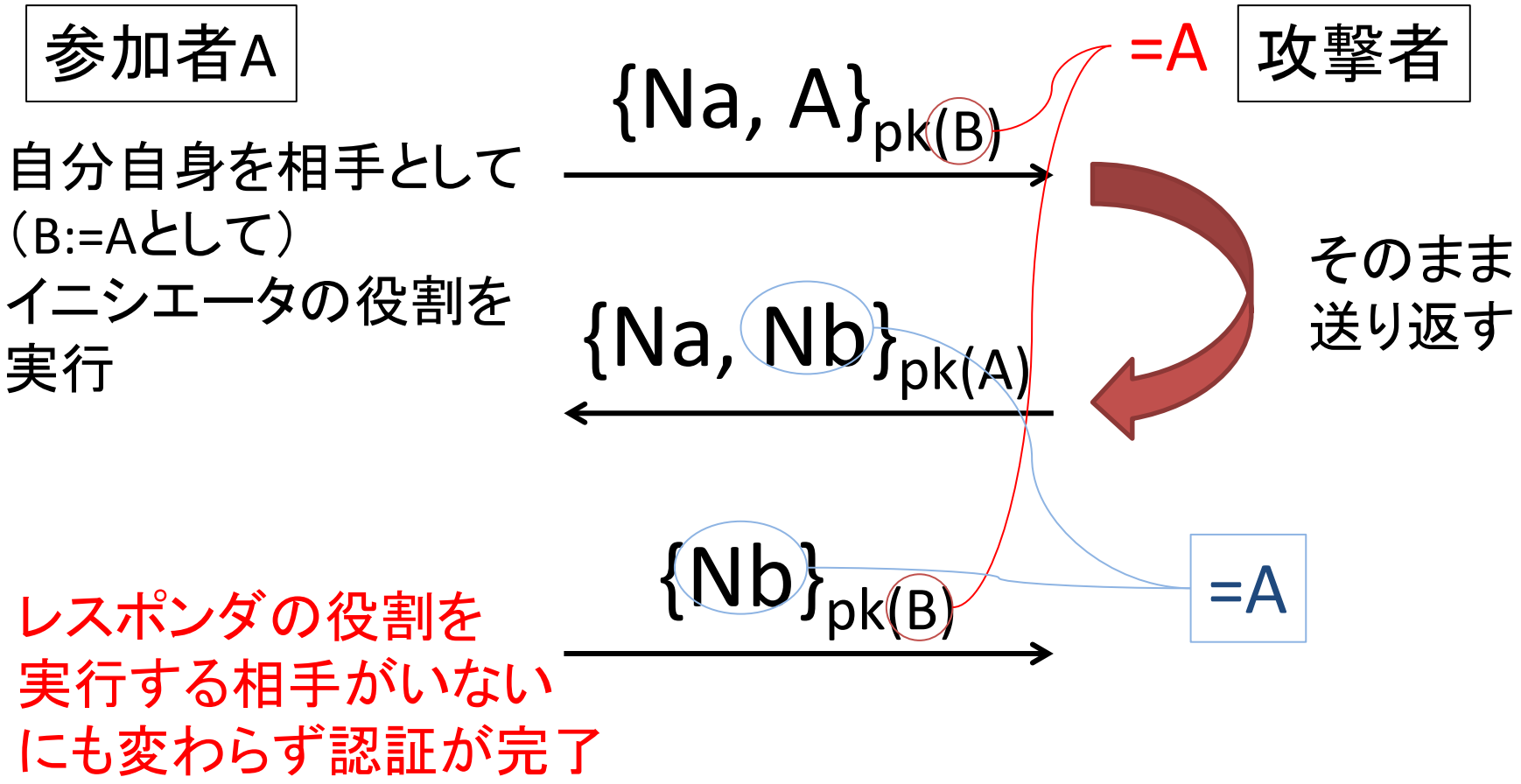
1. ツールの特性
   a. ツールで評価可能な安全性の定義の違い
   b. ツールが異常終了したり、停止しない
   c. 安全性の前提や数学的演算がうまく扱えない
2. 評価者のプロトコル・安全性の解釈
   a. プロトコル仕様の解釈
   b. モデル化の方法
   c. 選択する安全性

ツールごとの安全性の定義の違い

- 認証プロトコルの安全性に、なりすましだけでなく交換したノンスや鍵の一致も含む（ProVerif, Scyther）
- 認証プロトコルの安全性に、Reflection attackへの耐性も含む（ProVerif, Scyther）

- 秘匿性に、他のセッションで鍵が漏れた場合の秘匿性も含む （KDDI)

NSPKへのReflection Attack



これが攻撃として意味があるかはケースバイケース

差異の原因

1. ツールの特性
   a. ツールで評価可能な安全性の定義の違い
   b. ツールが異常終了したり、停止しない
   c. 安全性の前提や数学的演算がうまく扱えない
2. 評価者のプロトコル・安全性の解釈
   a. プロトコル仕様の解釈
   b. モデル化の方法
   c. 選択する安全性

ツールが異常終了、終了しない

- ProVerifは近似的なモデルで評価を行うため、実際には実行できない攻撃まで検出
  そのような場合には「失敗」を出力

- ProVerifでプロトコルの並列実行数を制限しない場合に、数時間待っても評価が終了しない場合があった

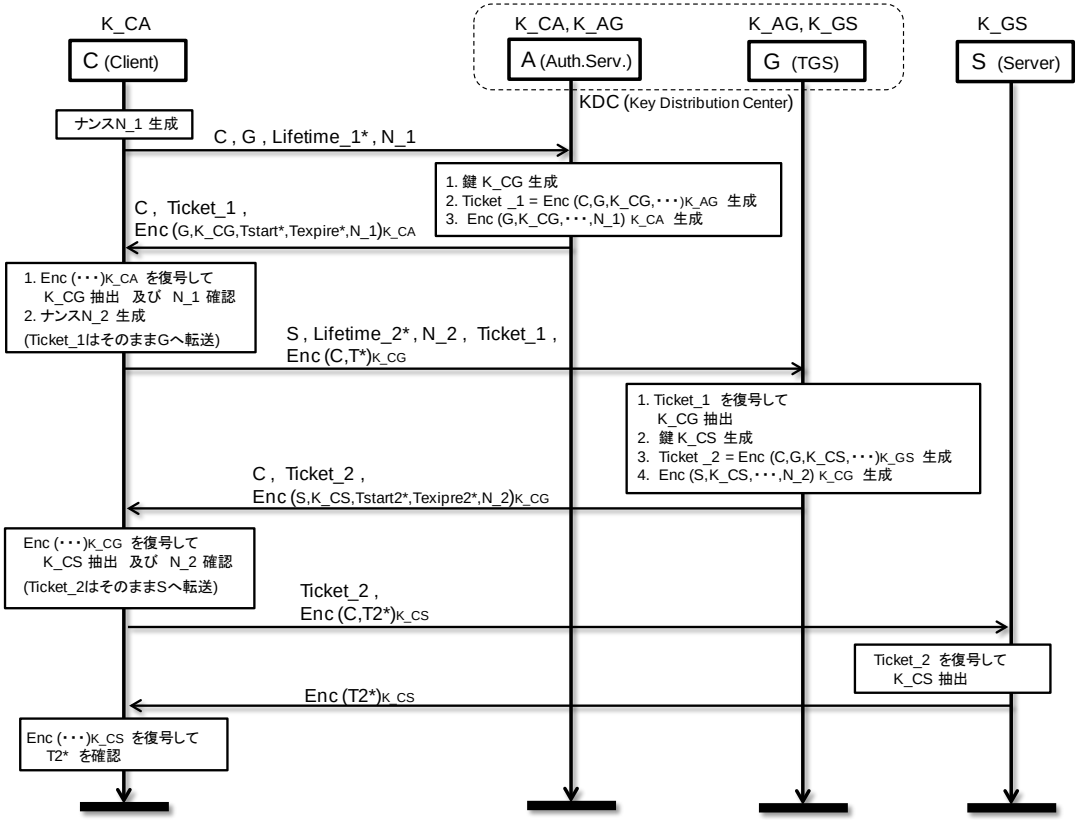
評価結果がツール・評価者で異なる原因

1. ツールの特性
   a. ツールで評価可能な安全性の定義の違い
   b. ツールが異常終了したり、停止しない
   c. 安全性の前提や数学的演算がうまく扱えない
2. 評価者のプロトコル・安全性の解釈
   a. プロトコル仕様の解釈
   b. モデル化の方法
   c. 選択する安全性

安全性の前提がうまく扱えない

- Scytherでは、複数の参加者を信頼するプロトコルを（工夫なしに）扱えない

Kerberosでは、認証サーバ・TGS・ログインするサーバを信頼



数学的演算がうまく扱えない

- どのツールも、整数や群の演算は近似的にしか扱えないため、結果に違いが生じる

  また、結果の正しさ（健全性）も検討が必要

差異の原因

1. ツールの特性
   a. ツールで評価可能な安全性の定義の違い
   b. ツールが異常終了したり、停止しない
   c. 安全性の前提や数学的演算がうまく扱えない
2. 評価者のプロトコル・安全性の解釈
   a. プロトコル仕様の解釈
   b. モデル化の方法
   c. 選択する安全性

プロトコル仕様の解釈

認証プロトコルEAP-AKAでは複数のハッシュ関数・MACを使う

- ProVerifによる評価では、これらに相関があると仮定（f1(Kpa, at_rand)からf2(Kpa, at_rand)などが計算可能と仮定）　⇨安全でない
- 他の評価ではそのようなことはないと仮定（単に別の関数として定義）　⇨安全

評価結果がツール・評価者で異なる原因

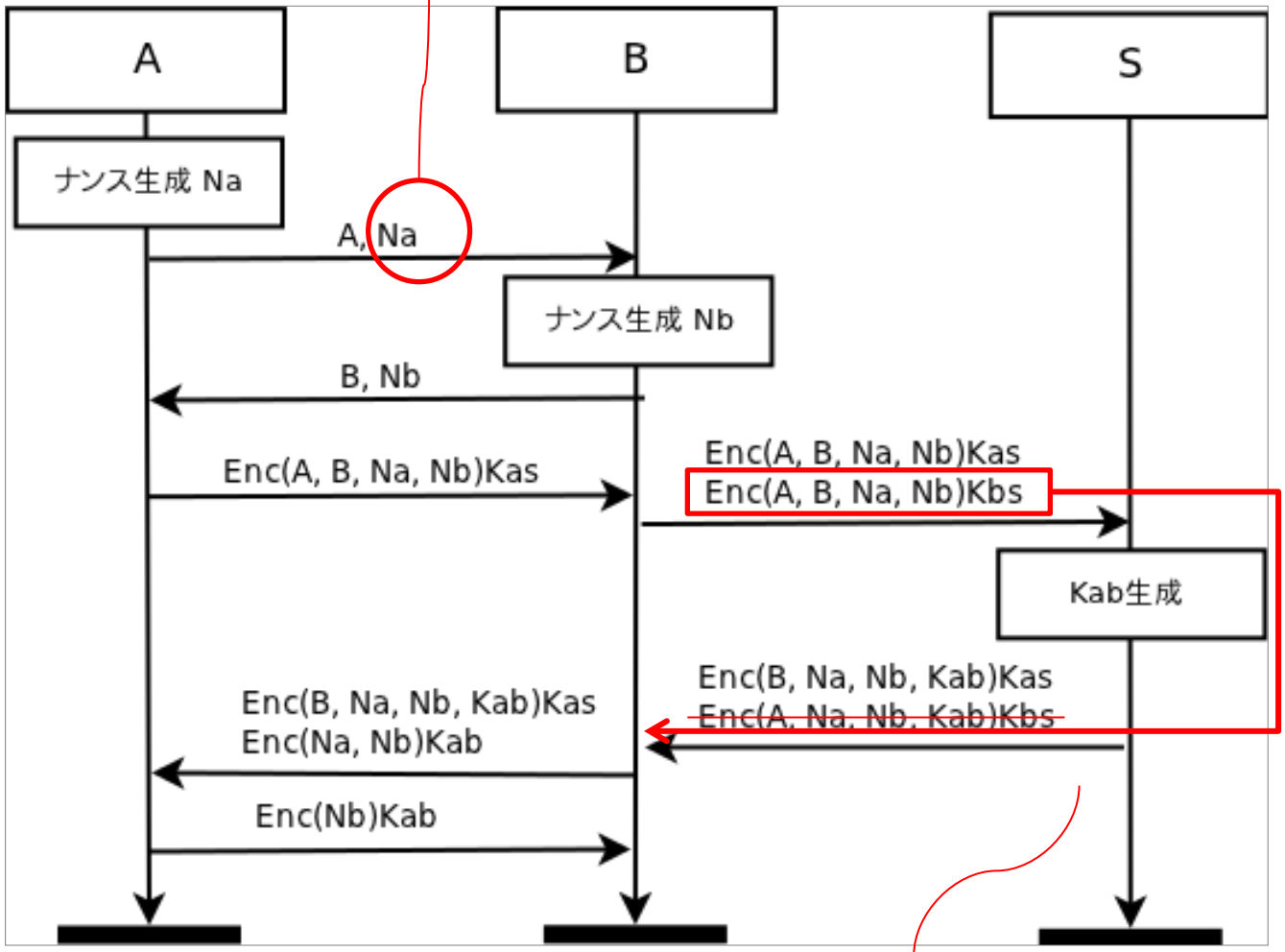
1. ツールの特性
   a. ツールで評価可能な安全性の定義の違い
   b. ツールが異常終了したり、停止しない
   c. 安全性の前提や数学的演算がうまく扱えない
2. 評価者のプロトコル・安全性の解釈
   a. プロトコル仕様の解釈
   b. モデル化の方法
   c. 選択する安全性

モデル化の方法

- 各ツールともデータに型（ノンス型、参加者名等）を与えることができ、参加者は型チェックによりメッセージを廃棄
- 例えば、ノンスと参加者名を別の型とする場合よりも、同じ型（ビット列型）とする場合のほうがより多くの攻撃が可能になる
- 型のチェックを実際に行うかはプロトコル仕様・実装により異なる

例：Woo-Lam相互認証

Naの代わりにBを送信（ただしBが型チェックをしない場合のみ可能）



そのまま送り返すとNbをセッション鍵Kabとして使用させられる

差異の原因

1. ツールの特性
   a. ツールで評価可能な安全性の定義の違い
   b. ツールが異常終了したり、停止しない
   c. 安全性の前提や数学的演算がうまく扱えない
2. 評価者のプロトコル・安全性の解釈
   a. プロトコル仕様の解釈
   b. モデル化の方法
   c. 選択する安全性

選択する安全性

- 鍵交換プロトコルIKE（v1, 署名による認証）の安全性として、ProVerifによる評価ではreflection attackを考慮に入れなかった、
  一方、Scytherではこれを考慮に入れた

差異の原因

1. ツールの特性
   a. ツールで評価可能な安全性の定義の違い
   b. ツールが異常終了したり、停止しない
   c. 安全性の前提や数学的演算がうまく扱えない
2. 評価者のプロトコル・安全性の解釈
   a. プロトコル仕様の解釈
   b. モデル化の方法
   c. 選択する安全性

⇨複数の評価を併用・比較するのが適切