本資料について

本資料は下記著書を基にして作成されたものです。
著書の内容の正確さは保障できないため、正確な知識を求める方は原本を参照してください

- 著書名　ワイヤード/ワイヤレスネットワークにおけるブロードキャスト通信のセキュリティ
- 著者　Adrian Perrig, J.D.Tygar
- 翻訳　溝口　文雄
- 出版社　共立出版株式会社

ブロードキャスト通信の セキュリティ

渡邊研究室
02j056　後藤　裕司

はじめに

- ブロードキャストの長所は一つのパケットを何百万もの受信者に送信できることである。逆に、この偉大なる特性は多大なる危険性をもたらす原因となる。
- たった一つの悪意のあるパケットで、コンピュータをロックし、再起動させてしまうことも可能である。
- 効果的なブロードキャスト認証は早急な問題である。

ブロードキャスト通信

- ブロードキャストアプリケーション
  - インターネットによるコンテツ配信(デジタルTV、ラジオ)
  - ソフトウェアの配信(ソフトウェアのアップデートなど)
  - 流通、車両管理(タクシーの正確な位置情報)
  - パーソナルワイヤレス通信(携帯などで情報を受信)
  - ホームオートメーション(家電などを管理)
  - 金融マーケット(金融市場情報)
  - マルチプレーヤーゲーム

暗号の基礎1

- 対称暗号
  - 送信者と受信者は同一の鍵(秘密鍵)を共有
  - 非対称暗号よりも3桁から5桁のオーダで早い
- 非対称暗号
  - 公開鍵と秘密鍵を利用する
  - 送信者は秘密鍵で署名し、受信者は公開鍵で検証
- 一方向関数
  - 簡単に計算することができるが**逆を計算することは計算上実行不可能**

暗号の基礎2

- MAC (message authentication code)
  (メッセージ認証コード)
  - 受信者はメッセージが要求した送信者からのものであることを検証することができる

  検証方法

  1) 送信者と秘密鍵を共有する

  2) 送信者が送信するすべてのメッセージに共有鍵で計算された認証タグ (またはMAC) を加える

  3) 受信者は共有鍵を利用してMAC関数を計算して認証タグが正しいことを検証する

暗号の基礎3

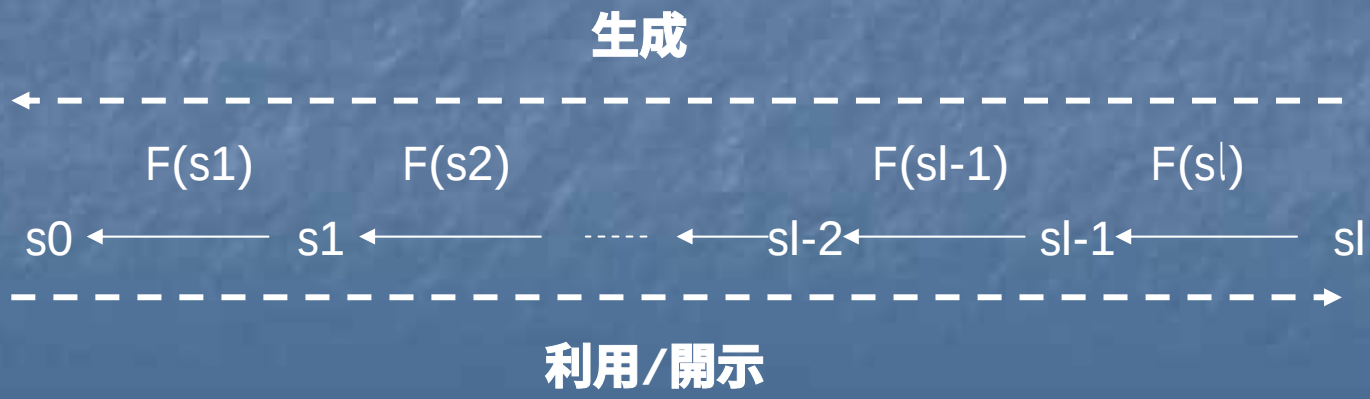
- コミットメントプロトコル
  - コミットメント関数は秘密sをあかさずにsを封印する
  - 秘密の値sを選択し、c=F(s)を計算し、cを公開
  - 後でsを公開することで選択したsを明らかにできる
- 自己認証値
  - 付加的な情報なしで受信者が認証できる値
  - コミメットプロトコルを利用
  - 受信者が認証されたチャネル上で、秘密sへのコミットメントcを得るならば秘密sは自己認証値
  - 受信者はF(s)とcが等しいことを検証することによって、即座に認証することができる

暗号の基礎4

- 一方向チェーン
  - コミットメント関数を利用して生成
  - 中間の値がロスしても、その後に続く値を使って再計算をすることができる。

生成方法（例：長さlのチェーン）

- チェーンの最後の要素であるslをランダムに選ぶ
- 一方向関数Fを繰り返し適応することでチェーンを生成
- 逆の順番で値を開示



s0を通じてチェーンの任意の要素を検証することができる

ブロードキャスト通信の課題

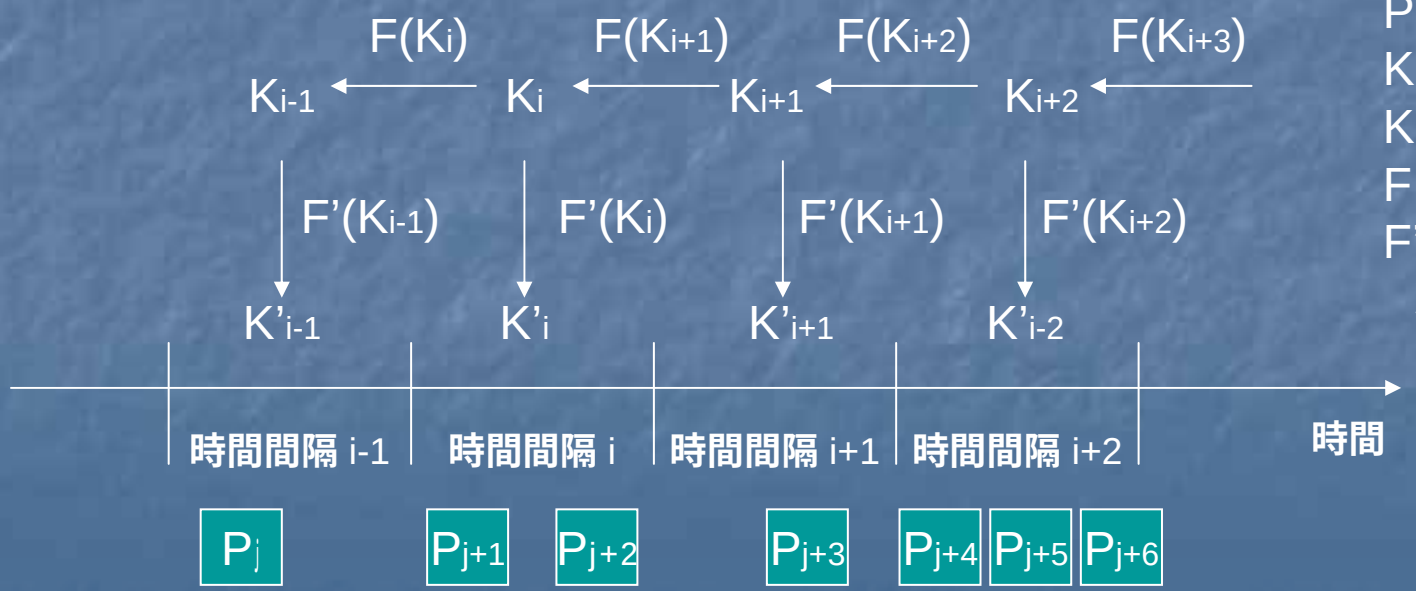
- 信頼性(reliability)
  - 大規模なブロードキャストではデータの再送はしない
- 受信者の異種性(receiver heterogeneity)
  - 帯域幅や計算能力の違い
- 輻輳制御(congestion control)
  - 輻輳状態にならないように通信を制御
- セキュリティ(security)
  - 多くの受信者に対応できない
  - セキュアでない
  - 計算や通信のオーバーヘッド
  - パケットロスに対して頑丈でない

TESLA (timed efficient stream loss-tolerant authentication)

- ブロードキャスト認証のためのプロトコル
- セキュリティの課題をすべて低いコストで満たすことができる
- 以下の要請を必要
  - 送信者と受信者は、緩い時間同期が必要
  - 受信者もしくは送信者は、メッセージをバッファする必要がある。

TESLAの動作

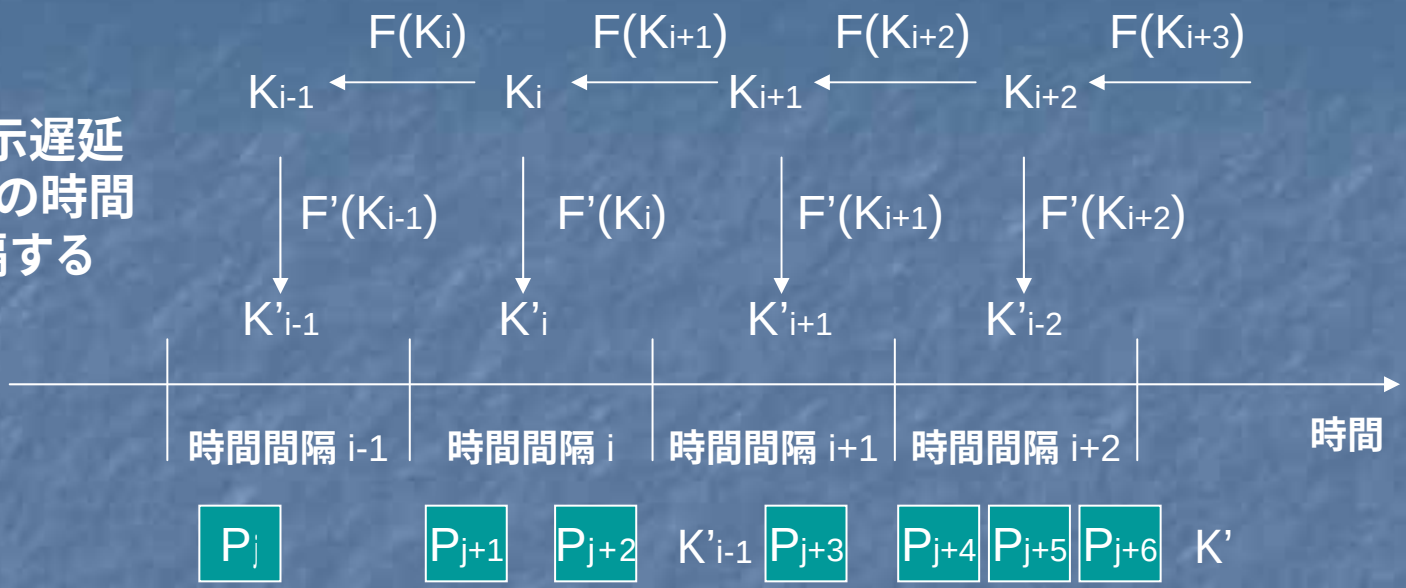
- 送信者
  - 一定時間の間隔に時間を分割
  - 一方向チェーンの値の開示時間を定義
  - 自己認証値の一方向チェーンを形成（生成の逆順に利用）
  - 各パケットにMACを付加
  - パケットとともに開示可能な最新の一方向チェーンの値も送信



P :パケット
K :一方向チェーン（鍵）
K'：MAC鍵
F :一方向関数
F':疑似乱数関数族を利用した一方向関数

TESLAの動作

鍵開示遅延を2つの時間間隔する



すべての受信者

検証可

検証可

鍵がまだ秘密であるかチェックし、パケットをバッファリング

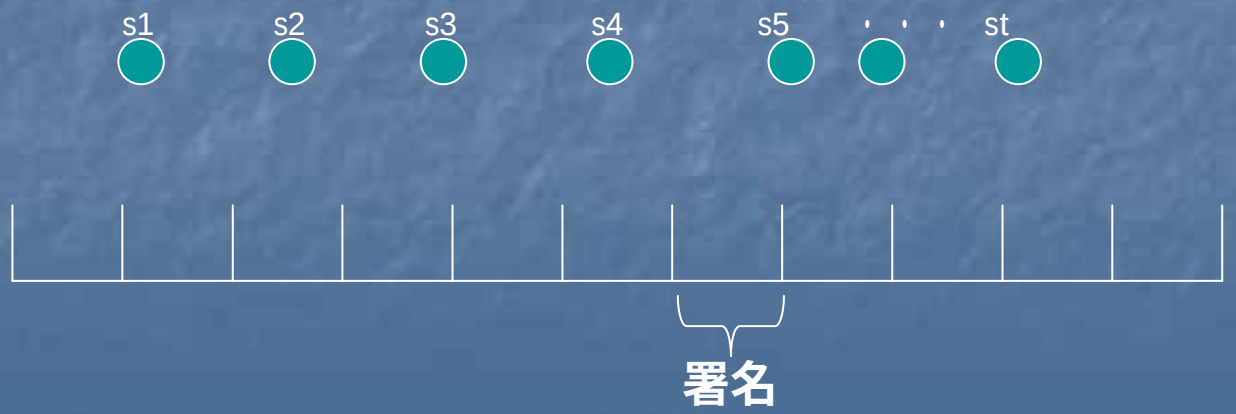
パケットロスしても、開示された鍵を利用して以前の鍵を計算することができ安全なパケットと検証することができる

BiBa (BinsとBalls)

- 特性
  - 認証情報の高速生成
  - 受信者による高速検証
  - 即時認証 (パケットのバッファリングをしないのでリアルタイムデータに有効)
  - パケットロスに対する頑丈性
  - 完璧なスケーラビリティ
  - 適切な通信オーバーヘッド (各パケットにおいて100バイト)
- 欠点
  - TESLAと同様、送信者と受信者の間で緩い時間同期が必要

BiBa　署名

- メッセージに署名するためにハッシュを計算
- 各ボールは自己認証値
- 署名者がビンの集合にランダムに多数のボールを投げる
- 衝突に関係するボールが署名
- 攻撃者は署名者が署名で開示したボール群だけ知り得る
- 署名の検証は非常に軽量

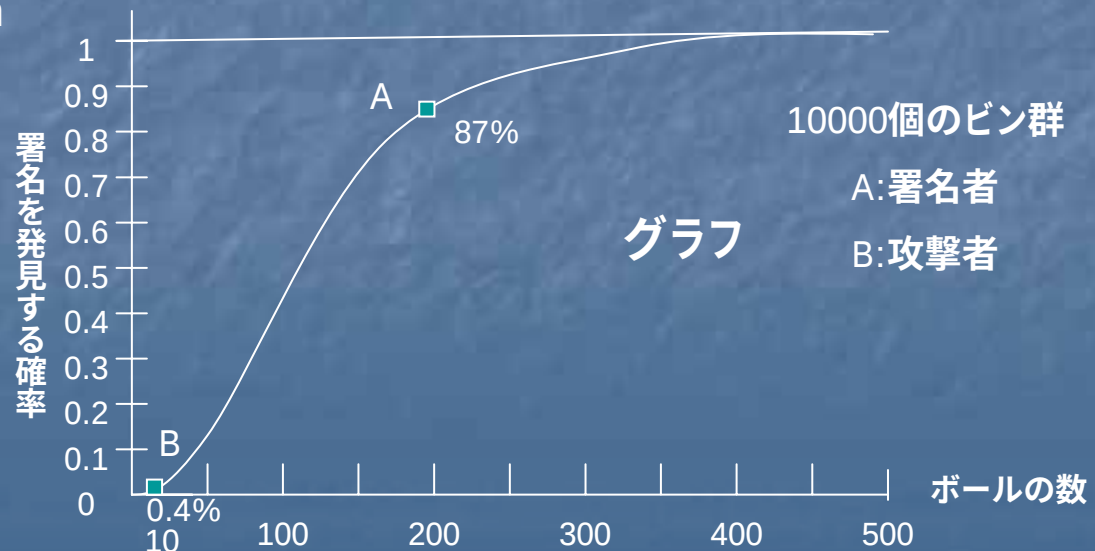


BiBa セキュリティ

- 署名者がt個のボール群を持ち、メッセージmがあたら得られた時、BiBa署名を見つけ出す確率は少なくとも一つの2方向衝突を発見する確率と等しくなる。

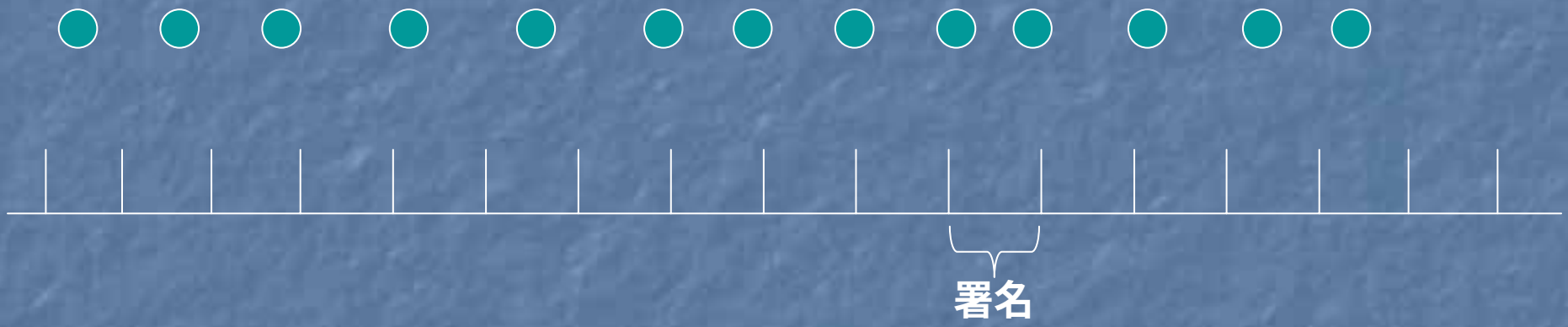
  Pc：少なくとも一つは衝突が起こる確率

$$Pc \sim 1-e^{\frac{t(t+1)}{2n}}$$

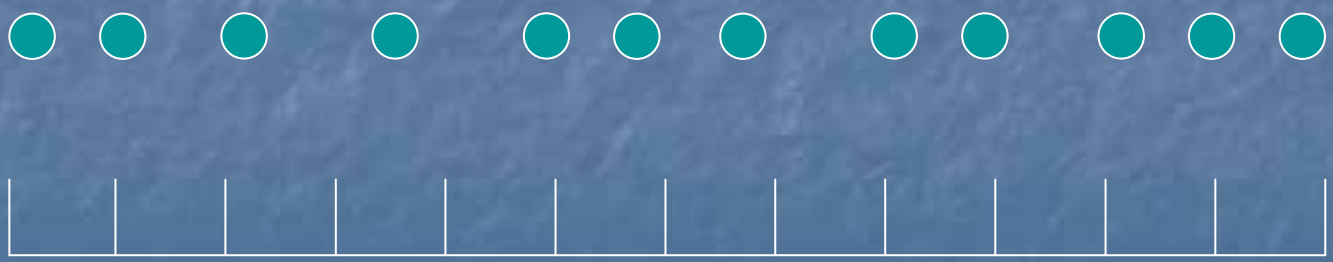


BiBaの拡張

- セキュリティ強化
  - ボール群とビン群の数を増加させる

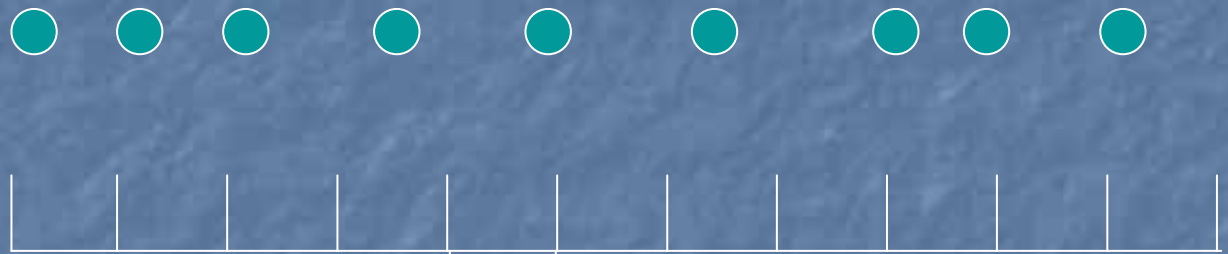


  - 複数の2方向衝突を用いる

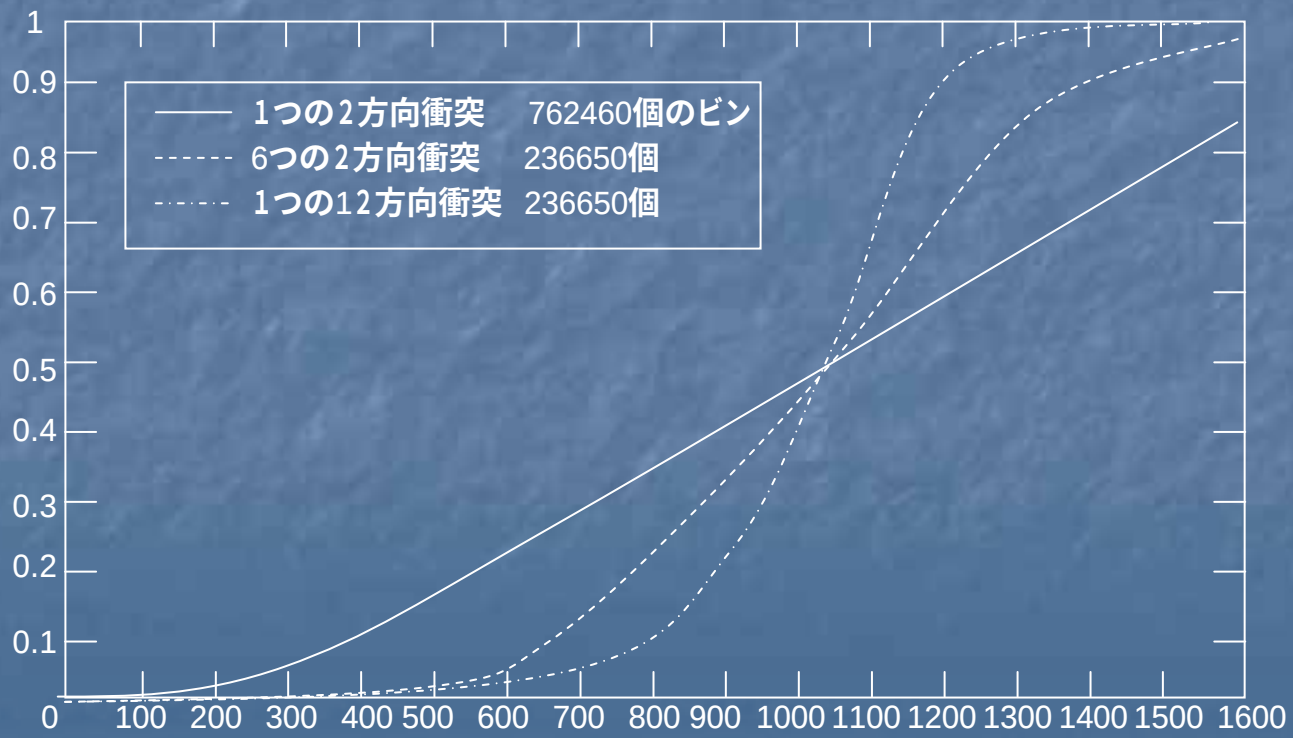


BiBaの拡張　続き

- 2方向衝突の代わりに多方向衝突を用いる



署名

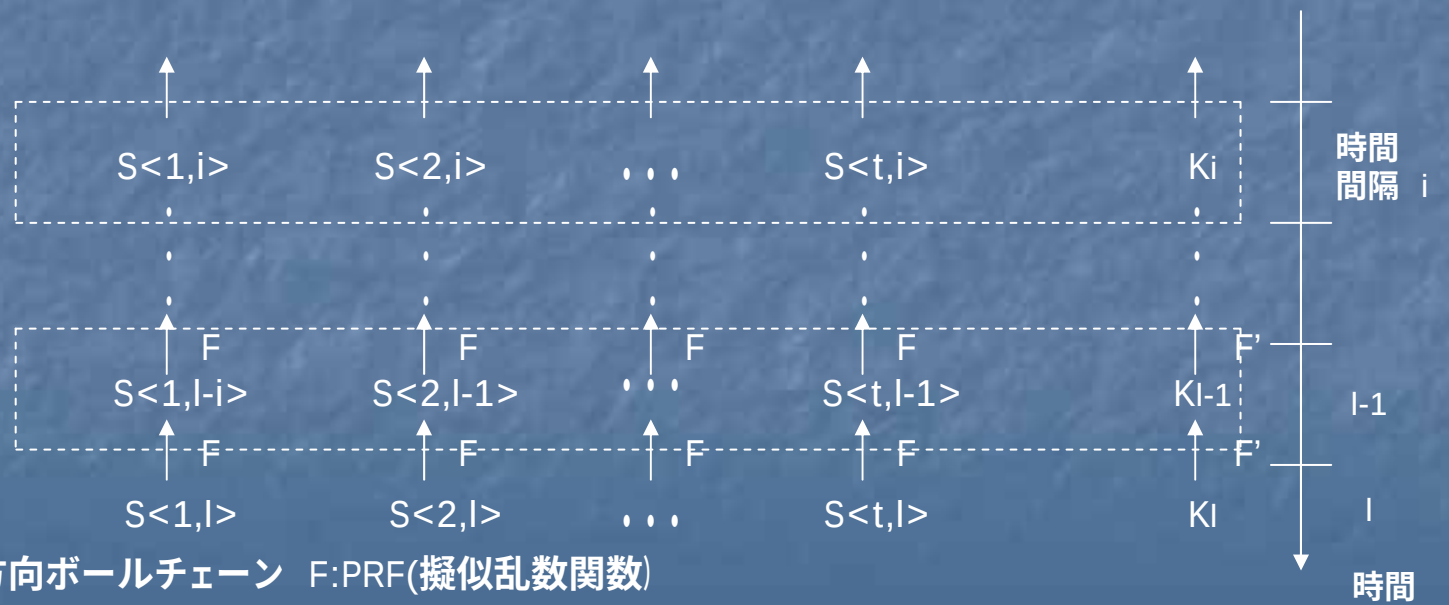


BiBa ブロードキャスト認証プロトコル

- 受信者が送られてきたデータを検証できる必要がある
  - わかりやすい方法では、送信者がブロードキャストする各メッセージ上のBiBa署名を計算することである。
  - 送信者は少数のボールしか開示できないので、少数のメッセージしか署名できない
  - 送信者は潜在的に無限のメッセージの流列を認証しなければならない
- ボール群を受け取ったときに受信者がそれらのボール群を即座に認証し、自動的にボールを補充可能な方法を必要とする

BiBa ブロードキャスト認証プロトコル

- ボールの自己認証属性の実現と、補充のために一方向チェーンを利用
- ボール群の値を求めるためソルト群の値を利用
- アクティブなボールのうちBiBa署名に関与するものだけ開示
- 攻撃者が、事前計算によりボール群における他の前像を発見する攻撃を和らげる

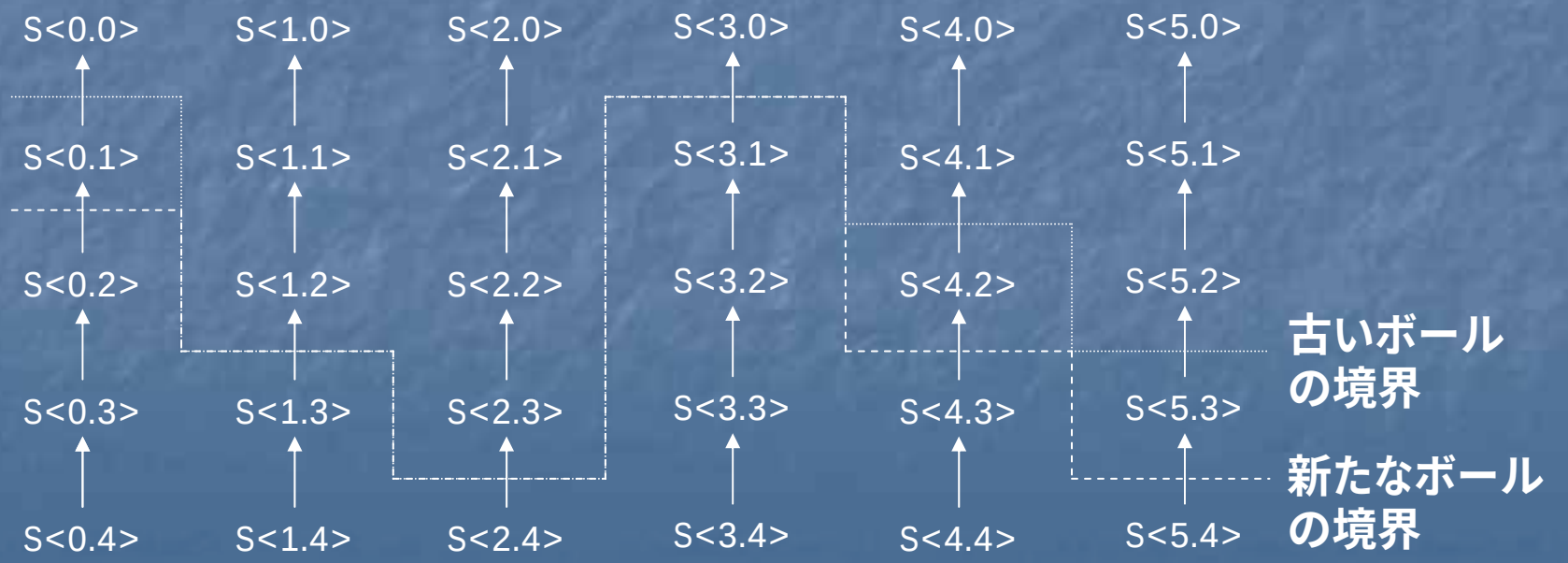


S:一方向ボールチェーン　F:PRF(擬似乱数関数)

K:一方向ソルトチェーン

BiBa ブロードキャスト認証プロトコル 拡張

- 境界上のボールは開示される
- 送信者、受信者は境界の値を知っている
- 境界下に直接隣接しているボール群のみ使用
- BiBa署名後に新たに開示されたボール群の下にボール境界を拡張



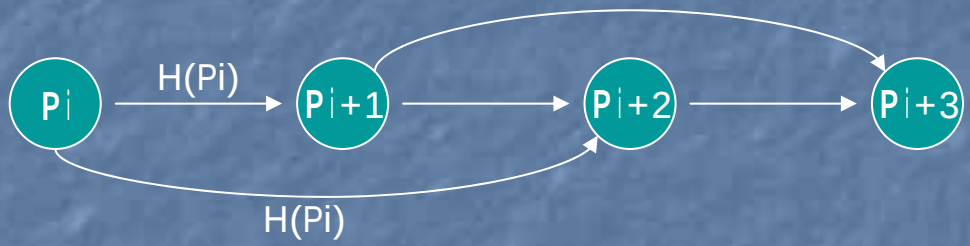
EMSS (efficient multicast stream signature)

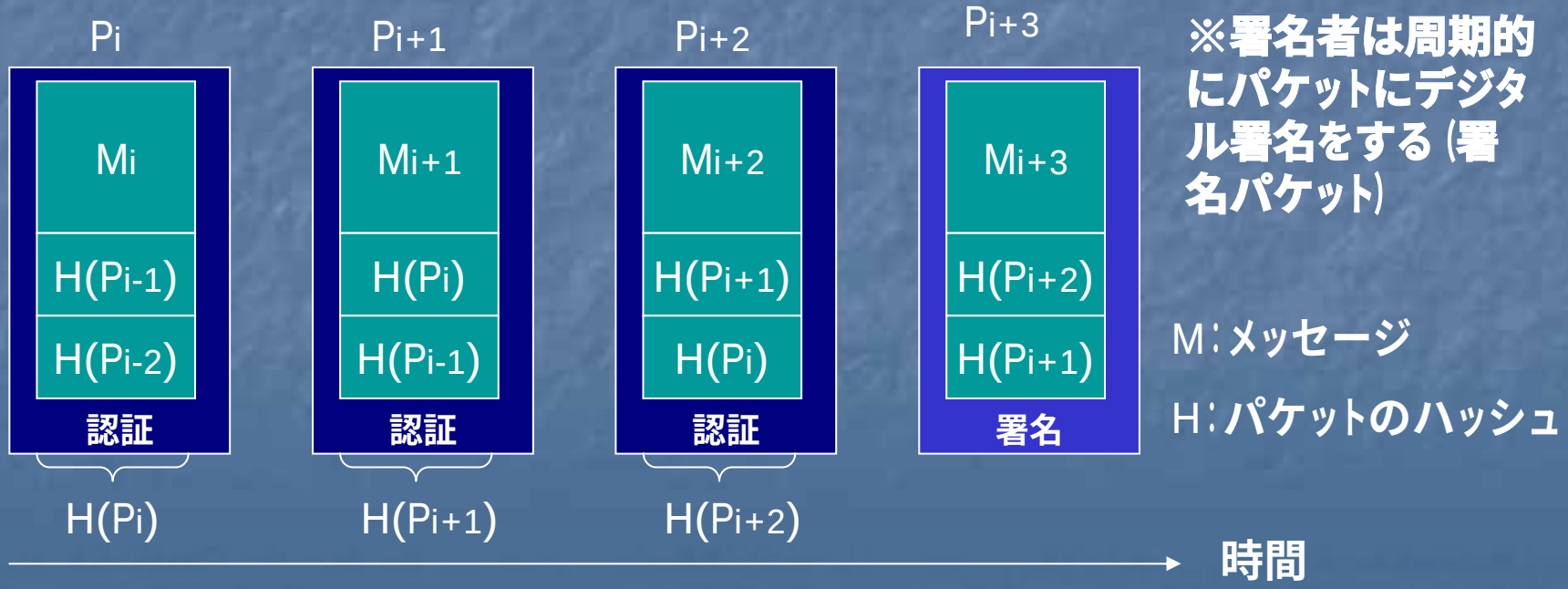
効果的なマルチキャストストリーム署名

- 特性
  - 認証情報の生成、検証に対する低オーバーヘッド
  - 通信オーバーヘッドの低減
  - バッファリングを必要としない
  - ここのメッセージに対する即時の署名認証の実現
  - パケットロスに対する頑健性
  - 多数の受信者への対応

EMSS　認証方法

- 一定のハッシュリンクの距離を用いる
- パケットのハッシュリンクを後方のパケットに埋め込む





※署名者は周期的にパケットにデジタル署名をする（署名パケット）

M：メッセージ

H：パケットのハッシュ

比較

	TESLA	BiBa	EMSS
認証 (A)署名 (S)	A	A	S
時間同期	○	○	×
認証遅延	○	×	×
送信者のバッファリング	×	×	○
受信者のバッファリング	○	×	×
リアルタイムストリーム	○	○	×
パケットロスの頑健性	○	○	×
通信のオーバーヘッド(bytes)	24	128	32
生成時のオーバーヘッド	1	2048	2
検証時のオーバーヘッド	2	100	10

センサネットワークのセキュリティ

- ワイヤレスセンサネットワークは将来において広く開拓されていく。
- 多くの研究はこのようなネットワークを実現したり使いやすくすることに主眼を置いており、あまりセキュリティに対しては関心を払っていない
- 本書では、セキュリティプロトコルSNEPとμTESLAを提供する

センサネットワークセキュリティの要件

- データの機密性
  - 近隣のネットワークにセンサの値を漏らしてはならない
  - 意図する受信者のみが保持する秘密鍵で情報を暗号化し、機密性を実現
- データ認証
  - データが正しい送信者によって送信されたものかどうかを受信者が検証できるようにする
- データの新規性
  - 敵が古いデータを再送していないことを保証する

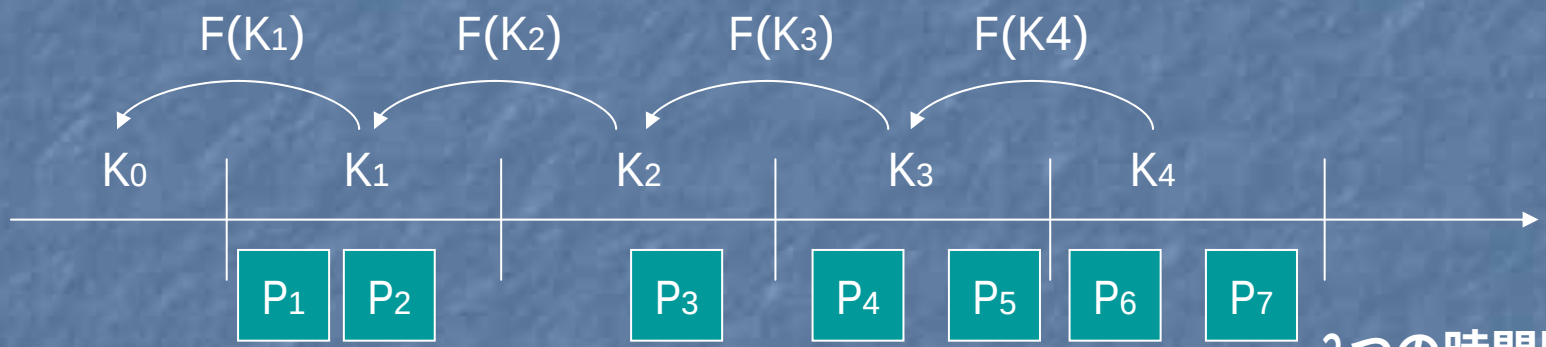
SNEP (Sensor Network Encryption Protocol)

- データの機密性、 2者間データ認証、完全性、新規性を実現
- 特性
  - セマンティックセキュリティ
    - 各メッセージの後でカウンタの値が加算されるので同じメッセージでも異なる暗号化がなされる
  - データ認証
    - メッセージ認証コード (MAC) を用いる
  - 再送攻撃からの保護
    - MACに含まれるカウンタ値は古いメッセージの再送を防止
  - 弱い新規性
    - メッセージが正しく認証されれば、古いカウンタ値と比較

μTESLA

TESLA		μTESLA
最初のパケットにデジタル署名	負荷が高い →	対象鍵を用いる
各パケットで鍵を開示	多大な電力を要する →	一定期間ごとに1回だけ鍵を開示
一方向チェーン	格納するには大きすぎる →	センサの認証数を制限

μTESLAの動作



2つの時間間隔が経過した後で鍵を開示

$K_0 = F(F(K_2))$

K_2　　$K_1 = F(K_2)$

すべての受信者

受信者は認証されたKを使いパケットを検証する

未解決問題

- TELSAプロトコルはブロードキャスト認証を超えてどの程度の適用範囲まで拡張できるか？
- BiBa署名をより効率よくできるか？公開鍵サイズを縮小できるか
- BiBa署名を別の暗号基本方式の開発のために拡張できるか？

おわり