アドホックネットワークにおける端末の信頼度を考慮した分散型公開鍵管理方式

立山 崇之　野口 拓　川合 誠
立命館大学理工学研究科

1 はじめに

固定インフラを必要とせず無線端末同士でネットワークを形成するアドホックネットワークは, ユビキタス社会への貢献が期待される一方で, ルーティングプロトコルや電力効率, セキュリティといった様々な課題に直面している. アドホックネットワークのセキュリティ分野の課題の 1 つに公開鍵管理の問題がある. アドホックネットワークにおける公開鍵管理には, 従来の CA(Certificate Authority) といった一点集中型の認証局を用いることは困難であり, 分散型の公開鍵管理方式が必要とされている. これまでいくつかのアドホックネットワーク向けの分散型公開鍵管理方式が提案されてきたが, その多くが公開鍵管理に参加する端末の信頼関係を基盤とし, 各端末が不正をしないことを前提としているため, 不正を行う端末が存在する状況ではうまく機能しない.[1]

本稿では, 既存の分散型公開鍵管理方式である証明書連鎖に信頼証明書という概念を導入することで, 個々の端末の信頼度の違いを考慮した公開鍵管理方式を提案し, ネットワークシミュレータを用いて検討を行った結果を報告する.

2 証明書連鎖

分散型公開鍵管理方式の 1 つに証明書連鎖 [2] がある. 証明書連鎖とは,CA のような一点集中型の認証局を用いることなく, 各端末が他の端末の公開鍵についての公開鍵証明書を自律的に発行し, それを交換し合うことで公開鍵管理を行う方式である. 集めた証明書をチェーンのようにたどることで他の端末の公開鍵認証を行う. 公開鍵証明書には証明書の発行者, 鍵の所有者, 公開鍵認証度などの情報が含まれており, 証明書連鎖の始点端末からみた終点端末の公開鍵認証度は, 証明書連鎖に含まれる公開鍵証明書の公開鍵認証度の積によって求められる. 公開鍵認証度とは, 発行者から見た公開鍵と公開鍵所有者の連結情報の自信度を数値化 (0.0〜1.0) したものである. 図 1 に証明書連鎖の例を示す. 図 1 の例では, 端末 S から見た端末 D の公開鍵の認証度は $0.9\times0.8\times0.7\times0.9=0.4536$ となる. 1 つの対象端末に対し複数の証明書連鎖が存在する場合も考えられ, その際はそれぞれの証明書連鎖ごとに認証度を求めることができる.

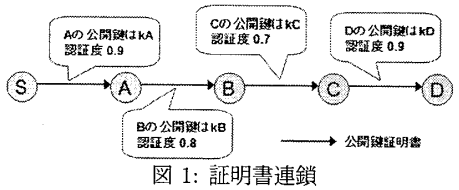

図 1: 証明書連鎖

2.1 証明書連鎖の問題点

証明書連鎖は信用できる第三者機関に依存せず, ネットワーク中の各端末が自律的に公開鍵管理を行うため, 参加する端末が正当な振る舞いをすることが前提となる. そのため, もし不正な証明書を発行する端末が存在すると証明書連鎖に不正な証明書が含まれる可能性があり, 証明書連鎖がうまく機能せず認証が失敗してしまう. またその場合, 証明書連鎖に不正な証明書が含まれるかどうかを見分けることはできない [1].

2.2 CKM の確率的減衰モデルを用いた証明書連鎖

CKM(Composite Key Management)[1] は, 証明書連鎖と仮想 CA[3] とを統合した公開鍵管理方式である.CKM では証明書連鎖の正当性がチェーンの長さによって変わる確率的減衰モデルが提案されている.CKM の確率的減衰モデルを用いた証明書連鎖では, ネットワーク内の端末が悪意がある確率 p(ここで $0\leqq p\leqq 1$) を減衰定数として与えると, 長さ d の証明書連鎖が正当である確率は $(1\text{-}p)^{d-1}$で表される.ある端末を対象とする長さ n の証明書連鎖に含まれる各証明書の認証度の値を$(V_1, V_2, \cdots, V_n)$ とすると, その証明書連鎖が示す対象端末の公開鍵の認証度は $V_1\times V_2\times\cdots\times V_n\times(1\text{-}p)^{n-1}$ となる.

3 提案方式

2 章で述べた CKM の確率的減衰モデルを用いた証明書連鎖は, 正当な証明書連鎖でも長さが長ければ, 正当性は低く見積もられ, 逆に不当な証明書連鎖でも, 長さが短ければ正当性は高く見積もられる. その結果, 不当な証明書連鎖の情報に基づき, 誤った公開鍵管理をしてしまう可能性があり, 証明書連鎖の問題点を完全に解決したとはいえない. そこで本稿では, 確率的減衰モデルではなく, 信頼証明書という概念を導入することで個々の端末の信頼度の違いを考慮した公開鍵管理方式を提案する.

3.1 信頼証明書を用いた公開鍵認証

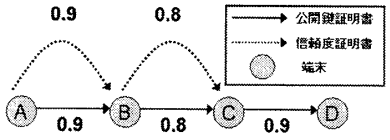

図 2: 提案方式

図 2 は提案方式のシステムモデルで, 実線矢印は証明書連鎖と同等の公開鍵証明書を表し, 点線矢印は信頼証明書を表している. 信頼証明書には証明書の発行者, 対象者, 信頼度などの情報が含まれており, 信頼度は対象者から見た対象者の振る舞いの信頼度を数値化 (0.0〜1.0) したものを表す. 例えば A と B の間の点線矢印は A が発行した B を対象とした信頼度 0.9 の信頼証明書を表す. 提案方式において, ある端末を対象とする長さ n の証明書連鎖に含まれる各公開鍵証明書の公開鍵認証度を $(V_1, V_2, \cdots, V_n)$, 各信頼証明書の信頼度を $(T_1, T_2, \cdots, T_{n-1})$ とすると, その証明書連鎖が示す対象端末の公開鍵の認証度は $V_1\times T_1\times V_2\times T_2\times\cdots\times V_{n-1}\times T_{n-1}\times V_n$ となる. 例えば図 3 の場合,CKM の確率的減衰モデルを用いた証明書連鎖 (p=0.1 とする) では, 証明書連鎖 (a) の認証度は 0.232, 証明書連鎖 (b) の認証度は 0.576 となり, 端末 X が発行した不当な証明書を含む証明書連鎖 (b) の方が認証度が高くなるが, 提案方式では, 証明書連鎖 (a) の認証度は 0.294, 証明書連鎖 (b) の認証度は 0.072 となり, 不当な証明書が含まれていれば長さ

"Distributed Public-Key Management Taking Node Trustworthiness into Consideration in Ad-Hoc Networks"
Takayuki Tateyama　Taku Noguchi　Makoto Kawai
Graduate School of Science and Engineering, Ritsumeikan University

が短い証明書連鎖でも認証度は低くなる. 提案方式では, 対象とする端末までの証明書連鎖の中で, 最も認証度の高い証明書連鎖の情報を元に, 端末とその公開鍵の結合を行う.

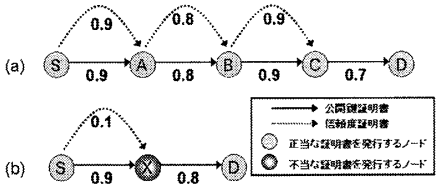

図 3: 正当な証明書連鎖と不当な証明書連鎖

3.2 信頼証明書の発行条件

信頼証明書は端末の振る舞いの信頼度を示すという性質上, 公開鍵証明書のように, 出会った相手を対象に即座に発行するのは現実的ではない. そのため, 各端末があらかじめ持っている端末間の友好関係に基づき発行する. 発行された信頼証明書は公開鍵証明書と同様に, ネットワーク上で出会った端末同士がそれぞれ持っている証明書を交換する. たとえば図 3 の場合, 各端末は表 1 のような友好関係に基づき信頼証明書を発行している.

表 1: 友好関係の例

端末	友好関係のある端末	信頼度
S	A	0.9
	X	0.1
A	B	0.8
B	C	0.9
...	...	...

3.3 信頼証明書が欠けた証明書連鎖について

提案方式では, 図 3 のように全ての端末間に信頼証明書が存在している状況が望ましい. しかし, 各端末の友好関係だけに基づいて信頼証明書を発行する場合, 友好関係の存在しない端末間では, 信頼証明書は発行されない. そのため図 4 のように証明書連鎖において, 信頼証明書が欠ける場合が考えられる. その場合, 信頼証明書が欠けている部分 (図 4 では端末 B と端末 C の間) に,0.5 という信頼度を持つ信頼証明書があるとみなして計算を行う.

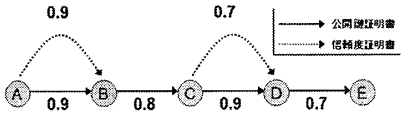

図 4: 信頼証明書が欠けた証明書連鎖

4 計算機シミュレーションによる性能評価

CKM の確率的減衰モデルを用いた証明書連鎖を比較対象とし,JiST/SWANS[4] を用いてシミュレーションによる検討を行った. 本シミュレーションでは前提条件として端末利用者間の友好関係をあらかじめ設定し, 不当な証明書を発行する端末 (不正端末) や友好関係のある端末 (友人端末) の数を変化させ, それに応じた正当結合率 (「公開鍵と所有者を正しく紐付けできた数／紐付けを行った総数」) と, 不正端末数が 5 のときの送信データサイズ (1 端末当たりの平均送信データサイズ) を測定した.

4.1 端末の振る舞いについて

本シュミレーションにおける普通端末の動作を以下に示す.

- 他端末についての公開鍵証明書を発行する際, 公開鍵と鍵の所有者の連結情報を正しく署名する.
- 普通端末の信頼度は 0.7, 0.8, 0.9 のいづれかとする.

また, 本シミュレーションでは, 不正な証明書を発行する端末 (不正端末) をネットワーク中に存在させている. 不正端末の動作を以下に示す.

- 不正端末は, 他端末についての公開鍵証明書を発行する際, 公開鍵と鍵の所有者の連結情報をでたらめに署名する.
- 不正端末の信頼度は 0.01 とする.

4.2 シミュレーション環境

表 2 にシミュレーション環境を示す.

表 2: シミュレーション環境

試行回数	各50回
シミュレーション時間	900s
総端末数	50個
端末帯域	1MByte/s
フィールドサイズ	1000m×1000m
初期配置	grid
移動性	RandomWayPoint (no pause)
移動速度	5m/s
通信可能範囲	半径100m
証明書交換条件	通信可能範囲に25秒以上滞在
1端末の友好関係数	10,20個
不正端末数	1,3,5,7,9個

4.3 シミュレーション結果

図 5 に測定した正当結合率を示す. どの状況でも提案方式のほうが高い正当結合率を示した. また図 6 に測定した送信データサイズを示す. 証明書連鎖に比べて提案方式の送信データサイズが多いという結果が得られたが, これは証明書連鎖には存在しない信頼証明書を交換し合うためであると考えられる. また, 友人数 20 の提案方式の送信データサイズは 371Byte/s であり, 帯域に比べて極めて小さい.

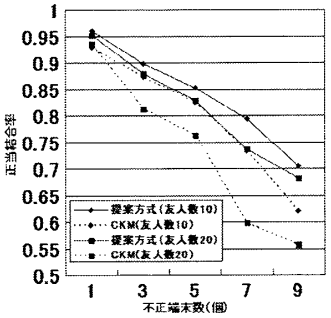

図 5: 正当結合率

方式	送信データ量(Byte/s)
提案方式 (友人数10)	189
提案方式 (友人数20)	371
CKM証明書連鎖 (友人数10)	110
CKM証明書連鎖 (友人数20)	289

図 6: 送信データサイズ

5 まとめ

本稿では, アドホックネットワークでの公開鍵認証において, ネットワークに参加する端末の振る舞いの信頼度を考慮した公開鍵認証方式を提案した. シミュレーションによる検討の結果, 提案方式は CKM の証明書連鎖に比べ, わずかに送信データサイズは増加するものの, 高い正当結合率を実現することを確認した. 今後筆者は,watchdog[5] などのノード監査を利用して, 動的に信頼証明書を発行する機能を実装・評価する予定である.

参考文献

[1] Seung Yi et al. ,"Composite Key Management for Ad Hoc Networks", MobiQuitous2004, pp.52-61,2004.

[2] Srdjan Capkun et al. ,"Self-Organized Public-Key Management for Mobile Ad Hoc Networks", IEEE TMC,Vol.2,No.1,2003.

[3] Bing Wu et al. , "Secure and efficient key management in mobile ad hoc networks", IEEE Network, pp.288-295,2005.

[4] http://jist.ece.cornell.edu/.

[5] Sergio Marti et al. , "Mitigating Routing misbehavior in Mobile Ad Hoc Networks", MOBICOM2000, pp.255-265,2000.