モバイル向け証明書検証システムの開発と評価
モバイルセキュリティ基盤技術の研究開発 I

梅澤 克之†‡ 高橋 礼† 内山 宏樹† 坂崎 尚生† 笈川 光浩†
洲崎 誠一† 平澤 茂一‡

†(株) 日立製作所 システム開発研究所
212-8567 神奈川県川崎市幸区鹿島田 890 日立システムプラザ新川崎
{ume,aytkhs,uchiyama,sakazaki,moikawa,susaki}@sdl.hitachi.co.jp
‡早稲田大学大学院理工学研究科
169-8555 東京都新宿区大久保 3-4-1
{ume,hirasawa}@hirasa.mgmt.waseda.ac.jp

あらまし モバイル環境における異なる携帯電話事業者間の公開鍵証明書の相互認証・相互検証技術，携帯端末のセキュリティ機能を代行するモバイルサービス代行技術，モバイル環境での利用者の属性情報の安全で適切な利用技術等の研究開発を，(株)NTT ドコモ，(株) 日立製作所，日本電気 (株)，(株)KDDI 研究所の 4 社コンソーシアムによって実施している．本稿では公開鍵証明書の検証技術に関して，有効性確認の従来方式および提案方式の性能を表す理論式を導出し，モバイル環境における理論値を比較検討する．

Development and evaluation of certificate verification system in mobile environment

Katsuyuki Umezawa†‡ Aya Takahashi† Hiroki Uchiyama† Hisao Sakazaki†
Mitsuhiro Oikawa† Seiichi Susaki† Shigeichi Hirasawa‡

†Hitachi, Ltd. Systems Development Laboratory
Hitachi System Plaza Shinkawasaki, 890, Kashimada, Saiwai-ku, Kawasaki-shi, Kanagawa, 212-8569, Japan
{ume,aytkhs,uchiyama,sakazaki,moikawa,susaki}@sdl.hitachi.co.jp
‡Graduate School of Science & Engineering, Waseda University
3-4-1, Okubo, Shinjuku-ku, Tokyo 169-8555, Japan
{ume,hirasawa}@hirasa.mgmt.waseda.ac.jp

Abstract We developed the public key certificate verification system in consideration of restrictions peculiar to mobile environment, such as processing speed of a cellular phone terminal, memory capacity, and network transmission speed. We derive the theoretical formula that is showing the performance of a validity check of the public key certificate of the conventional system and a proposal system, and compare and examine the theoretical value in mobile environment in this report.

1 はじめに

モバイル環境における異なる携帯電話事業者間の公開鍵証明書（以下証明書）の相互認証・相互検証技術，携帯端末のセキュリティ機能を代行するモバイルサービス代行技術，モバイル環境での利用者の属性情報の安全で適切な利用技術等の研究開発を，(株) NTT ドコモ，(株) 日立製作所，日本電気（株），(株) KDDI 研究所の4社コンソーシアムによって実施している．本稿では携帯端末等を用いたモバイル情報通信サービスにおける証明書の検証技術について報告する．

一般に，PKI(Public Key Infrastructure) 技術を使って通信相手の正当性を確認するためには，証明書を厳密に検証することが必須である．我々は，既に文献 [1] において携帯端末の処理速度，メモリ容量，ネットワークの通信速度等のモバイル環境特有の制約を考慮したモバイル向け公開鍵証明書検証システムを開発し，システム全体の通信量の視点から解析を行った．本報告では，証明書の有効性確認の従来方式と文献 [1] で提案している方式の，計算量を含めた性能を表す理論式を導出し，モバイル環境における理論値を比較検討する．さらに，開発したシステムの実測値を評価する．以下では，まず，2章で現状の証明書の有効性確認方式について記述する．3章で1回の認証に必要な平均検証時間の理論式を導出し，4章でモバイル環境でのパラメータを当てはめて比較評価する．そして最後に5章でまとめを示す．

2 従来技術

厳密に証明書の検証を行うには,「認証パスの構築」や「認証パスの検証」以外に，認証パス中の証明書が失効されていないことを確認するための「証明書の有効性確認」が必要である．「証明書の有効性確認」の方法には，CRL (Certificate Revocation List) 方式 [4][5] や，OCSP (Online Certificate Status Protocol) 方式 [6]，CVS (Certificate Validation Server) 方式 [7] などがある[1]．以下にその概要を示す．

2.1 CRL 方式

CRL は失効された証明書のシリアル番号の一覧であり，一般的には認証局 (CA) 単位で発行・管理される．ある証明書の有効性を確認したい場合，その証明書を発行した認証局のリポジトリから CRL を取得し，CRL 内にその証明書のシリアル番号が記載されているか否かをチェックすることで判断する．CRL は，通常一定の周期ごとに発行され，証明書の有効期間が満了したものについては，CRL から除外される．CRL 方式には，完全 CRL 方式と，δ-CRL 方式がある[2]．完全 CRL 方式は，CRL の発行時点で，失効されていてかつ有効期間内であるすべての証明書の番号を含める方式である．一方，δ-CRL 方式は，比較的長い時間間隔で，base-CRL と呼ぶ完全 CRL と同じ情報を含む CRL を発行し，base-CRL の発行の間では δ-CRL と呼ぶ base-CRL より短い発行間隔の CRL を発行する方式である．δ-CRL には base-CRL の発行以降に新たに失効されかつ有効期間内である証明書の番号だけが含まれる．

2.2 OCSP 方式

OCSP 方式とは，証明書の有効性を OCSP レスポンダとよばれるサーバにオンラインで問い合わせる方式である．要求メッセージとして有効性を確認したい証明書の情報（証明書の ID 等）を送付すると，その応答として，有効 (good)，失効 (revoked)，不明 (unknown) の3つのいずれかが返信される．

2.3 CVS 方式

CRL 方式や OCSP 方式には，証明書検証者が認証パスの構築や証明書の検証を行わなければならず証明書検証者側の負担が大きいといった問題がある．この負担を軽減するために考えられた方式が CVS 方式である．CVS 方式は，本来の証明書の検証者に代わって，認証パスの構築・検証および認証パス中の全証明書の有効性確認を代行する方式である．検証者が，サーバに検証対象となる証明書と信頼する認証局の証明書を送付すると，検証対象証明書の正当性を確認した結果が返信される．

[1] この他に SCVP 方式 [8] があるが，現在ドラフト版のため今回の評価からは除外する．

[2] この他にも失効情報を複数の CRL に分割して公開する区分 CRL 方式や，間接 CRL 方式，証明書失効ツリー (CRT:Certificate Revocation Tree) 方式などがある．

3　証明書検証の平均総時間

本節では，ある Entity が 1 回の証明書検証を行う際に必要な検証時間（通信時間+計算時間）の平均値（平均検証時間）を評価する．

3.1　モデルの定義

まず証明書の有効性確認のモデルを定義する．図 1 は文献 [3] に示されている (δ-)CRL 方式によるモデルである．図 2 と図 3 は証明書検証局 (VA) と Entity 間で OCSP 方式を用いるモデルであり，図 2 は Entity が証明書の有効性確認を個別に複数の VA に問い合わせるモデル，図 3 は証明書パス中のすべての証明書の有効性確認を 1 回の問い合わせで行うモデルである．図 4 は VA と Entity 間で CVS 方式を用いるモデルである．

3.2　平均検証時間

CRL は，2.1 節で示したように通常一定の周期ごとに発行されるので，その周期内であれば，一度取得してしまえば 2 度目以降は CRL を取得する必要はない．このように，CRL の取得の必要性は証明書検証の事象が発生する確率によるので，CRL の取得時間は平均値で求める必要がある．CRL の取得時間の平均値を C_x，端末およびサーバで証明書の検証に必要な計算時間を M_x，有効性確認要求時間を R_x とすると，平均検証時間 T_x は，次式で表せる．

$$T_x = C_x + M_x + R_x \qquad (1)$$

ただし，x はモデルを表す．次節以降で，モデルごとの C_x, M_x, R_x を求める．

3.3　証明書検証を行う確率

C_x を求める前に，まず証明書検証の事象が発生する確率を求める．一般的にある時間間隔に平均 λ 回発生する事象の発生回数 X の確率分布は，Poisson 分布 $P(X) = \lambda^X \cdot e^{\lambda}/X!$ に従うことが知られている．

文献 [3] より，認証頻度 q[回/day・個]，CA の個数 k[個] とすると，時間間隔 T[day] の間にある検証者がある CA に属する Entity を認証する回数の期待値は qT/k 回になる．よって，ある Entity が，ある CA に属する Entity を認証する回数は，$\lambda = \frac{qT}{k}$ の Poisson 分布に従うと考えられる．よって，Entity において時間間隔

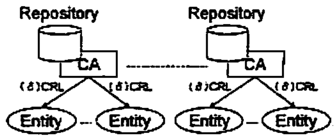

図 1: (δ-)CRL モデル

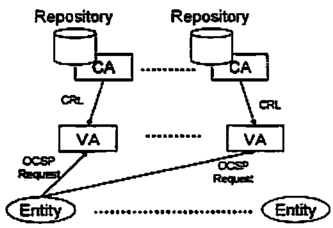

図 2: OCSP モデル 1

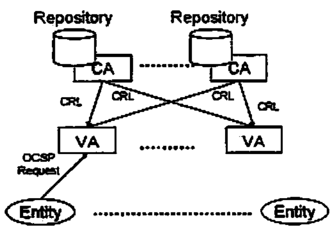

図 3: OCSP モデル 2

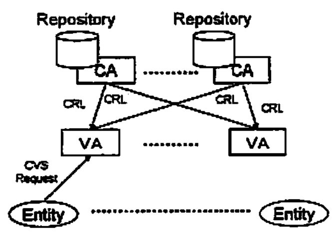

図 4: CVS モデル

T の間に 1 回以上認証が行われる確率 $p_{eX\geq1}^{T}$ は下記で表せる．

$$p_{eX\geq1}^{T} = 1 - e^{-\frac{qT}{k}} \qquad (2)$$

つぎに，VA において認証（Entity からの有効性確認要求に基づく処理）が発生する確率を求める．複数のグループの事象の発生回数 X の確率分布が Poisson 分布に従っているときにそれらを合計した事象の発生回数 X' の確率分布も Poisson 分布従うことが知られている．VA の数を N_v とすると，VA における認証頻度は $q' = \frac{q \cdot N}{N_v}$ となる．OCSP モデル 1 では k/N_v 個，OCSP モデル 2 と CVS モデルでは k 個の CA に属する Entity を認証することになるので，時間間隔 T の間に VA がある CA に属する Entity

を認証する回数の期待値は前者で $qT/k \cdot N$ 回，後者で $qT/k \cdot \frac{N}{N_v}$ 回となる．よって，VA において時間間隔 T の間に 1 回以上認証が行われる確率は，OCSP モデル 1 のそれを $p'^{T}_{vX\geq 1}$，OCSP モデル 2 と CVS モデルのそれを $p_v{}^{T}_{X\geq 1}$ とすると下記で表せる．

$$p'^{T}_{vX\geq 1} = 1 - e^{-\frac{qT}{k}\cdot N} \quad (3)$$

$$p_v{}^{T}_{X\geq 1} = 1 - e^{-\frac{qT}{k}\cdot\frac{N}{N_v}} \quad (4)$$

3.4 CRL の平均取得時間の導出

CRL モデルおよび δ-CRL モデルでは CA-Entity 間で CRL の取得が行われる．また，その他のモデルでは CA-VA 間で CRL の取得が行われる[3]．CRL の発行間隔 T_C 内に CRL の取得が行われる確率は，ある CA に属する Entity を 1 回以上認証する確率である．よって CRL モデルにおける 1 回の CRL 取得に要する平均時間 C_{CRL} は，下記のように表せる．

$$C_{CRL} = \frac{k \cdot p_e{}^{T_C}_{X\geq 1} \cdot l_{CRL}}{T_C \cdot q \cdot s} \quad (5)$$

ただし，l_{CRL} は，1 つの CA が 1 回に発行する CRL のサイズであり，文献 [3] より，下記である．

$$l_{CRL} = \frac{N'pL}{k} \cdot l_{sn} + l_{sig}[bit] \quad (6)$$

また，q は 1 Entity が 1 日に認証される平均回数，s は CA-Entity 間（モバイル網）の通信速度 [bit/sec] である．

δ-CRL モデルの場合は，base-CRL と δ-CRL の両方を取得するための時間が必要である．前者を $C_{baseCRL}$，後者を C_{delta} とすると，次式が成立する．

$$C_{deltaCRL} = C_{baseCRL} + C_{delta} \quad (7)$$

base-CRL の取得時間 $C_{baseCRL}$ は，式 5 における時間間隔を base-CRL の発行間隔 T_B とした次式で表せる．

$$C_{baseCRL} = \frac{k \cdot p_e{}^{T_B}_{X\geq 1} \cdot l_{baseCRL}}{T_B \cdot q \cdot s} \quad (8)$$

[3]文献 [3] に従い，CA 証明書の失効頻度は十分小さいのでその検証に必要な CRL 取得のための通信量は無視できると仮定する．

ただし，$l_{baseCRL}$ は，式 1 の l_{CRL} と同様となる．δ-CRL の取得時間 C_{delta} は，n 番目に発行された δ-CRL のサイズ $l_{delta}(n)$ を時間平均した次式で表せる．

$$C_{delta} = \frac{k \cdot p_e{}^{T_C}_{X\geq 1} \cdot \sum_{n=1}^{\frac{T_B}{T_C}-1} l_{delta}(n)}{T_B \cdot q \cdot s} \quad (9)$$

$l_{delta}(n)$ は，文献 [3] より，

$$l_{delta}(n) = \frac{N'pL}{k}\left\{1 - \left(1 - \frac{T_C}{L}\right)^n\right\} \cdot l_{sn} + l_{sig}$$

である．

その他のモデルの場合は，すべて CA-VA 間の CRL の取得であり，1 回の CRL 取得に要する平均時間 C_x(x はモデル) は，下記のように表せる．

$$C_{OCSP1} = \frac{\frac{k}{N_v} \cdot p'^{T_C}_{vX\geq 1} \cdot l_{CRL}}{T_C \cdot q' \cdot \beta s} \quad (10)$$

$$C_{CVS} = C_{OCSP2} = \frac{k \cdot p_v{}^{T_C}_{X\geq 1} \cdot l_{CRL}}{T_C \cdot q' \cdot \beta s} \quad (11)$$

ただし，$\beta(\geq 1)$ は，VA-Entity 間（モバイル網）の通信速度に対する CA-VA 間（バックエンド）の通信速度の倍率である．

3.5 計算時間の導出

相手から証明書を受け取った Entity は一般的に下記の処理を行う．

(1) 認証パスの構築
(2) 証明書の署名の検証
(3) 証明書有効性確認要求の生成
(4) CRL を用いた失効確認
(5) 証明書有効性確認結果の署名検証

これらの計算処理のうち (1) および (2) の処理はどのモデルにおいても $r-1$ 回行う必要があるが，CVS モデルでは VA が行うのに対してその他の方式は Entity が行う点が異なっている．また (3) および (5) の処理は，オンラインで有効性確認を行わない CRL モデルおよび δ-CRL モデルでは行われず，OCSP モデル 1 では $r-1$ 回，その他のモデルでは 1 回行う必要がある．さらに (4) の処理は，CRL モデルおよび δ-CRL モデルでは Entity が行うのに対して，その他のモデルでは VA が行う．よって 1 回の証明書検証

に必要な計算時間 M_x(x はモデル) は下記のように表せる.

$$M_{CRL} = M_{deltaCRL} = (r-1)(M + M'') \quad (12)$$

$$M_{OCSP1} = (r-1)(M + \alpha M'' + M') \quad (13)$$

$$M_{OCSP2} = (r-1)(M + \alpha M'') + M' \quad (14)$$

$$M_{CVS} = (r-1)(\alpha M + \alpha M'') + M' \quad (15)$$

ただし記号の意味は下記である.

- M : Entity 端末における認証パスの構築および証明書の署名検証時間 [sec]
- M' : Entity 端末における証明書検証要求の生成および証明書検証結果の署名検証時間 [sec]
- M'' : Entity 端末における CRL を用いた失効確認時間 [sec]
- r : 証明書パスの長さ (CA 階層+1)[階層]
- α : Entity 端末の計算速度に対する VA サーバの計算速度の比 ($0 < \alpha < 1$)

ここで, 計算時間を $r-1$ 倍しているのは, ルート CA 証明書は何らかの方法で既検証済みでありトラストアンカーとして Entity は信頼済みという仮定を置いているためである.

3.6 有効性確認要求時間の導出

有効性確認時間は, それぞれの方式の有効性確認要求のデータサイズと通信時間によって導出できる. オンラインで有効性確認を行わない CRL モデルおよび δ-CRL モデルでは有効性確認時間は 0 となる[4]. 確認要求データのビット数は OCSP モデル 1 では r 個のリクエストを生成しなければならないので $r(D_{sn} + D_{sig})$, OCSP モデル 2, CVS モデルでは, 1 つのリクエストに複数の項目を入れることができるので, それぞれ $rD_{sn} + D_{sig}$,$rD'_{sn} + D'_{sig}$ となる. 1 回の証明書検証に必要な有効性確認要求時間 R_x(x はモデル) は下記のように表せる[5].

$$R_{CRL} = R_{deltaCRL} = 0 \quad (16)$$

$$R_{OCSP1} = \frac{(r-1)(D_{sn} + D_{sig})}{s} \quad (17)$$

$$R_{OCSP2} = \frac{(r-1)D_{sn} + D_{sig}}{s} \quad (18)$$

$$R_{CVS} = \frac{rD'_{sn} + D'_{sig}}{s} \quad (19)$$

ただし記号の意味は下記である.

- D_{sn} : OCSP 要求の項目 1 つあたり (証明書 ID 等) のビット数 [bit]
- D_{sig} : OCSP 要求の項目数によらず一定な要素のビット数 [bit]
- D'_{sn} : CVS 要求の項目 1 つあたり (証明書等) のビット数 [bit]
- D'_{sig} : CVS 要求の項目数によらず一定な要素のビット数 [bit]
- s : VA-Entity 間 (モバイル網) の通信速度 [bit/sec]

4 モバイル環境適用時の比較

携帯端末がサーバを認証するときの通信量を評価するにあたり, 表 1 のようにパラメータを設定した. 本パラメータは文献 [1] の提案方式のパラメータと同一の値である.

表 1: 評価用パラメータ

Entity(認証者) の数 N[個]	87,000,000
被認証者の数 N'[個]	3,000,000
失効発生頻度 p[回/day]	0.1/365
証明書の有効期間 L[day]	365
完全 CRL,δ-CRL の発行間隔 T_C[day]	1
CRL の項目 1 つあたりサイズ l_{sn}[bit]	72
CRL の項目によらず一定な要素のサイズ l_{sig}[bit]	728
CA の数 k[個]	500
VA の数 N_v[個]	10
OCSP 要求の項目 1 つあたり (証明書 ID 等) のビット数 D_{sn}[bit]	632
OCSP 要求の項目数によらず一定な要素のビット数 D_{sig}[bit]	72
CVS 要求の項目 1 つあたり (証明書等) のビット数 D'_{sn}[bit]	6,464
CVS 要求の項目数によらず一定な要素のビット数 D'_{sig}[bit]	64

モバイル環境を想定した表 1 のパラメータを適用した平均検証時間 T_x(x はモデル) は下記のようになる.

$$T_{CRL} = \frac{42636.13}{s} + 4M \quad (20)$$

$$T_{deltaCRL} = \frac{23470.36}{s} + 4M \quad (21)$$

$$T_{OCSP1} = \frac{0.0084}{\beta s} + \frac{1408}{s} + (2\alpha + 4)M \quad (22)$$

$$T_{OCSP2} = \frac{0.0842}{\beta s} + \frac{1336}{s} + (2\alpha + 3)M \quad (23)$$

$$T_{CVS} = \frac{0.0842}{\beta s} + \frac{19456}{s} + (4\alpha + 1)M \quad (24)$$

[4] 厳密には自身で所持している CRL に検証対象証明書の ID が含まれているか否かの判定時間は要する.

[5] 厳密には結果の返答時間がかかるが, そのサイズは小なので省略する.

なお，$M' = M$，$M'' = M$，$q = 30$ として計算した．また，図5に $\alpha = 0.1$（サーバの処理速度が端末の処理速度の10倍），$\beta = 100$（CA-VA 間の通信速度がモバイル網通信速度の100倍）のときの平均検証時間を示す．

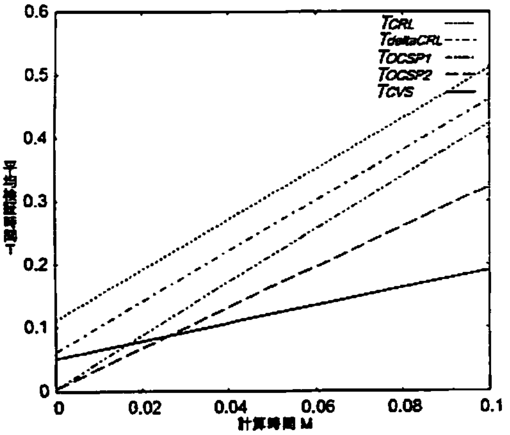

図 5: 通信速度 $s = 384k$[bit/sec] のときの各方式の平均検証時間

図5より，端末の計算時間によらずCRL方式および δ-CRL方式よりCVS方式のほうが検証時間が短いことが確認できる．また，$M = 0.02622$ のときに T_{CRL} と T_{OCSP2} が交わっているので，端末における計算時間が26.22msより多くの時間がかかってしまう場合にはCVS方式のほうが平均検証時間が短くなることが確認できる．

4.1 実測値（計算時間＋通信時間）の評価

CDMA 1x WIN 規格の携帯電話機（BREW 端末）およびCPUにIntel(R) Pentium(R) 4プロセッサ3.4GHz，メモリ2GB，OSにWindows(R) XP operating system を搭載したコンピュータで実装したモバイル向け証明書検証サーバの性能評価を行った．その結果，署名生成時間の100回の平均値は，携帯電話機で10.03ms，サーバで0.16msであった．これより $M = 0.01003$，$\alpha = 0.16/10.03$ を $T_{OCSP2} - T_{CVS} > 0$ に代入し，s について解くと $s > 917933$ となる．つまりモバイル網における通信速度がおよそ900kbps以上の場合にはOCSP方式に比較してCVS方式は有効だということがわかる．

5 まとめと今後の課題

公開鍵証明書の有効性確認の従来方式および提案方式の性能を表す理論式を導出し，モバイル環境における理論値を比較し，モバイル環境においてCVS方式が適している範囲を示した．

今後は，モバイル特有の他の制約を考慮した方式においても性能劣化が小となることをします予定である [2]．

謝辞　本研究は，独立行政法人情報通信研究機構 (NICT) の委託研究「モバイルセキュリティ基盤技術の研究開発」の一環として行なわれた．

商標等に関する表示

- Windows は米国 Microsoft Corporation の米国およびその他の国における登録商標です．
- Intel, Pentium は，米国およびその他の国における，Intel Corporation またはその子会社の商標または登録商標です．
- BREW および BREW に関連する商標は，Qualcomm 社の商標または登録商標です．

参考文献

[1] 梅澤，高橋，内山，坂崎，笠川，洲崎，平澤，"モバイル向け証明書検証サーバの開発"，電子情報通信学会技術報告（IT），2005年9月（予定）．

[2] 梅澤，笠川，洲崎，平澤，"モバイル向け証明書検証方式の評価"，第28回情報理論とその応用シンポジウム，予稿集，2005年11月（予定）．

[3] 田中，飯野，"PKIの証明書失効に必要な通信量の確率論的評価"，情報処理学会論文誌，Vol.45 No.12, (2004).

[4] *ITU-T Recommendation X.509 (2000)—ISO/IEC 9594-8:2001: Information Technology - Open Systems Interconnection - The Directory: Public-key and Attribute Certificate Framework*

[5] R. Housley, T. Polk, W. Ford, and D. Solo: *RFC 3280 - Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*, IETF, April 2002.

[6] M. Myers, R. Ankney, A. Malpani, S. Galperin and C. Adams: *RFC 2560 - X.509 Internet Public Key Infrastructure - Online Certificate Status Protocol - OCSP*, IRTF, June 1999.

[7] 政府認証基盤相互運用性仕様書, H15/12/17 改定, 共通システム専門部会了承.

[8] T.Fressman, R.Housley, A.Malpani, D.Cooper and T.Polk: *Simple Certificate Validation Protocol (SCVP)*, IETF, July. 2005.