SCIS 2006 The 2006 Symposium on Cryptography and Information Security
Hiroshima, Japan, Jan. 17-20, 2006
The Institute of Electronics, Information and Communication Engineers

企業ネットワークにおける管理負荷の少ない認証システムASEの提案

A proposal of Authentication System for an Enterprise network ASE having less management loads

坂野 文男* 保母 雅敏* 渡邊 晃*
Fumio Banno Masatoshi Hobo Akira Watanabe

あらまし インターネットの普及に伴い，電子商取引や電子申請等の電子化が進み，ユーザの確実な認証が重要な課題となっている．この課題を解決するために公開鍵認証基盤 PKI が注目されている．企業ネットワークにおいてもその利点に着目し，PKI を導入する傾向がある．本稿では企業が PKI を導入するうえでどのような課題があるかを考察し，その課題を解決するための認証基盤の一方式について提案する．PKI では，root CA の公開鍵はユーザがあらかじめ信頼できる方法で取得しておき，厳重に管理する必要がある．しかし,root CA の公開鍵は偽造が可能であり，このことを利用したセキュリティ上の脅威が懸念される．また，PKI では証明書の有効性を確認するために証明書失効リスト CRL を発行する必要がある．CRL は管理が面倒なうえ，その情報が必ずしも最新とは限らないという課題が指摘されている．本稿では，PKI を参考に企業内ネットワークで利用できる管理負荷の少ない認証基盤 ASE を検討した．ASE では，信頼関係を環状にし，公開鍵証明書は発行者が保持して自ら管理を行い，信頼関係の検証をオンデマンドで行う．ASE は，PKI にみられるようなセキュリティ上の脅威が無く，管理が容易でリアルタイム性の高い認証基盤を提供できる．

キーワード セキュリティ，認証基盤，証明書，PKI，CRL，ASE

1 はじめに

インターネットの普及に伴い，電子商取引や電子申請等の電子化が急激に進んでいる．しかし，ネットワーク上には盗聴，不正アクセス，なりすまし，改ざん，否認といったネットワーク固有の脅威が存在する．これらの脅威を回避するために公開鍵暗号を用いたセキュリティ基盤 PKI（Public Key Infrastructure）が注目されている．PKI は公開鍵暗号方式の暗号化を利用してユーザに秘匿を提供し，署名を利用してユーザに認証，完全性，否認拒否の機能を提供する．企業ネットワークにおいてもその利点に着目し，PKI による認証基盤を導入する傾向がある．

PKI では, 各ユーザの公開鍵を認証局（CA：Certificate Authority）が署名し，CA の公開鍵は更に上位の CA が署名する．しかし，最上位の CA（root CA）の公開鍵証明書を発行する機関がなく，通常は root CA 自身が自己署名する．そのため，root CA の公開鍵証明書が偽造されてもそれを確認する術がない．また，PKI では発行した公開鍵証明書を被発行者に手渡すのが原則であるため，証明書の有効性を確認するために証明書が失効していないかどうかを発行者に確認する必要がある．失効情報の確認方法には CRL（Certificate Revocation List）[1]モデルと OCSP（Online Certificate Status Protocol）[2]モデルがあるが両者ともその情報が必ずしも最新とは限らないという課題が指摘されている．

そこで本稿では，PKI を参考に企業内ネットワークで利用できる管理負荷の少ない新しい認証基盤 ASE(Authentication System for an Enterprise network)を提案する．ASE の特徴は，信頼関係を環状にし, 公開鍵証明書は発行者が保持して自ら管理を行い，信頼関係の検証をオンデマンドで行う．これにより，管理が容易でリアルタイム性の高い認証基盤を提供できる．

2 PKI

2.1 PKI の信頼関係と企業の業務形態

公開鍵証明書は現実社会における印鑑証明書や身分証明書に相当し，これを基にして，セキュリティの基盤を構築する．PKI では CA が公開鍵証明書を発行することにより信頼関係を構築することができる．信頼関係の

* 名城大学大学院理工学研究科, 〒468-8502 愛知県名古屋市天白区塩釜口 1-501，Graduate School of Science and Technology, Meijo University, 1-501, Shiogamaguchi, Tenpaku-ku, Nagoya-shi, Aichi 468-8502, Japan.

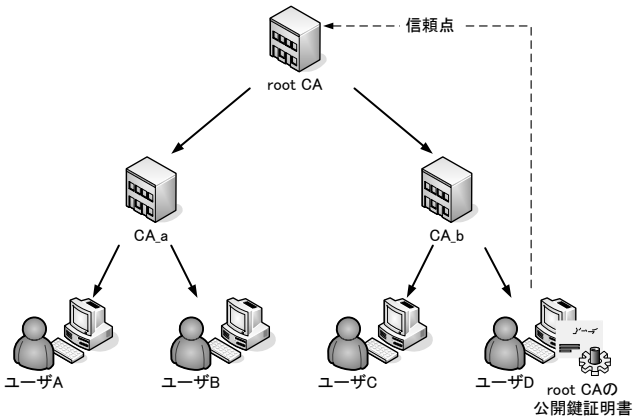


図１　階層型モデルによる信頼関係の構築

構築とはいくつかの CA が連携することである．図１に複数の CA を階層型（ツリー構造）に構成する階層型モデルの例を示す．ユーザの公開鍵証明書は CA により発行され，CA の公開鍵証明書は更に上位の CA により発行される．ユーザは root CA を信頼点とし，root CA の公開鍵証明書をあらかじめ安全な方法で取得し所持しておく．

階層型モデルは企業の業務形態と対応付けができる．例えば root CA を社長，下位の CA を部長，ユーザを部長の下で働く社員というように対応させることができる．社長は各部長について把握しており，部長は部下について把握しているはずである．そこで企業に認証基盤を導入する際，信頼関係の構築を階層型モデルで行えば社員の人事異動等による公開鍵証明書の変更にも対応しやすいと考えられる．

2.2　公開鍵証明書の有効性検証

PKI の場合，公開鍵証明書の有効性検証は認証パスの構築と認証パスの検証の手順で行う．認証パスの構築では，認証するユーザの公開鍵証明書から検証者の信頼点となる root CA までの公開鍵証明書を収集し，対象となる公開鍵証明書が信頼点と関連づけられていることを確認する．認証パスの検証では，収集したすべての公開鍵証明書において，署名内容が正しいか，有効期間が切れていないか，失効していないかなどを検証する．認証パスの構築と認証パスの検証が問題なく終了することにより公開鍵証明書の有効性検証が終了する．

図２に，図１におけるユーザ D がユーザ A の公開鍵証明書の有効性検証を行う場合の処理の例を示す．まず検証するユーザ A の公開鍵証明書を収集する．ユーザ A の公開鍵証明書をチェックすることにより発行者は CA_a とわかり，次に CA_a の公開鍵証明書を収集する．CA_a の公開鍵証明書をチェックすることより発行者は root CA とわかるが，root CA はユーザ D の信頼点のためここで認証パスの構築を終了し，認証パスの検証へ移る．

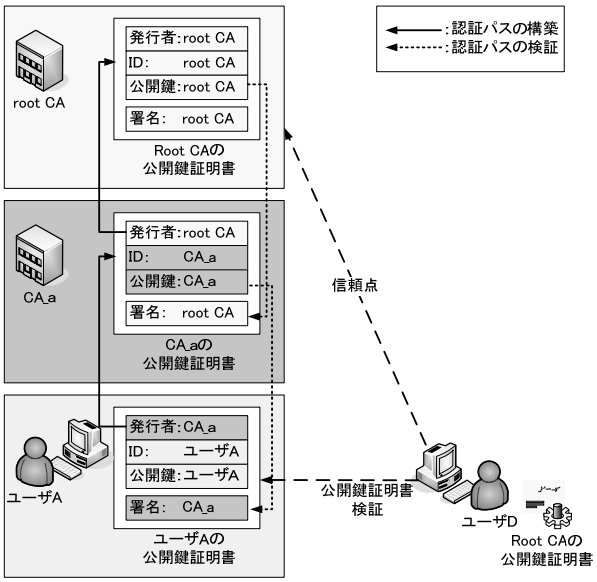


図２　公開鍵証明書の有効性検証方法

認証パスの検証は，認証パスの構築時に公開鍵証明書を手に入れた順序とは逆に行う．すなわち root CA，CA_a，ユーザ A の順番に公開鍵証明書の検証を行い，すべての公開鍵証明書の有効性検証が成功した場合，ユーザ A の公開鍵証明書が信頼できるものと判断する．このとき，公開鍵証明書が失効していないことを確認するために，下記のような失効情報を確認する必要がある．

2.3　失効情報の確認

認証パスの検証を行うためには公開鍵証明書の発行者が失効情報を管理しておく必要がある．失効情報の管理方法には CRL モデルと OCSP モデルがある．

CRL モデルの概要を図３に示す．まず公開鍵証明書を発行した CA が CRL（Certificate Revocation List；証明書失効リスト）を定期的な周期で発行する．各ユーザは公開鍵証明書の検証をする前にあらかじめ CA から CRL を収集しておく必要がある．各ユーザは公開鍵証明書の検証時に事前に収集した CRL に検証対象の公開鍵証明書が記載されていないことを確認することにより公開鍵証明書の有効性を確認する．

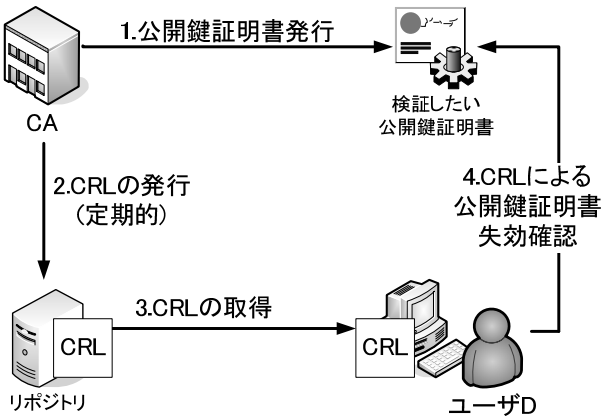


図３　CRL モデルの概要

OCSPモデルは公開鍵証明書の検証時に失効状態を集中管理する OCSP レスポンダに対しリアルタイムで有効性を確認するプロトコルである．検証者は検証対象の公開鍵証明書を取得した後，検証対象の公開鍵証明書に記載されている OCSP レスポンダに対し公開鍵証明書の状態を問い合わせる．OCSP レスポンダは当該公開鍵証明書が失効していないか，失効情報との照合を行い，結果をユーザへ返答する．ユーザは返答の内容により公開鍵証明書の状態を確認する．

2.4 PKI の課題

(1)Root CA の公開鍵証明書の偽造

PKI では，最上位の root CA の公開鍵証明書を発行する機関がなく，root CA 自身が公開鍵証明書を発行（自己署名）しているが，この公開鍵証明書の発行者が正当であることを検証する方法がない．このことは，root CA の公開鍵は偽造される可能性があることを示している．Windows では root CA の公開鍵証明書がレジストリに保存されているが，このレジストリを直接操作することにより書き換えができる．Root CA の公開鍵証明書をインストールしたり削除を行う場合，一般に図 4 の様に確認画面が表示されるが，レジストリから直接 root CA の公開鍵証明書を操作すると確認画面が表示されない．悪意あるプログラムなどによりレジストリを操作されてもユーザはそのことに気づかない．

Root CA の公開鍵証明書が偽造されると下記のように悪用される可能がある. まず悪意ある店舗が SSL 通信の認証と暗号化を行うインターネットショッピングサイトを立ち上げる．Root CA の公開鍵証明書を入れ替えられたユーザは，SSL 通信時に公開鍵証明書の検証が問題なく終了するため，認証が成功したものと解釈する．そこで，ユーザは店舗を信用してクレジット番号等個人情報を入力して購入手続きをしてしまう．

このように root CA の公開鍵証明書が偽造された場合，ユーザは偽のサーバを信用させられる可能性があり，さらにこのような状態に陥っても何ら異常を検出することができない．

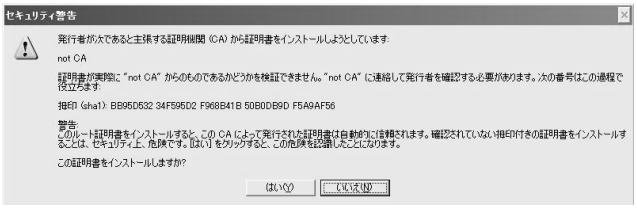


図 4　root CA の公開鍵証明書インストール確認画面

(2)失効情報の管理

PKI では発行した公開鍵証明書の有効性を確認するために公開鍵証明書の失効情報を管理する必要がある．公開鍵証明書は発行者の手を離れ被発行者が所持しているため，特定のユーザの公開鍵証明書を失効させたくても対象のユーザが公開鍵証明書の削除を行わず使用し続ける可能性がある．従って，公開鍵証明書が失効していないかどうかを発行者に確認する必要がある．このために用いられるのが失効情報である．失効情報は原則的に増加し続けるため管理が大変であり，失効情報のデータが大きくなると, 有効性の確認時に多くの時間を要する．失効情報の確認に CRL モデルを利用する場合は，検証者が公開鍵証明書の検証をする前に CA から CRL をあらかじめ収集しておく必要がある. CRL は決められた周期で発行されるため，公開鍵証明書が失効された場合でも，次回の CRL が発行されるまでは失効情報が利用者に伝わらず, 最新の情報が得られない場合がある. OCSP モデルを利用する場合においても，OCSP レスポンダの失効情報の更新は CRL を利用することが多く，必ずしも最新の情報であるとは限らない．

3 提案方式 ASE

3.1 概要

本稿では，企業内ネットワークを想定し，上記のような PKI の課題を解決するために, 以下のような方式を提案する. まず PKI では信頼関係を階層化にするのに対し, ASE では信頼関係を環状にする． 次に PKI では公開鍵証明書が発行者の手を離れ，被発行者へ渡されるのに対し，ASE では発行者自らが保持/管理する．また，PKI では公開鍵証明書の有効性確認に失効情報を事前に入手するか問い合わせる必要があるのに対し, ASE では公開鍵証明書をオンデマンドで収集し，その時点で失効の有無を確認する．

これにより，root CA の公開鍵証明書の偽造を検出でき，管理が容易でリアルタイム性の高い認証基盤を提供することが可能となる．

3.2 信頼関係の構築

ASE の信頼関係を図 5 に示す. ルートサーバは部門ごとに設置された認証サーバに公開鍵証明書を発行する．認証サーバは各部門の社員に公開鍵証明書を発行する．各社員はルートサーバに公開鍵証明書を発行する． このように信頼関係を環状にさせることにより，公開鍵証明書の検証時に自分を最上位に位置づけることができる．公開鍵証明書の収集時に，自分の持つ秘密鍵を信頼点とすることにより，全ての公開鍵証明書が正しいことを検証できる．環状の信頼関係を一度築けば全ての公開鍵証明書は偽造を検出できるようになり，安全性が保証される．

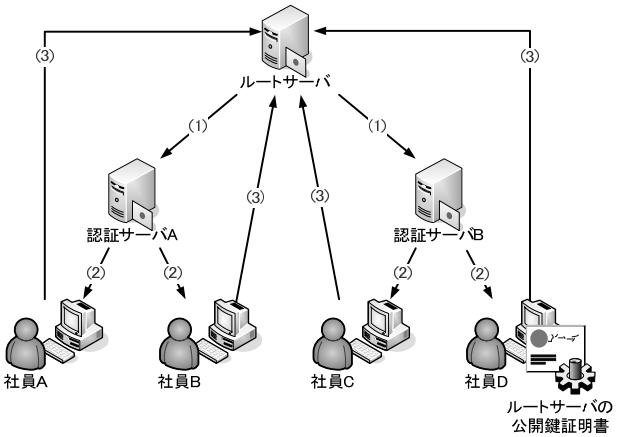


図 5　ASE の信頼関係

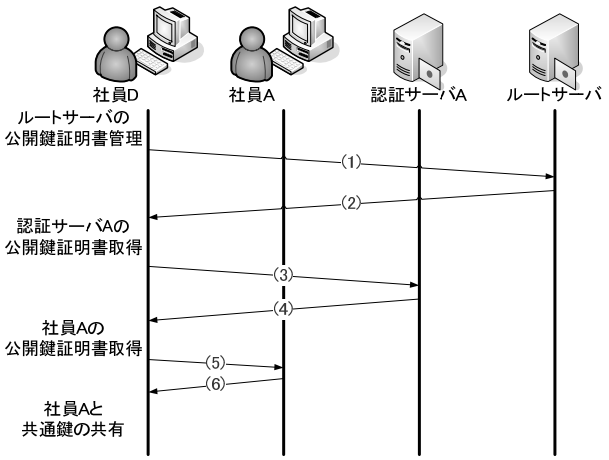


図 7　ASE の公開鍵証明書の管理方法

3.3　公開鍵証明書の管理

ASE の公開鍵証明書の管理方法を図 6 に示す．図 6 は，図 5 において社員 D が社員 A を認証する場合に必要となる公開鍵証明書のみを表示している．PKI は社員 A の公開鍵証明書を，被発行者の社員 A 本人が保持/管理しているのに対し，ASE では社員 A の公開鍵証明書を発行者の認証サーバ A が保持/管理している．同じように ASE では認証サーバ A の公開鍵証明書は発行者のルートサーバが保持/管理していて，ルートサーバの公開鍵証明書は発行者の社員 D が保持/管理している．このように発行した公開鍵証明書を被発行者に渡すのでなく発行者自身が保持/管理する．認証時にはオンデマンドで必要となる公開鍵証明書をすべて収集するため，管理方法をこのように改めても特に問題は発生しない．この管理方法により公開鍵証明書が失効した場合は，管理している公開鍵証明書を単に削除するだけで済む．

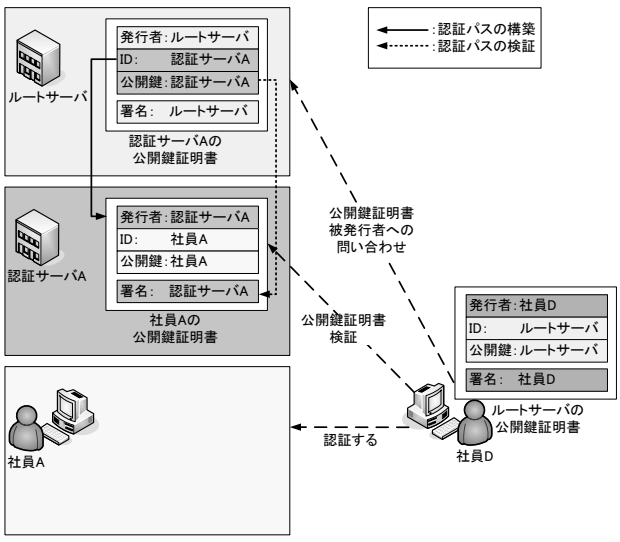


図 6　ASE の公開鍵証明書の管理方法

3.4　公開鍵証明書の有効性検証

ASE における公開鍵証明書の有効性検証方法を図 7 に示す．公開鍵証明書の有効性検証に必要となる情報は，検証者が検証時にオンデマンドで収集する．具体的な検証方法は以下のようになる．

(1)　社員 D はルートサーバへ社員 A の身元を問い合わせる
(2)　ルートサーバは社員 A が所属している認証サーバ A の公開鍵証明書を返答する
(3)　社員 D は認証サーバ A へ社員 A の公開鍵証明書を要求する
(4)　認証サーバ A は社員 A の公開鍵証明書返答する

上記により認証パスの構築は終了し，認証パスの検証へ移る．収集した公開鍵証明書の正当性を検証し，検証が成功した場合，社員 A の公開鍵証明書は信頼することができる．以上で証明書の有効性検証は終了し，失効情報の確認作業は不要である．社員 A の認証のためには更に下記手順を実行する．

(5)　社員 D は共通鍵を作成した後社員 A の公開鍵で共通鍵を暗号化し，社員 A へ作成した暗号文を送る
(6)　社員 A は社員 A 本人の秘密鍵を利用し，暗号化された共通鍵を復号し，共通鍵を利用して社員 D へ共通鍵を取得したことを返答する

以後の通信には上記共通鍵を用いて暗号化通信が可能である．

3.5　ASE の課題

ASE はルートサーバに対して各社員が署名を行うという性質上，公開鍵証明書の検証時にすべてのユーザがルートサーバへ問い合わせを行う必要があるため，ルートサーバへの負荷が多くなる．この課題を解決するにはルートサーバを分散化するなどの方法を検討する必要があると考えられる．

次に，PKI では失効情報より失効した日時がわかる．署名データが刻印される時に署名が存在していたことを

保証する Time-Stamp Protocol[3]を利用することにより，公開鍵証明書失効前に行われた署名を信頼することができる．しかし，ASE は失効情報を管理しないため失効日時を知りたい場合は別の手段を検討する必要がある．

4　評価

PKI（CRL），PKI（OCSP）と ASE の比較を表 1 に示す．

リアルタイム性について，PKI（CRL）は失効情報が一定周期で発行され，公開鍵証明書検証者は前もって CRL を手に入れる必要があるため，ユーザが最新の有効性を確認できない場合がある．PKI（OCSP）は公開鍵証明書の有効性を検証時に問い合わせるため PKI（CRL）よりリアルタイム性に優れているが，失効情報に CRL を利用している場合，ユーザが最新の有効性を確認できない場合がある．ASE ではオンデマンドで認証パスを構築するためリアルタイム性に優れ，ユーザが最新の有効性を確認できる．

管理コストについて，PKI（CRL）と PKI（OCSP）は失効情報を確実に管理する必要があり管理コストが高い．ASE は公開鍵証明書を発行者が管理し，失効時は対象となる公開鍵証明書を削除するだけでよい．さらにオンデマンドで公開鍵証明書の有効性を検証するため失効情報の管理を行う必要がない．

最上位の公開鍵証明書の検証について，PKI（CRL）と PKI（OCSP）は自己署名のため発行者が正当であることを検証できず，偽造されても検出ができない．ASE は検証者がルートサーバの公開鍵証明書を自ら検証できるため偽造が検出できる．

ルート機関の負荷について，ASE はルートサーバを各社員が署名を行うという性質上，ルートサーバへの問い合わせが多くなり負荷がかかる．

失効日時の確認について，PKI（CRL）と PKI（OCSP）は失効情報より失効した日時を知ることができるが，AES はそれがわからない．

表 1　PKI と ASE の比較

	PKI(CRL)	PKI(OCSP)	ASE
リアルタイム性	△	△+	○
管理コスト	△	△	○
最上位の公開鍵証明書	検証不可能 ×	検証不可能 ×	検証可能 ○
ルート機関の負荷	○	○	△
失効日時の確認	○	○	△

以上の評価から，ASE はリアルタイム性に優れ，管理コストが低い．また最上位の公開鍵証明書が検証可能なため，ルート機関の公開鍵証明書の偽造を検出できる．企業ネットワークにおける認証基盤としては，十分有効な手段と考えられる．

5　むすび

企業ネットワークにおいて認証基盤を導入するために，信頼関係を環状にし，公開鍵証明書は発行者が保持して自ら管理を行い，信頼関係をオンデマンドで検証を行う認証システム ASE を提案した．評価結果より ASE は企業ネットワークには有効な方式であると考えられる．今後は，ASE の課題を解決する方法を検討するとともに，実装および検証を行っていく予定である．

参考文献

[1] R. Housley, W. Polk, W. Ford, D. Solo, “Internet X.509 Public Key Infrastructure Certificate and CRL Profile”, RFC 3280, April 2002.

[2] M. Myers, R. Ankney, A. Malpani, S. Galperin, C. Adams, “X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP”, RFC 2560, June 1999.

[3] C. Adams, P. Cain, D. Pinkas, R. Zuccherato, “Internet X.509 Public Key Infrastructure Time-Stamp Protocol (TSP)”, RFC 3161, August 2001

[4] C. Adams, S. Farrell, “Internet X.509 Public Key Infrastructure Certificate Management Protocols” , RFC 2510, March 1999.

[5] M. Myers, C. Adams, D. Solo, D. Kemp, “Internet X.509 Certificate Request Message Format” , RFC 2511, March 1999.

[6] J. Ross, D. Pinkas, N. Pope, “Electronic Signature Policies” , RFC 3125, September 2001.

[7] W. Polk, R. Housley, L. Bassham, “Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile” , RFC 3279, April 2002.

[8] S. Chokhani, W. Ford, R. Sabett, C. Merrill, S. Wu, “Internet X.509 Public Key Infrastructure Certificate Policy and Certification Practices Framework” , RFC 3647, November 2003.

[9] 菊池 浩明，中西 祥八郎，“オンライン証明書検証プロトコルのスケーラビリティ”，情報処理学会論文誌，2002 年 8 月

[10] 田中 直樹，飯野 陽一郎，“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，2004 年 12 月

企業ネットワークにおける管理負荷の少ない認証システムASEの提案

名城大学大学院理工学研究科
坂野文男　保母雅敏　渡邊晃

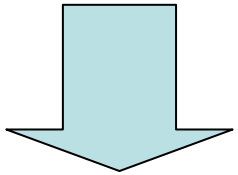
研究背景

- 近年のインターネット普及に伴い、電子商取引や、電子申請等の電子化が進んでいる
- ネットワーク上には「盗聴」、「なりすまし」、「改ざん」等の脅威がある

PKIの重要性が高まっている

企業への導入

- 企業内ネットワークにおいてもPKIによる認証基盤を導入する傾向がある
- PKIには課題がある

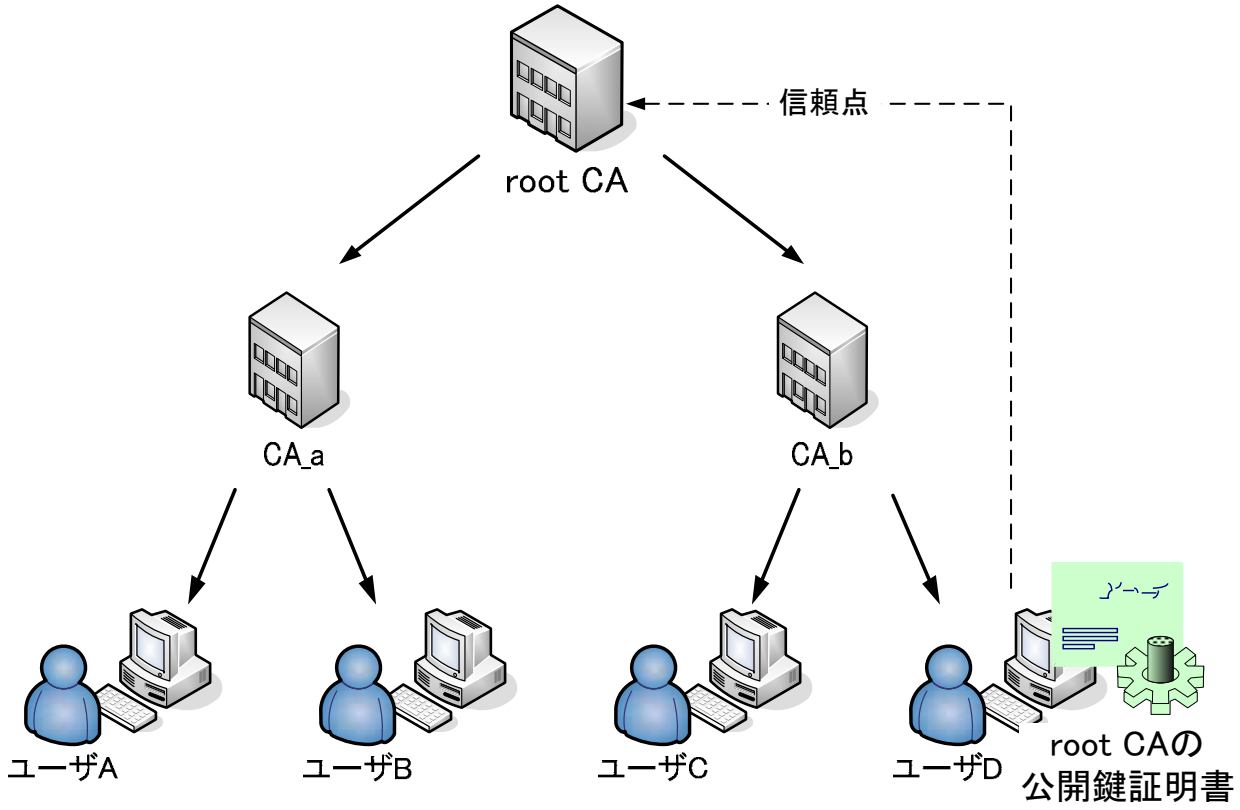


- PKIを参考に企業内ネットワークで利用できる管理負荷の少ない新しい認証基盤ASE(Authentication System for an Enterprise network)を提案する

信頼関係の構築

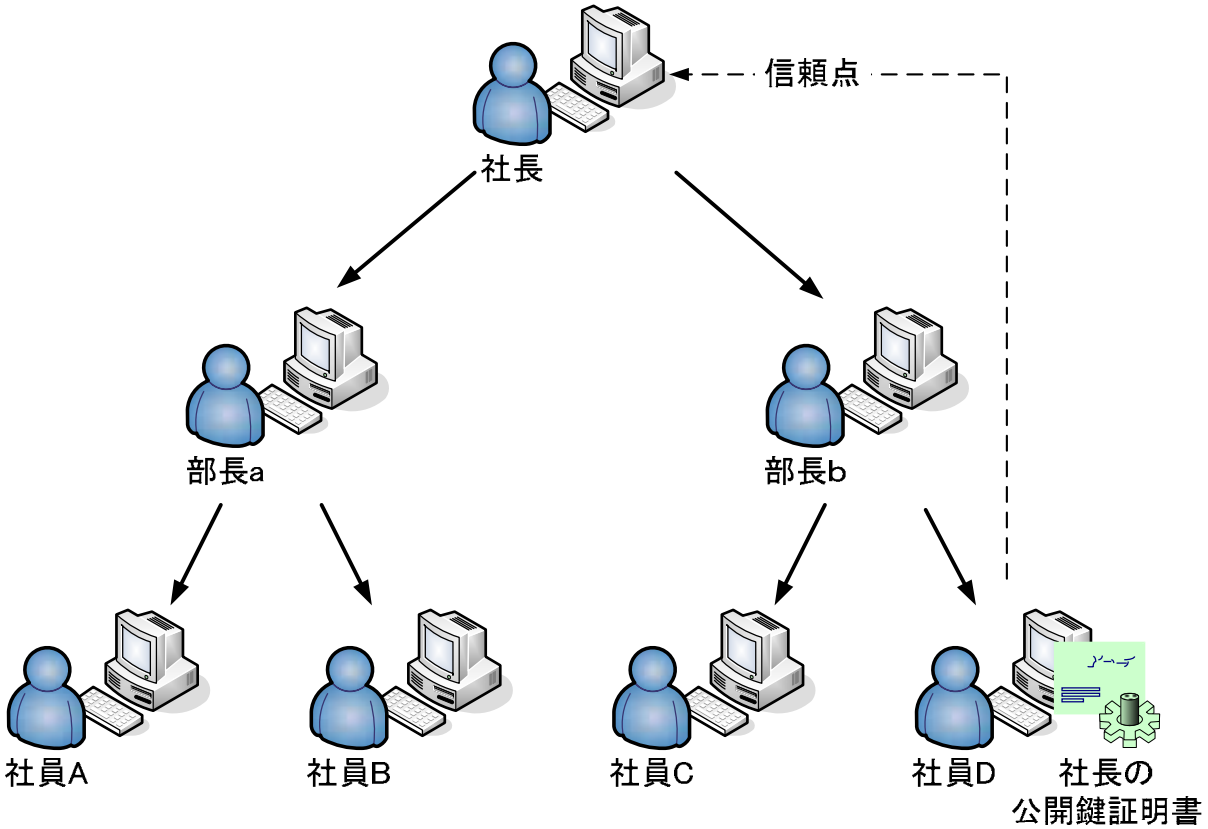
- 認証局CA (Certificate Authority)は公開鍵証明書を発行することにより信頼関係を構築する

階層型（ツリー構造）に構成する階層型モデル



業務形態への対応

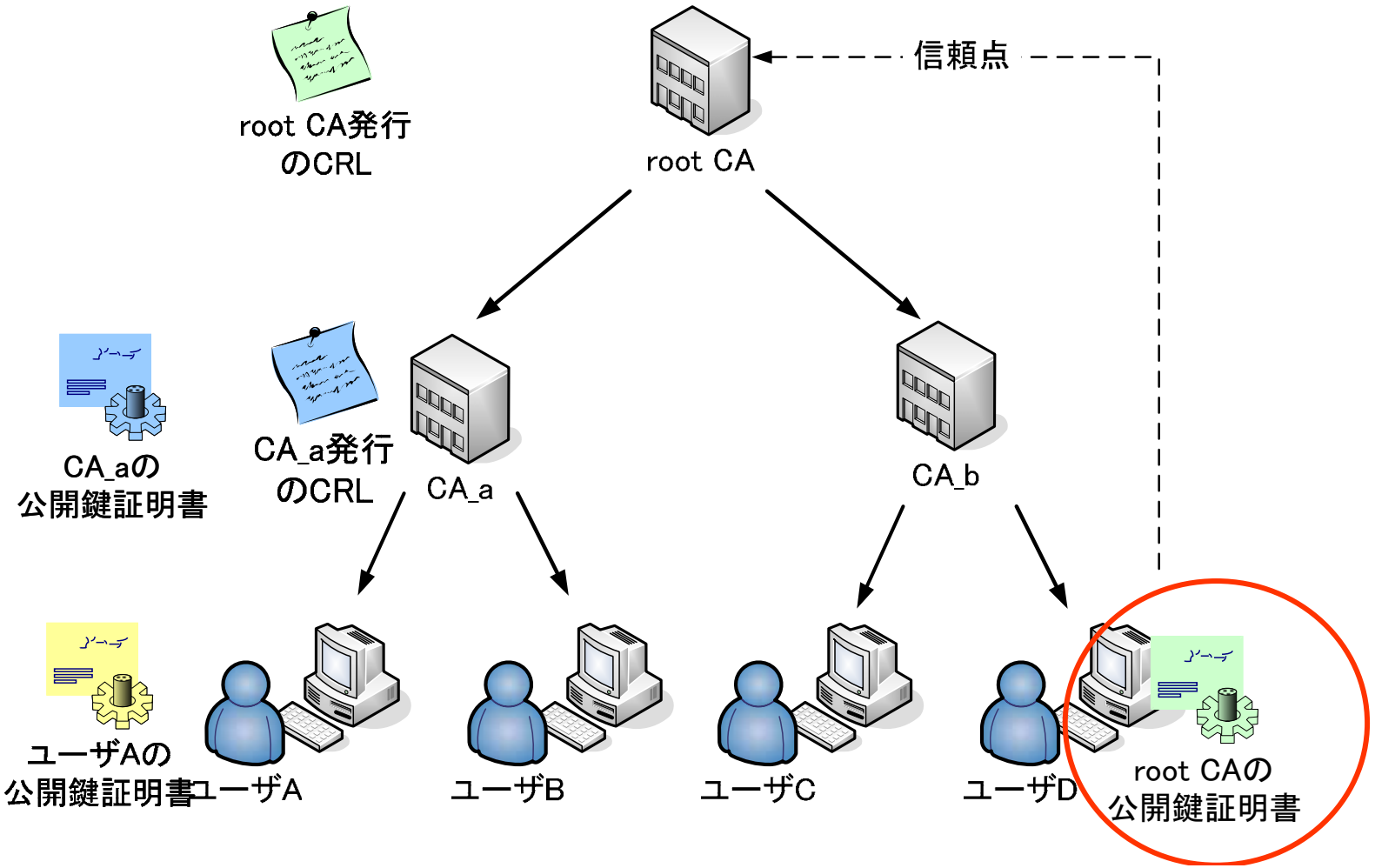
- 階層型モデルは業務形態と対応付けできる



- 信頼関係を階層型モデルで行えば人事異動等による公開鍵証明書の変更にも対応しやすい

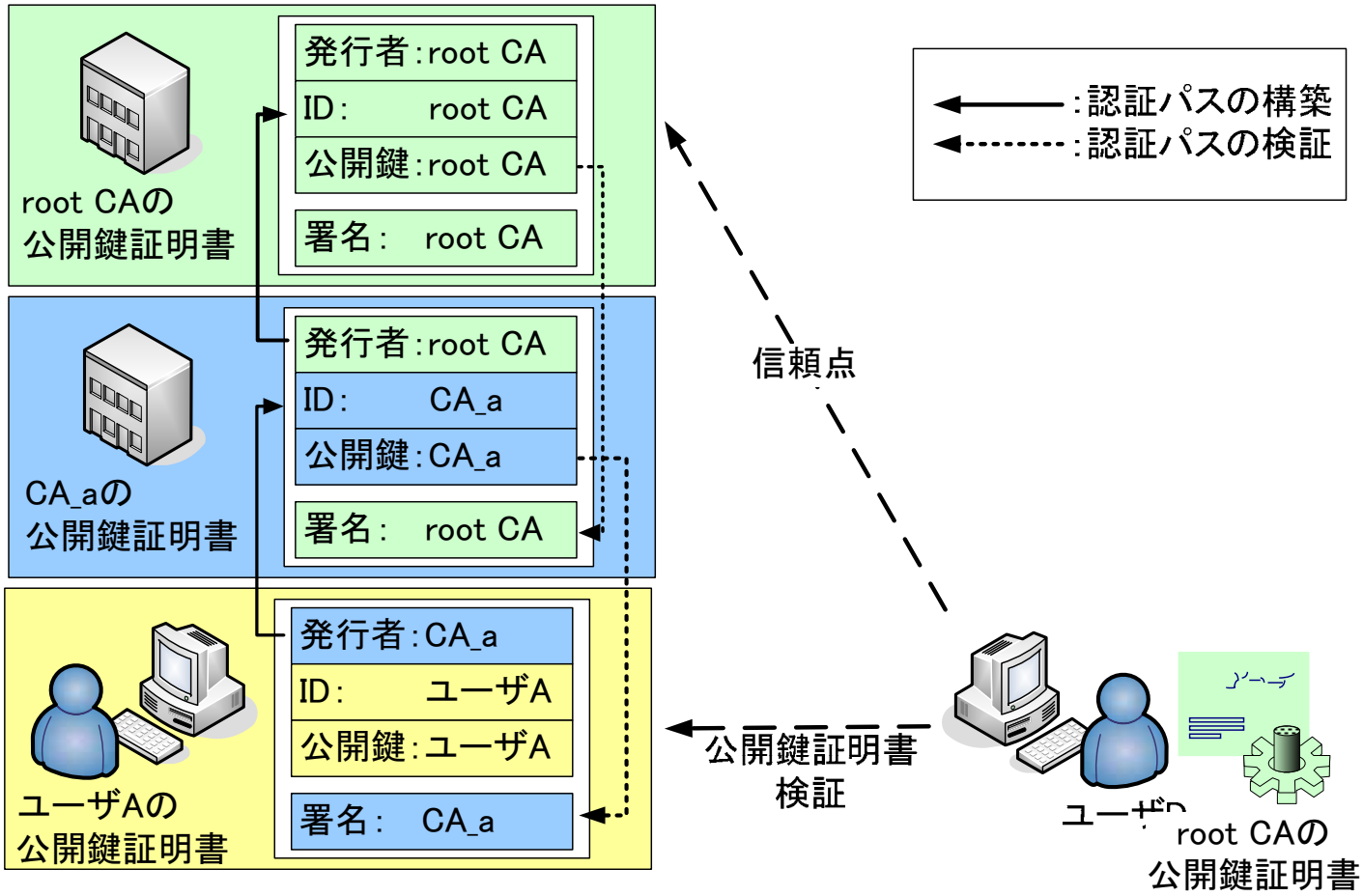
公開鍵証明書の認証

- ユーザDがユーザAを認証する場合



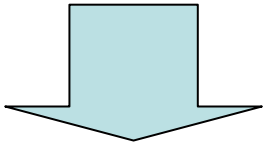
課題1　自己署名の公開鍵証明書

- 自己署名のため公開鍵証明書の発行者が正当であることを検証する方法がない



自己署名の公開鍵証明書偽造

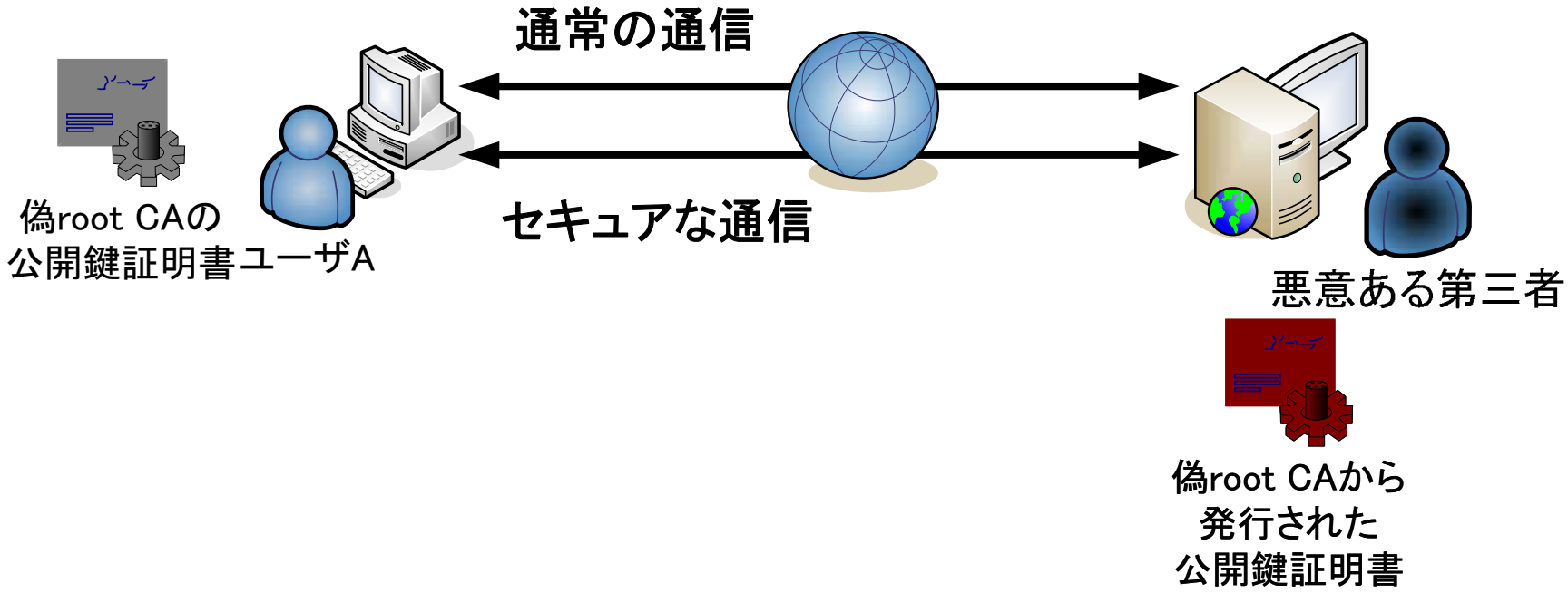
- WindowsではRoot CA の公開鍵証明書がレジストリに保存されている
- レジストリはコマンドラインで書き換え可能
- レジストリを操作しRoot CAの公開鍵証明書の書き換えを行うと確認画面が表示されない



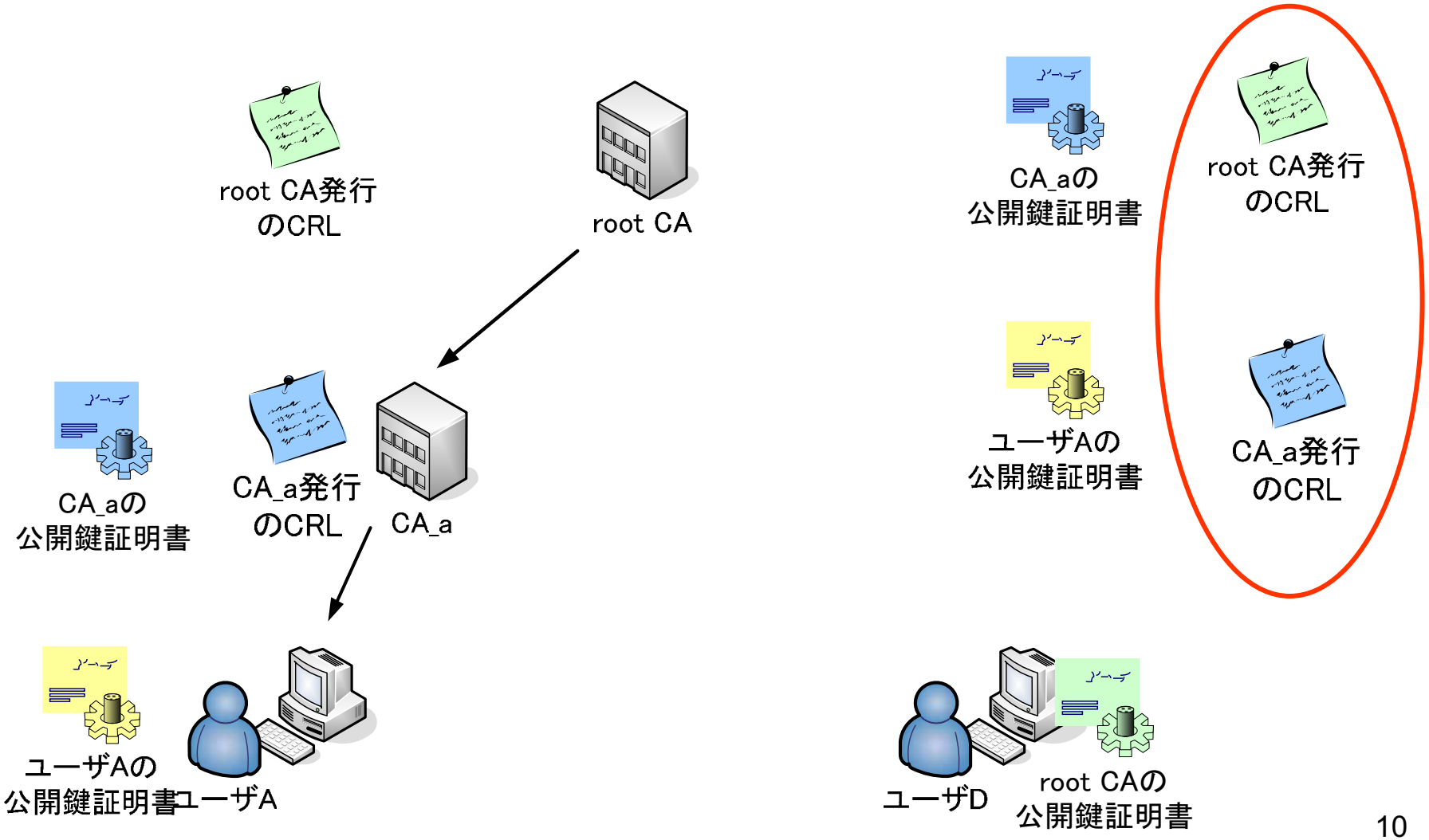
- 悪意あるプログラムで公開鍵証明書を偽造されるとユーザは気がつかない可能性が高い

偽造されたときの問題

- セキュアな通信を行う際，警告メッセージが表示されないため通信相手を信頼してしまう

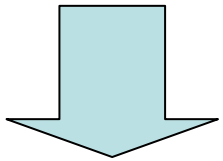


課題2　失効情報


root CA発行
のCRL
root CA
CA_aの
公開鍵証明書
CA_a発行
のCRL
CA_a
ユーザAの
公開鍵証明書
ユーザA
CA_aの
公開鍵証明書
root CA発行
のCRL
ユーザAの
公開鍵証明書
CA_a発行
のCRL
ユーザD
root CAの
公開鍵証明書

失効情報

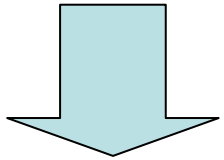
- 発行した公開鍵証明書の有効性を確認するために，公開鍵証明書の失効情報を管理する必要がある



- 失効情報自体の管理が面倒
- 失効情報のデータが大きくなると，有効性の確認時に多くの時間を要する

CRL（Certificate Revocation List)

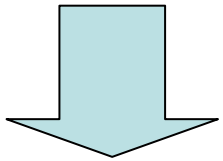
- 公開鍵証明書がCRLに掲載されていないことをもって有効性を確認する
- 事前にCRLを取得しておく
- 決められた周期で発行される



- 最新の情報ではない可能性がある

OCSP（Online Certificate Status Protocol）

- 公開鍵証明書の検証時にリアルタイムで
  OCSPレスポンダへ公開鍵証明書の状態を
  確認するプロトコル
- 返答の偽造の可能性があるため返答にも
  署名を行う



- 失効情報の更新はCRLを利用することが多く，
  必ずしも最新の情報であるとは限らない
- 公開鍵証明の検証のほかにOCSPからの
  返答も検証する必要がある

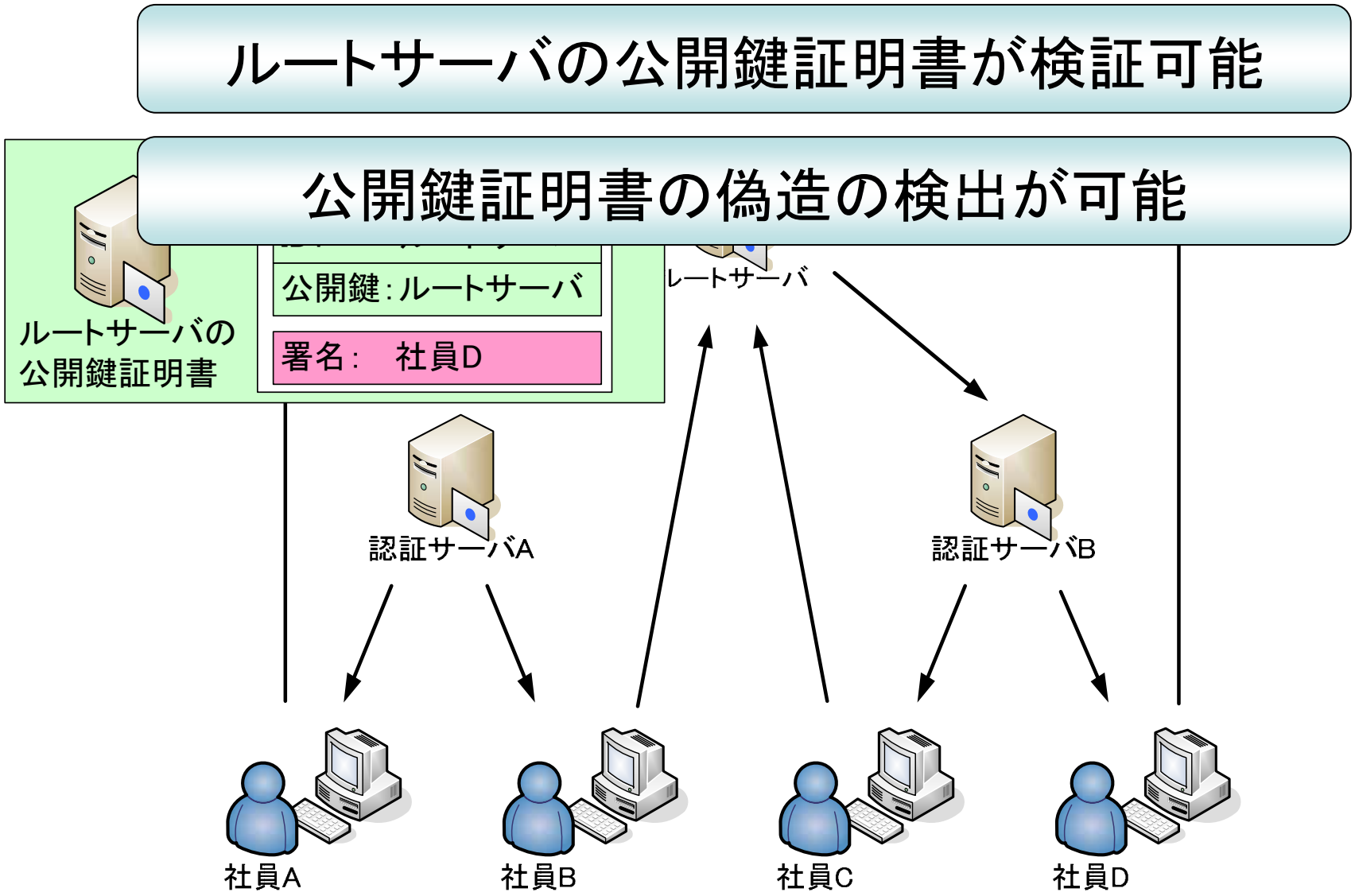
課題

- 企業ネットワークでは以下のことが課題になると考えられる
  - 自己署名の公開鍵証明書を偽造可能
  - 失効情報の管理が面倒
  - 公開鍵証明書の状態について、必ずしも最新の情報とは限らない

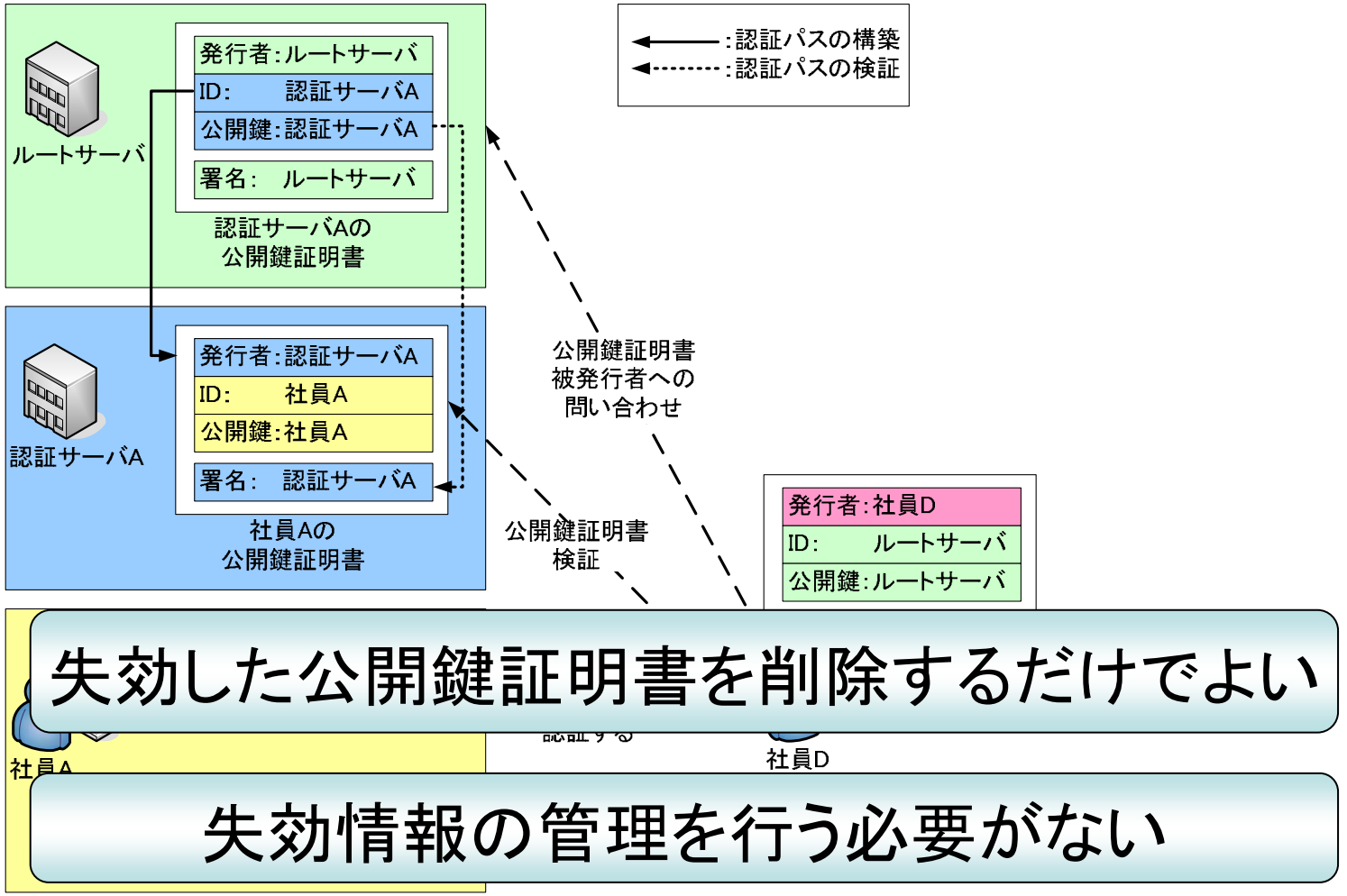
提案方式ASE

- 企業内ネットワークで利用できる認証基盤ASEを提案する
  - 信頼関係を環状にする
  - 公開鍵証明書は発行者が保持し，自ら管理する
  - 信頼関係はオンデマンドで検証する

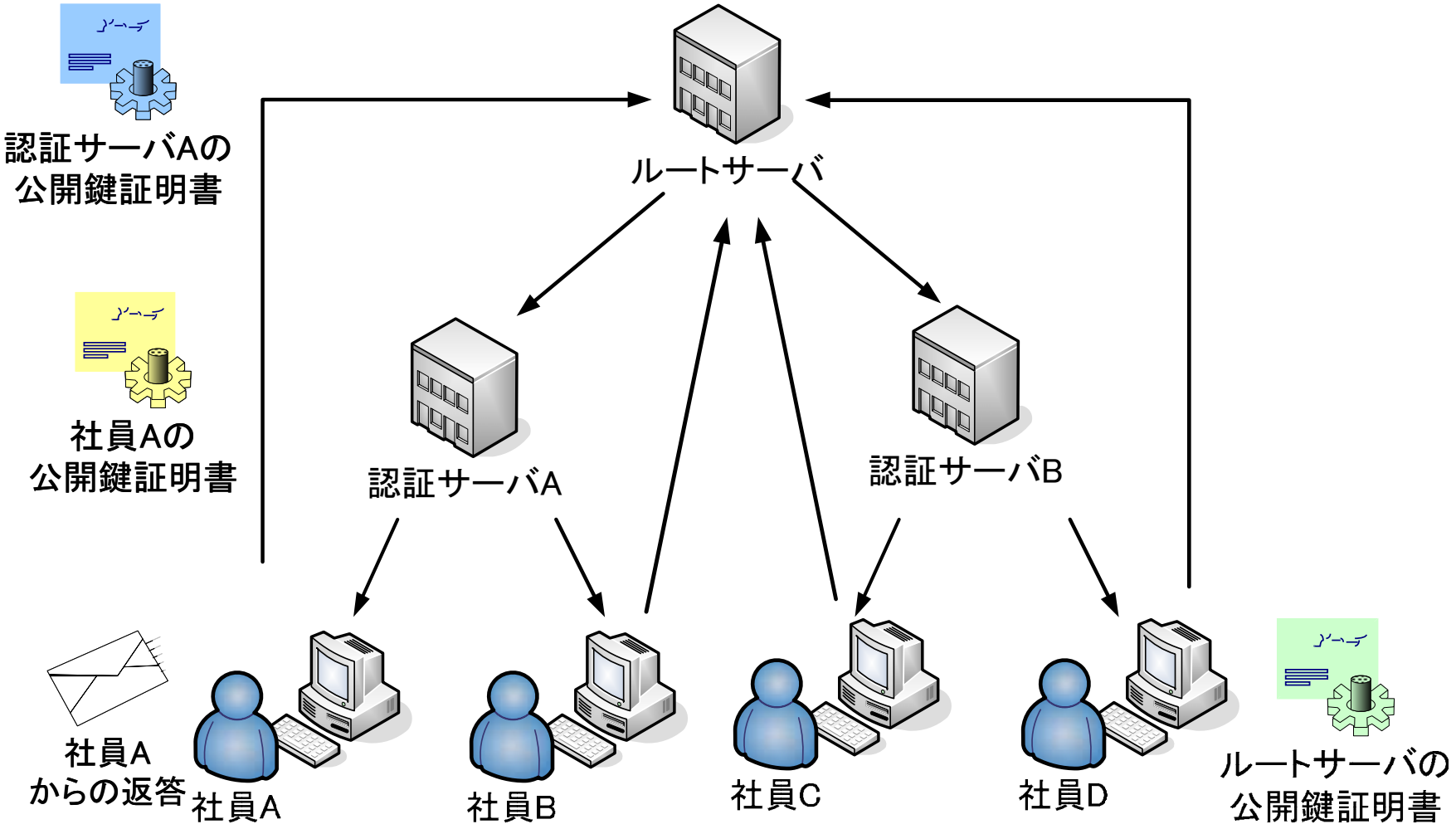
信頼関係を環状化


ルートサーバの公開鍵証明書が検証可能
公開鍵証明書の偽造の検出が可能
ルートサーバの
公開鍵証明書
公開鍵：ルートサーバ
署名：　社員D
ルートサーバ
認証サーバA
認証サーバB
社員A
社員B
社員C
社員D

公開鍵証明書は発行者が保持し，自ら管理する



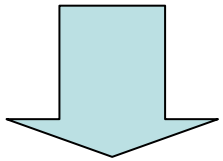
オンデマンド検証



リアルタイム性に優れている

ASEの課題1

- ルート機関の負荷
  - ルートサーバを各社員が署名を行うという性質上，ルートサーバへの問い合わせが多くなり負荷がかかる



- ルート機関を分散化するなどの方法で対策予定

ASEの課題2

- 公開鍵証明書の失効日時の確認不可能
  - 失効情報の管理を行わないため失効日時がわからない
  - 署名データが刻印される時に署名が存在していたことを保証するTime-Stamp Protocolを利用し，公開鍵証明書の失効後に署名が正しく行われていたか検証することができない

評価

	PKI(CRL)	PKI(OCSP)	ASE
リアルタイム性	△	△＋	○
管理コスト	△	△	○
最上位の 公開鍵証明書	検証不可能 ×	検証不可能 ×	検証可能 ○
ルート機関の負荷	○	○	△
失効日時の確認	○	○	△

- 公開鍵証明書を発行者自身が保管しオンデマンドで検証するため リアルタイム性に優れている
- 失効情報の管理を行う必要がない
- 検証者は最上位に位置するルートサーバの公開鍵証明書を自ら検証できる
- ルートサーバを各社員が署名を行うという性質上，ルートサーバへの問い合わせが多くなり負荷がかかる
- 失効情報を管理しないため公開鍵証明書の失効日時が分からない

企業ネットワークで十分有効な手段だと考えられる

むすび

- 企業ネットワークにおいて認証基盤を導入するために以下の認証システムASEを提案した
  - 信頼関係を環状にする
  - 公開鍵証明書はその発行者が保持し，自ら管理する
  - 信頼関係はオンデマンドで検証する
- 今後は，ASEの課題を解決する方法を検討するとともに，ASE実装および検証を行っていく予定である