

OpenPGP作法

特定非営利活動法人
フリーソフトウェアイニシアティブ

鈴木裕信

update:2007-05-8

本日の内容

OpenPGPのWeb of Trust方式による信頼確保の方法はオープンソース開発者に広く受け入れられている。一方で電子署名技術の基礎的理解もなく周囲に流されて使っているというケースも多い。OpenPGPのWeb of Trustは草の根的に使える利点がある半面、個々のユーザが使い方を理解し適切な自己判断できない場合には機能しない。一般にOpenPGPというと暗号技術のうちの秘匿性を中心に解説されるが、オープンソースの配布には完全性が重視され、その補完として認証性が使われる場合が多い。そのような状況を踏まえた上で、電子署名の基本技術から、Web of Trustを実現していくまでのマナーまでを手短に解説する。

高橋メソッド



OpenPGPの
Web of Trust
方式

Web of Trust

友達の

友達

電子署名技術 の基礎的理解

基礎

すぐに飽きる

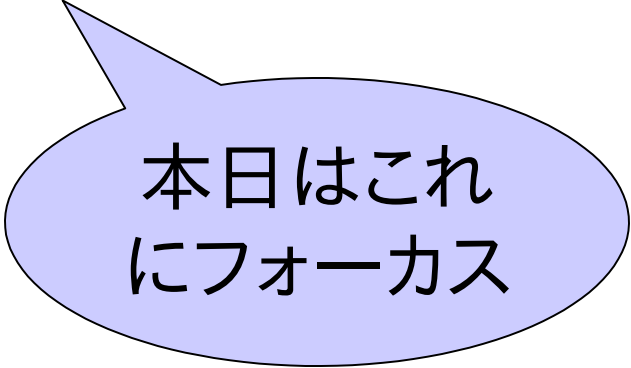
- 高橋メソッドを試してみた
- だめだわ、オレに合わない
- 何事も“ほどほど”が良い

暗号技術

- 暗号技術により次の三つの性質を達成することができる
 - 秘匿性
 - 正しい方法でのみ情報を参照できる
 - それ以外には情報は参照できない
 - 完全性
 - 情報がオリジナルであることを保証する
 - 内容を改ざんするとそれを検知できる
 - 認証性
 - 正しい情報を持つ者であることがわかる
 - 正しい情報を持たないものであることがわかる

電子署名技術

- 完全性
 - 電子署名のスキーム
- 認証性
 - 認証モデル



本日はこれ
にフォーカス

SchemeとPrimitive

- Scheme
 - 1つのことを実現する枠組みレベルのアルゴリズム
 - さらに基本的なアルゴリズムから構成される
- Primitive
 - 基本的なアルゴリズム

Digital Signature

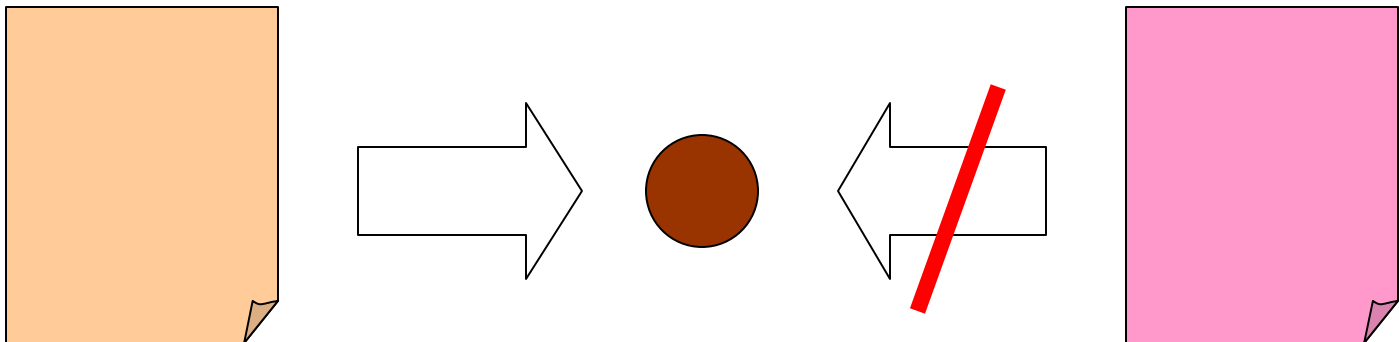


CRYPTREC
だとモジュール
と呼んでいる

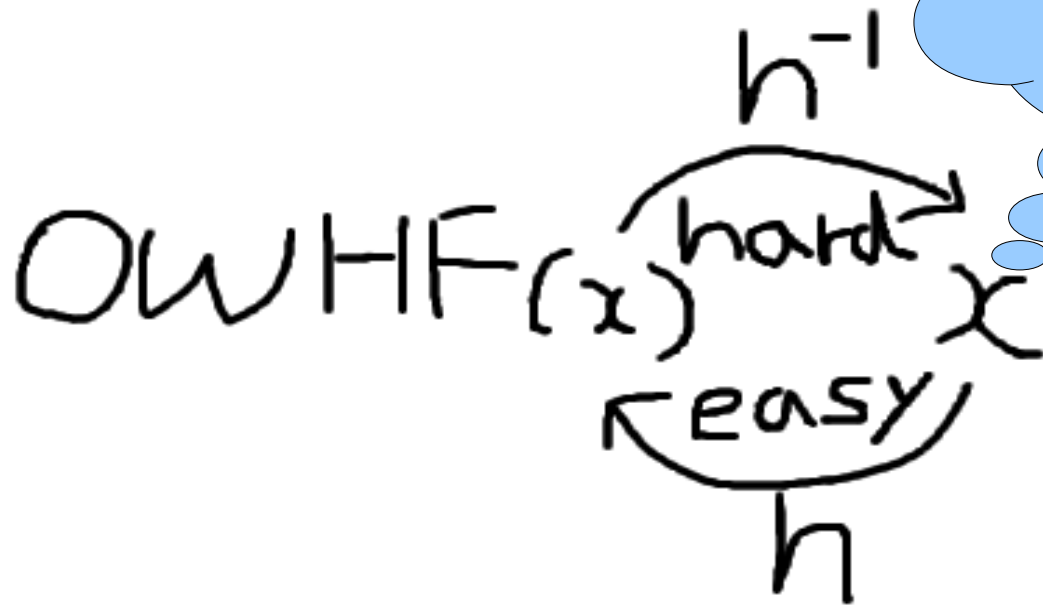
- Digital Signature (電子署名)はスキーム
 - 一方向性ハッシュ関数はプリミティブ
 - MD5, SHA-1, RIPEMD160, etc...
 - 公開鍵暗号法(計算方法)はプリミティブ
 - Diffie-Hellmanベースの計算法 (ElGamal, DSA), RSAの計算法(RSA)

一方向性ハッシュ関数

- One-Way Hash Function (OWHF)
 - ハッシュ値から元のデータを推定できない
- Collision Resistant Hash Function (CRHF)
 - 異なる入力から目的のハッシュ値を作ることができない



もうちょっと数学的に厳密な表記



ここでのeasy/hardは
computational
complexityの用語

もうちょっと数学的に厳密な表記

$$x \neq x'$$

$$\prod_{\forall} [CRHF(x) = CRHF(x')]$$

$$< \epsilon$$

ignoreできる

公開鍵暗号法の考え方

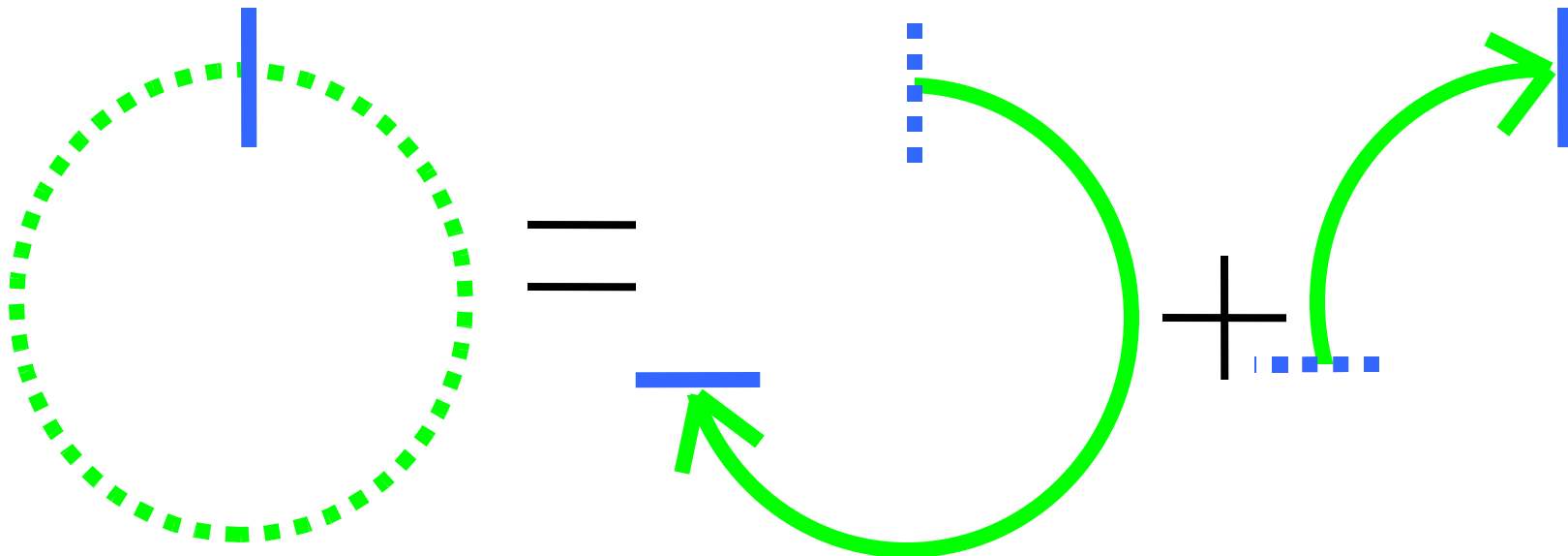
- Trapdoor one way function
 - 一方向にしか計算できないような関数

$$f: \{0,1\}^{l(n)} \times \{0,1\}^n \rightarrow \{0,1\}^{m(n)}$$

円(まる)を思い浮かべてみる

- 出発点からある位置まで円周を進む
- ここから出発の位置に戻るのには
 - ~~通ってきた円周を戻っていく~~
 - さらに円周を元の位置になるまで進む

Not trapdoor one way function



pを法とする

7を法とする

$$4 + 6 \pmod{7} = 3$$

$$3^3 \pmod{7} = 27 \pmod{7} = 6$$

正しいパラメータのペア

- 正しいパラメータのペアのみが正しい計算ができる
- それ以外は間違えた答えが出る

パラメータAとパラメータB

- パラメータAを公開鍵(public key)とする
 - パラメータAを検証鍵(verify key)
- パラメータBを秘匿鍵(secret key)とする
 - パラメータBを署名鍵(sign key)

DSA

- Digital Signature Algorithmの略
 - NISTが作ったアルゴリズム
 - DSS (Digital Signature Standard)の1つ
- 安全性は離散対数問題に依存

$$y = g^x \pmod{p}$$

$x \rightarrow y$ *easy*

$y \rightarrow x$ *hard*

RSAと同じ
hardさ

RSA



gnupgでは
eは通常41

- RSA公開鍵暗号法を署名に応用
- $n=pq$, $ed \equiv 1 \pmod{(p-1)(q-1)}$
 - e 検証鍵, d 署名鍵
 - 暗号化 $E = M^e \pmod n$ / 復号 $M = E^d \pmod n$
- 安全性はRSA問題(素因数分解問題)に依存

$$n = pq$$

$$pq \rightarrow n \quad \textit{easy}$$

$$n \rightarrow pq \quad \textit{hard}$$

入力すべてに計算するか？

- たとえば1MBのサイズの入力すべてに対して電子署名(公開鍵暗号の計算)をさせるのは非現実的
- ハッシュ値を作りそのハッシュ値に対し署名をかける
 - SHA-1は160bit (20byte)
 - 20byteに対しRSA-1024で検証するとPentium 4-2.4GHzで2-3ms以下



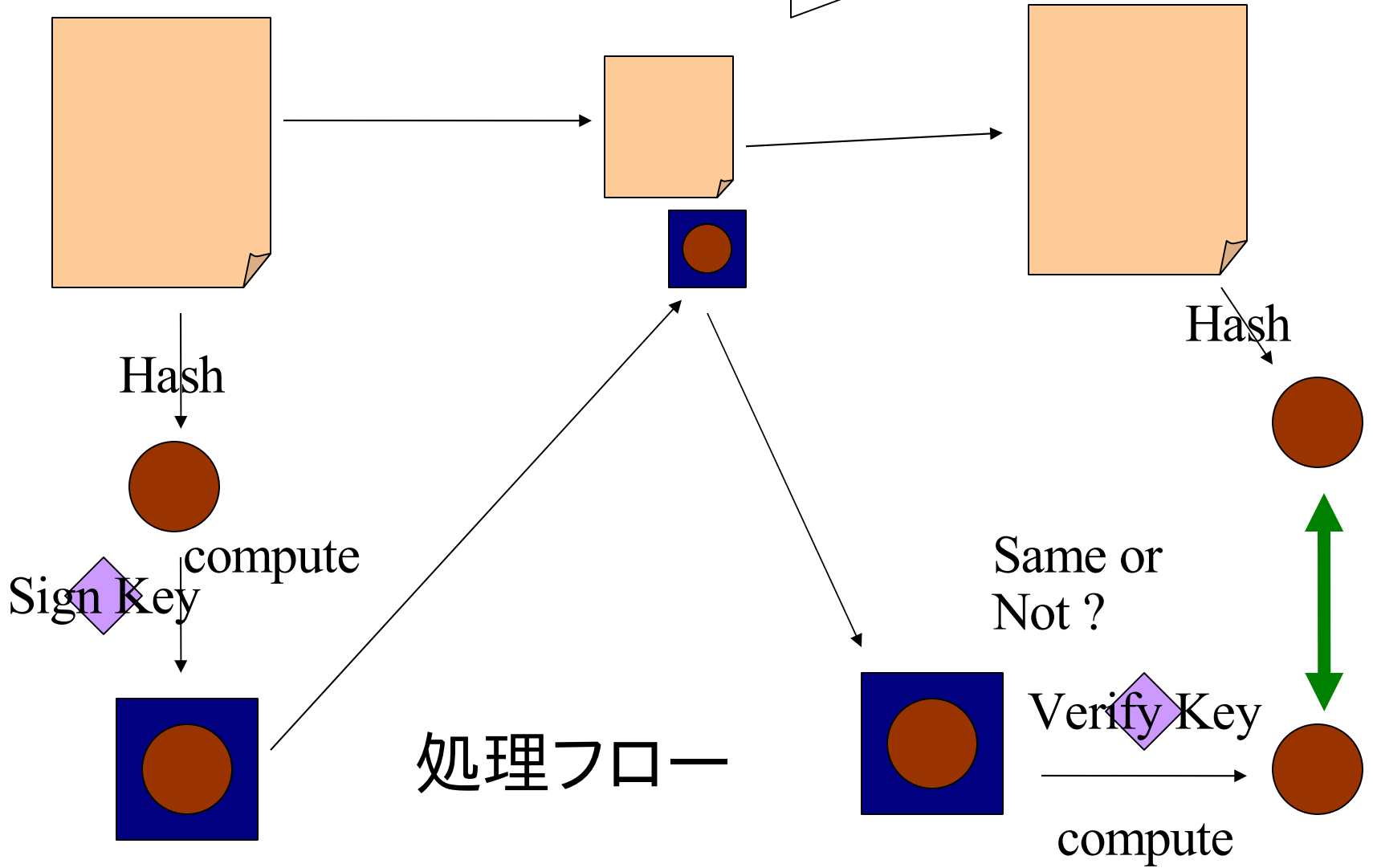
おさらい

署名鍵と検証鍵

署名鍵で署名したデータは書名鍵と対となっている
検証鍵でのみ正しい値が求められる
自分で署名鍵と検証鍵の一对を生成する
署名鍵は自分のみ持っている
検証鍵を相手に渡す
自分の署名鍵でデータを署名する
相手に署名したデータを渡す
相手は検証鍵で検証する

Signer

Verifier



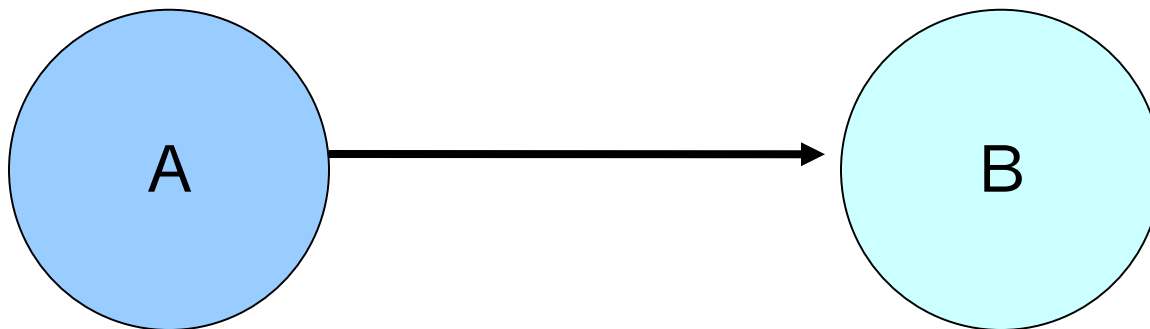
正しい鍵のペアであった

- 完全性は確保した
 - でも、その鍵の持ち主は誰？
 - その鍵に書き込まれている所有者属性なんて、ものは単なるデータ列にしか過ぎない

認証性

- ここでの認証性とは
 - 誰かが誰かだということを示すこと
 - それを確認できること
- 人が人を認証するとみる心こと
 - 人が介在する
 - 計算量云々の問題ではない世界
 - ヒューマンクリプト (by 今井秀樹先生)

署名していることと認証とは違う



AからBへ署名している

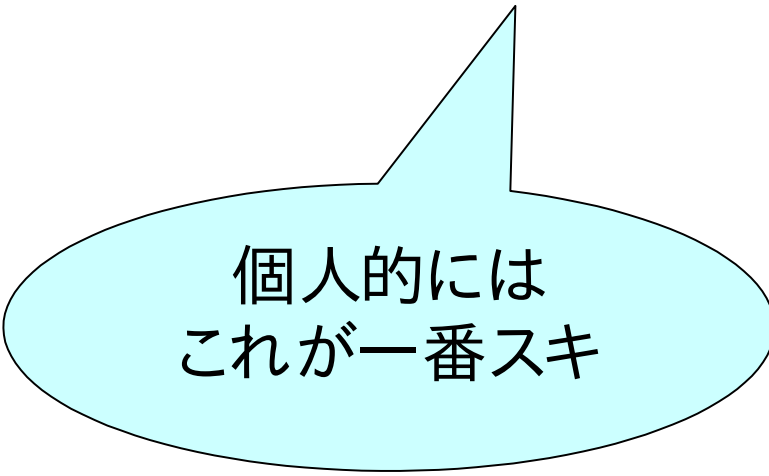
計算可能にしているだけで、それ以上でもそれ以下でもない

それに意味を持たせるのは人間の行為



認証モデル

- たぶん認証モデルはいくらでも作れる
 - 要は決めの問題
 - X.509, SPKI, Web of Trust, etc



個人的には
これが一番スキ

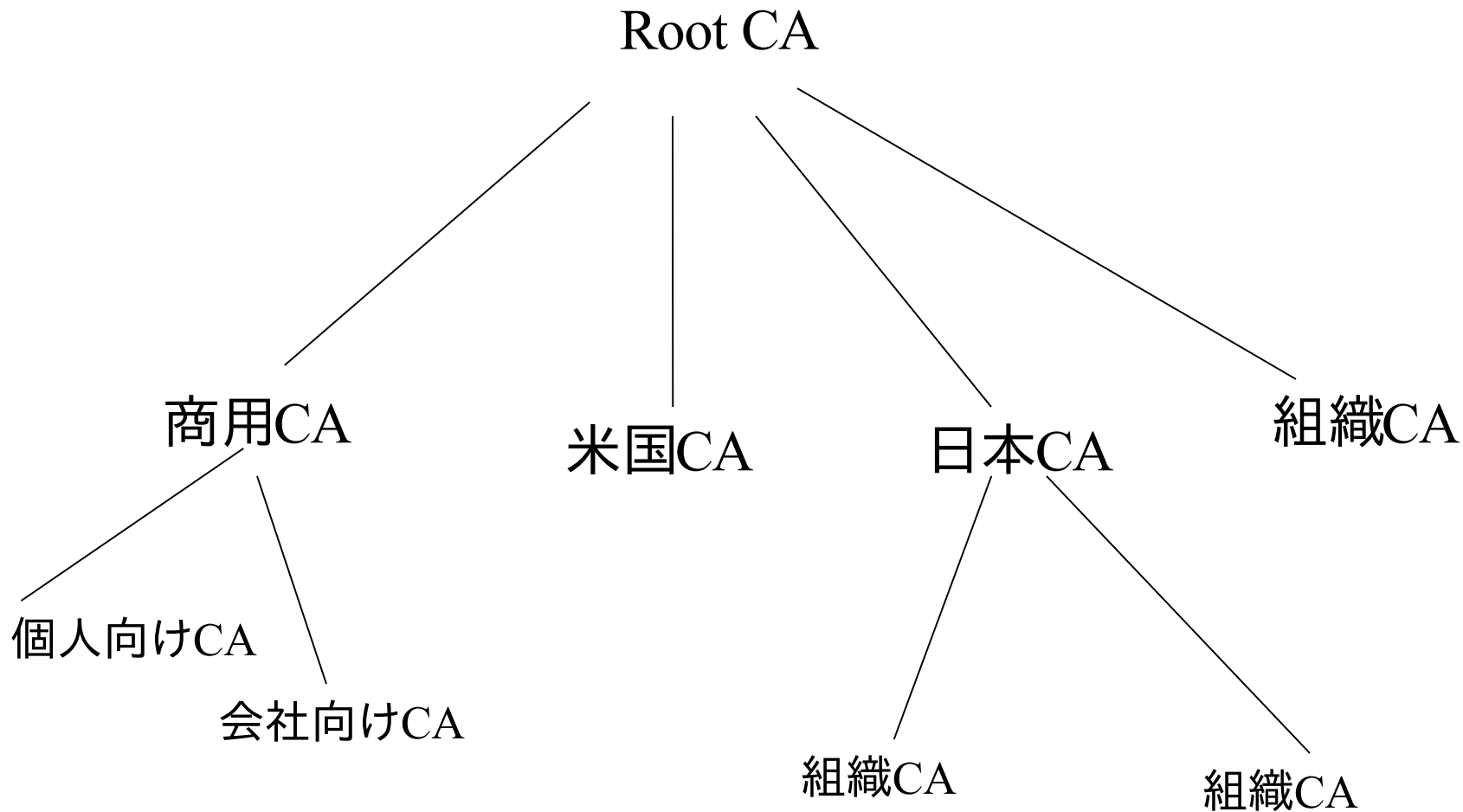
X.509

- 電子署名認証のためのモデルとして正式に決められた最初の規格
 - CCITT X.500シリーズの規格だからかなり古い
 - X.500の野望である世界を一つのツリー構造で表現する
 - でも幻想だよこれ

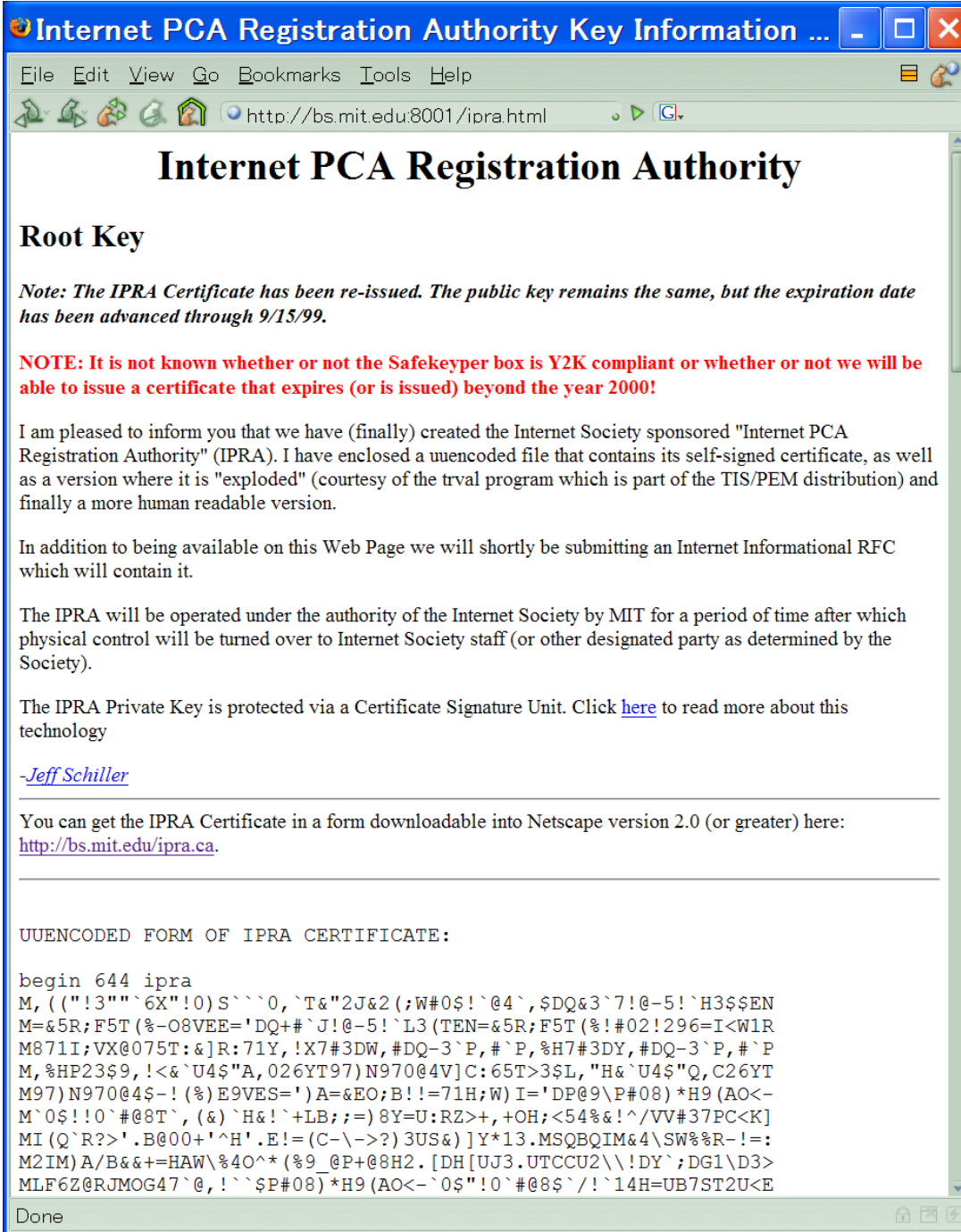
実際のX.509を使った運用

- ISOCのIPRAが実施
 - X.509規格は運用母体などを定義はしていない
 - TIS/PEMのためにRFC1422で認証構造を規定
 - 1994年9月よりに運用
- ルートからツリーで辿れる認証モデル運用
 - 唯一のrootからの本物のツリー構造
 - 親が子を署名して信頼のツリーを作る
 - JIS@MIT.EDUの所でルート鍵を管理
 - 日本では認証実用化実験協議会(ICAT)が日本のツリーの開始ノード

ツリー構造



IPRA Root Key



The screenshot shows a Netscape browser window with the title "Internet PCA Registration Authority Key Information ...". The address bar shows "http://bs.mit.edu:8001/ipra.html". The page content includes a title "Internet PCA Registration Authority", a section "Root Key", a note about a re-issued certificate, a red warning note about Y2K compliance, a paragraph explaining the creation of the IPRA, a paragraph about an upcoming RFC, a paragraph about the IPRA's operation, a paragraph about the private key's protection, a signature by Jeff Schiller, a link to download the certificate, and a section for the unencoded form of the IPRA certificate.

Internet PCA Registration Authority

Root Key

Note: The IPRA Certificate has been re-issued. The public key remains the same, but the expiration date has been advanced through 9/15/99.

NOTE: It is not known whether or not the Safekeyper box is Y2K compliant or whether or not we will be able to issue a certificate that expires (or is issued) beyond the year 2000!

I am pleased to inform you that we have (finally) created the Internet Society sponsored "Internet PCA Registration Authority" (IPRA). I have enclosed a uuencoded file that contains its self-signed certificate, as well as a version where it is "exploded" (courtesy of the trval program which is part of the TIS/PEM distribution) and finally a more human readable version.

In addition to being available on this Web Page we will shortly be submitting an Internet Informational RFC which will contain it.

The IPRA will be operated under the authority of the Internet Society by MIT for a period of time after which physical control will be turned over to Internet Society staff (or other designated party as determined by the Society).

The IPRA Private Key is protected via a Certificate Signature Unit. Click [here](#) to read more about this technology

-[Jeff Schiller](#)

You can get the IPRA Certificate in a form downloadable into Netscape version 2.0 (or greater) here:
<http://bs.mit.edu/ipra.ca>.

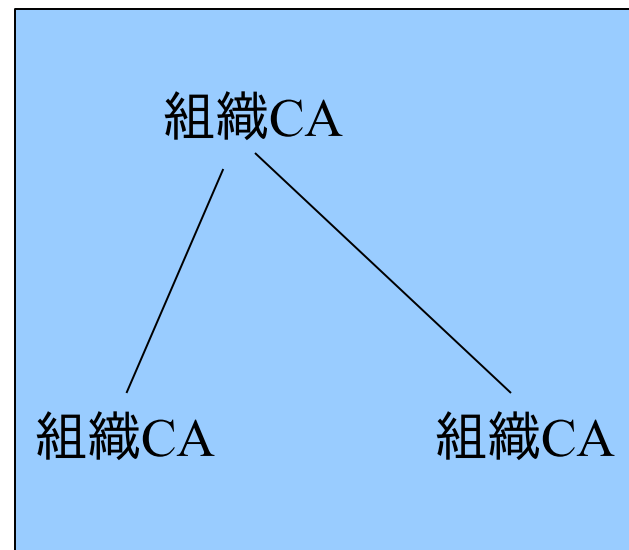
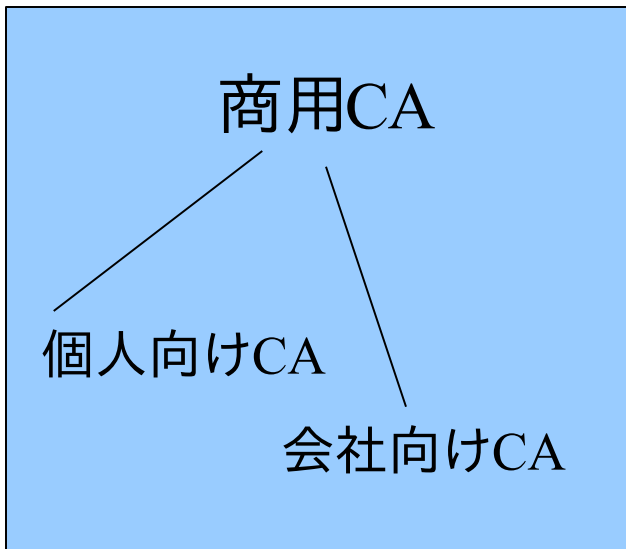
UUENCODED FORM OF IPRA CERTIFICATE:

```
begin 644 ipra
M, ("!3""`6X"!0)S``0,`T&"2J&2(;W#0$!`@4`, $DQ&3`7!@-5!`H3$$EN
M=&5R;F5T(%-08VEE=`DQ+#`J!@-5!`L3(TEN=&5R;F5T(%!#02!296=I<W1R
M871I;VX@075T:&]R:71Y,!X7#3DW,#DQ-3`P,#`P,%H7#3DY,#DQ-3`P,#`P
M,%HP23$9,!<&`U4$"A,026YT97)N970@4V]C:65T>3$L,"H&`U4$"Q,C26YT
M97)N970@4$-!(%)E9VES=')A=&EO;B!!=71H;W)I='DP@9\P#08)*H9(AO<-
M`0$!!0`#@8T`,(&)`H&!`+LB;;=)8Y=U:RZ>+,+OH;<54%&!^/VV#37PC<K]
MI(Q`R?>'`B@00+'^H'.E!=(C-\->?)3US&)]Y*13.MSQBQIM&4\SW%R-!=:
M2IM)A/B&+ +=HAW\%40^*(%9_@P+@8H2.[DH[UJ3.UTCCU2\`!DY`;DG1\D3>
MLF6Z@RJMOG47`@,!``$P#08)*H9(AO<-`0$"!0`#08$/!\`14H=UB7ST2U<E
```

1999年9月15日

- この日をもってRoot CAから辿る認証ツリーは消滅
 - 世界はカオスへ

現在の認証構造



組織CA

商用CA

商用CA

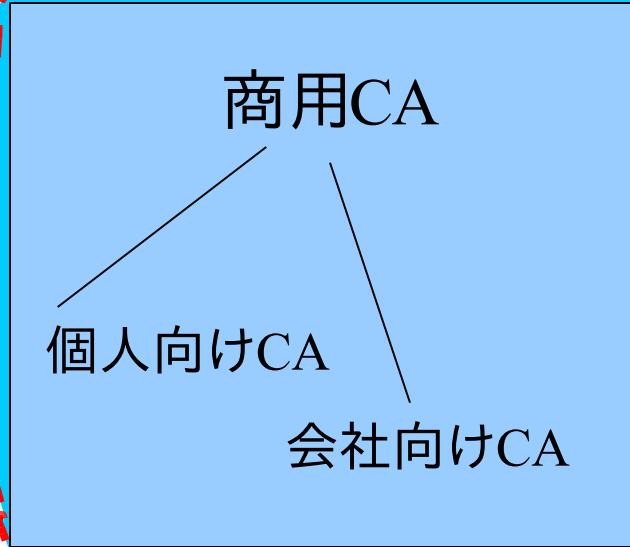
組織CA

組織CA

商用CA

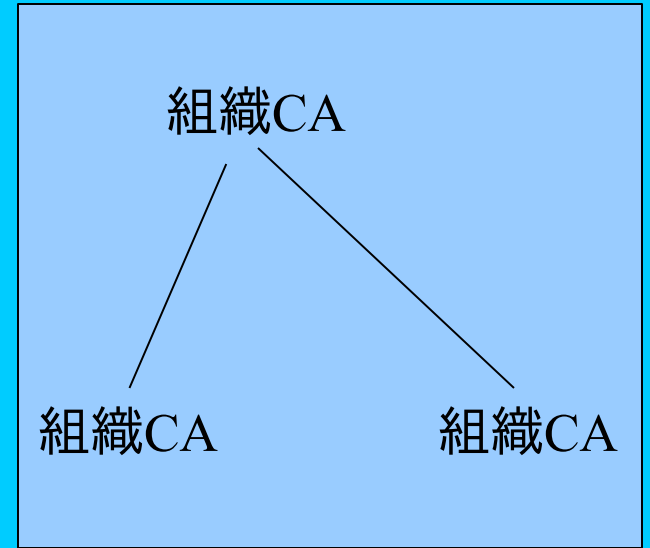
商用CA

唯一の違い



ブラウザに
インストール
されている

組織CA



商用CA

商用CA

組織CA

組織CA

商用CA

商用CA

ブラウザに登録されているCA



そもそも信用できるの？

- どんな基準で入っているわけ？
- どんな確認をしてCAリストに入れたわけ？
- ブラウザに含まれるCA情報をダウンロードする際に何か確認しているわけ？

思考停止

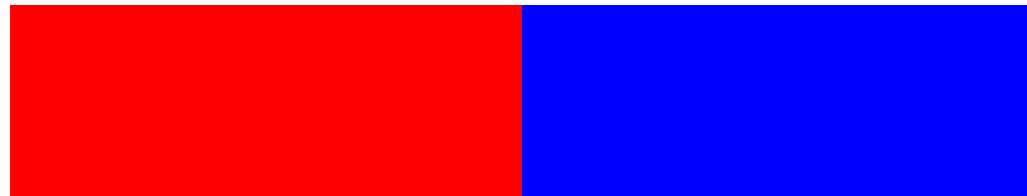
- 正しいものは正しいなぜなら正しいからだ
 - ではそこまで百歩譲りましょう

そこから発行された証明書は正しいの？

- 書類手続きをすればいいだけで、それはいくらでも騙せる方法はある
- 認証を出すところを一箇所騙せばそれでおわり
 - 偽者マイクロソフトに対しマイクロソフトの証明書を出した

あんしん・あんぜん

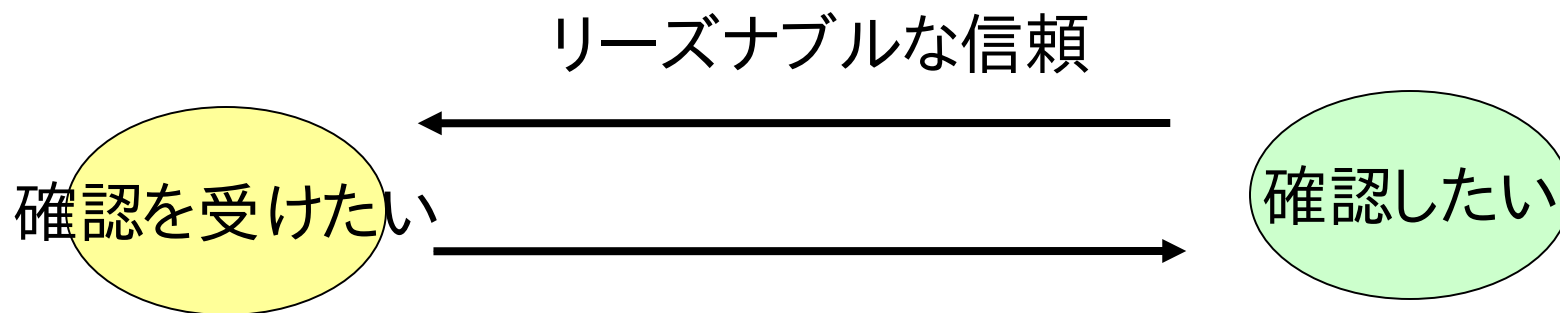
- 所詮は100%なんてものはありません
- 信用できない・信用できる
 - 単純に二値を取るようなものではない
 - グラデーションのように変化するもの



授信モデルを考えてみる

- 有向グラフを考えてみる
 - 署名する方向を有向グラフと考える
 - どのようなグラフ構造が信頼性が高くなるかなどのクライテリアを考えたりするのは学術的に面白いテーマ
 - 修士論文や博士論文に最適なテーマ
 - まだ誰も発表していないぞ！今がチャンス
 - まだパスを見つけるレベルまでしかやってないぞ！

相互確認



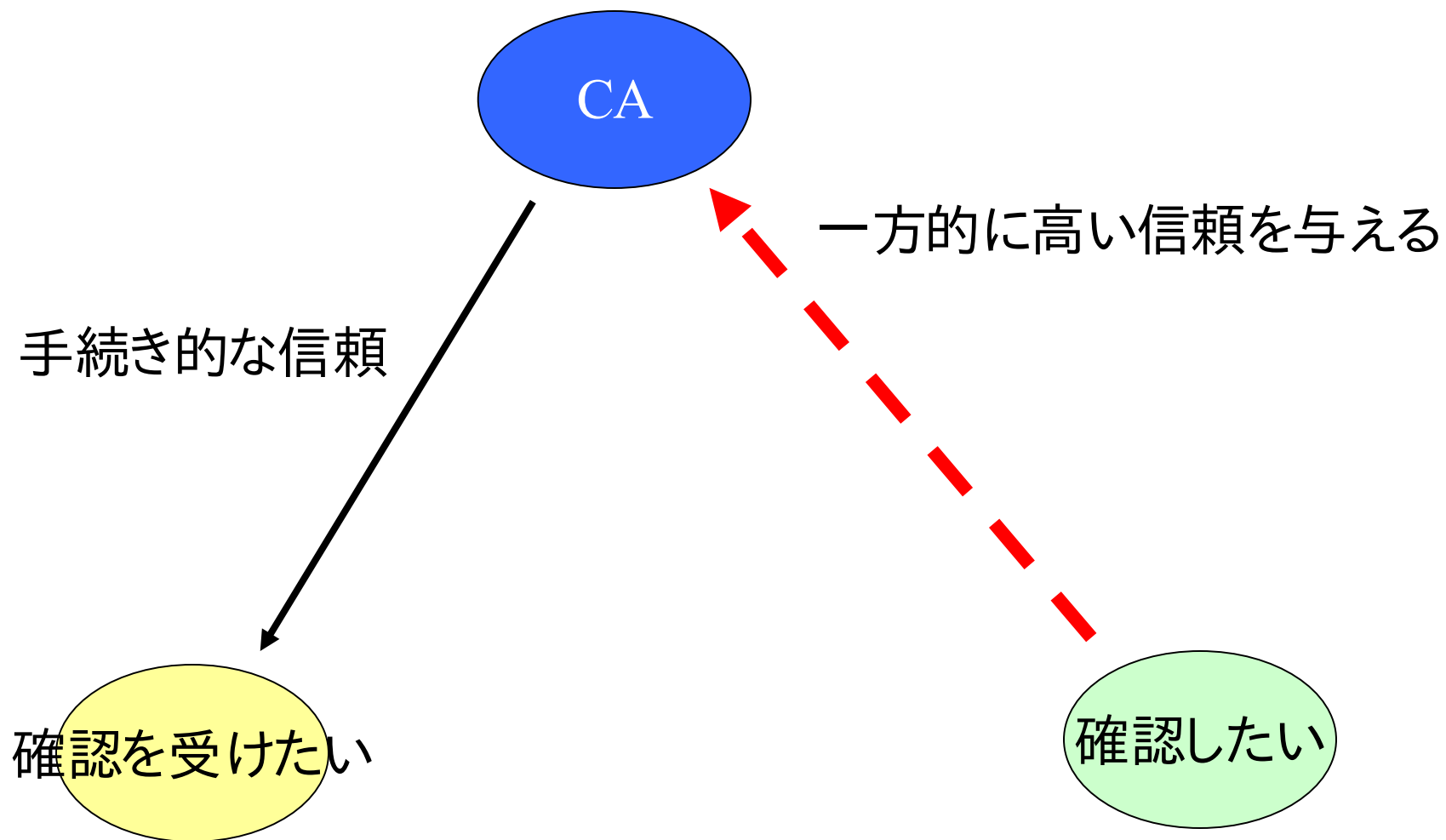
紹介者を使う

紹介者

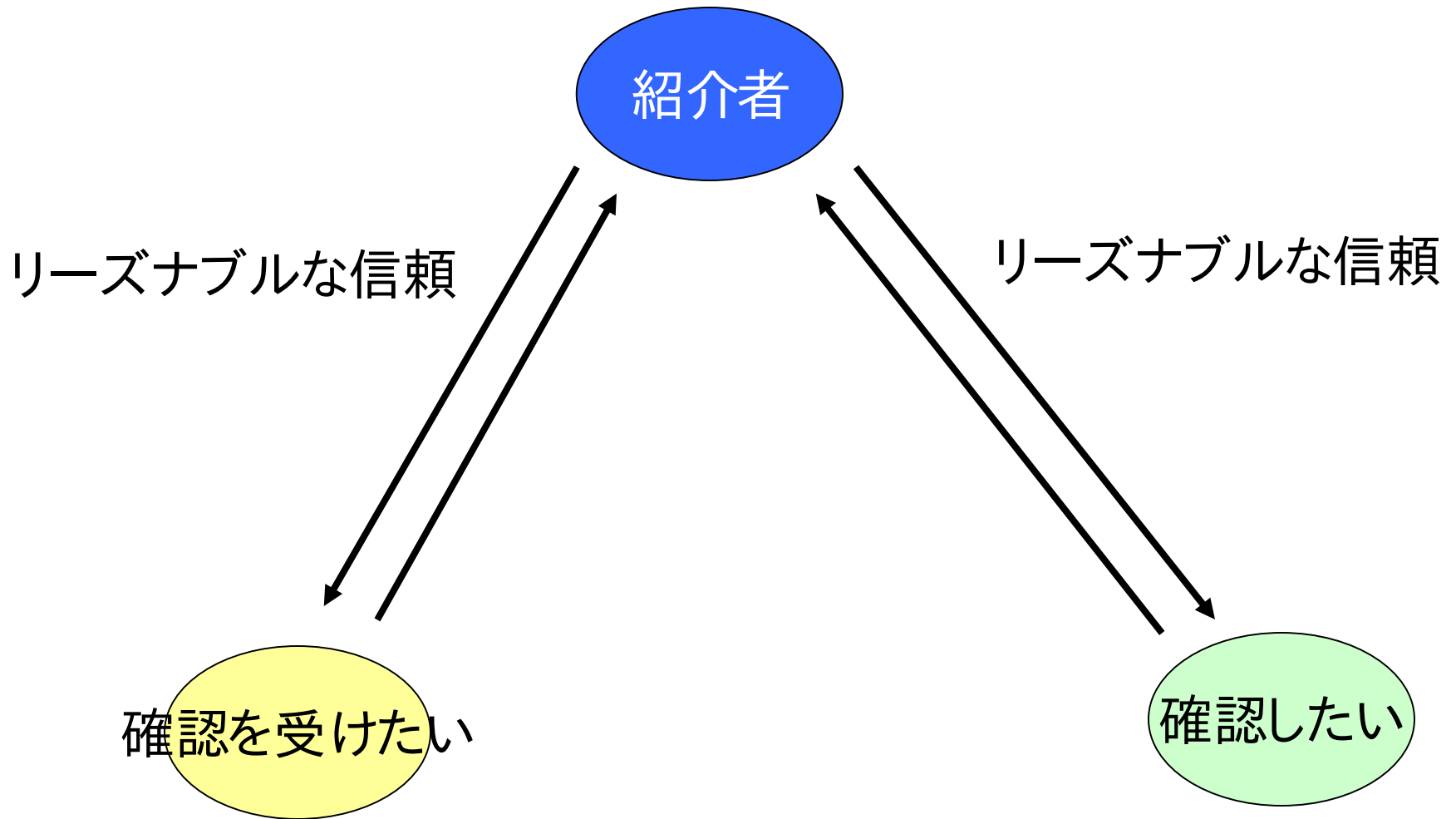
確認を受けたい

確認したい

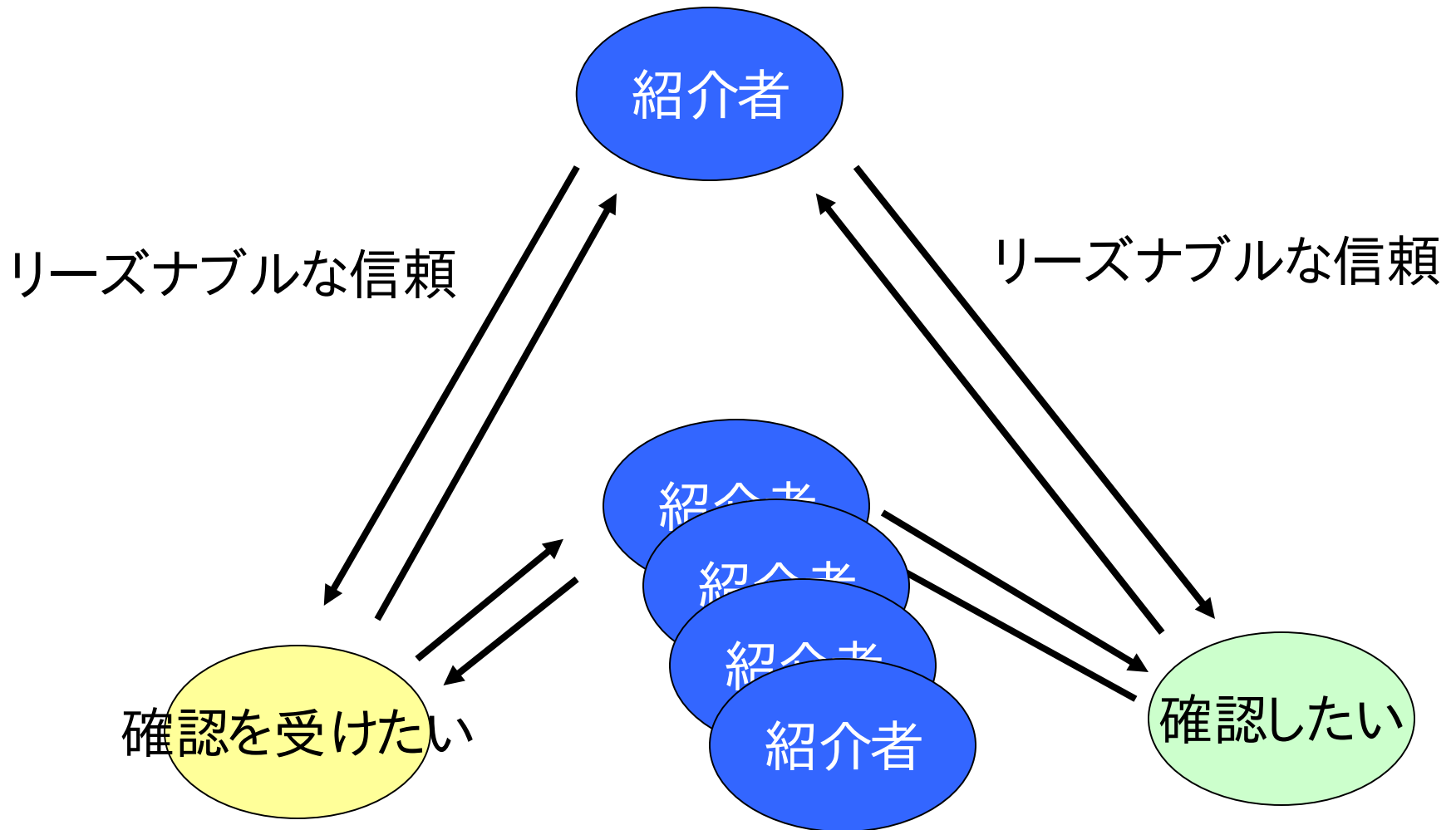
今のX.509の運用



Web Of Trust (simple)



Web Of Trust (multiple routing)



友達の友達は友達

- 信頼のチェーンを使う
- 一対一の場合はとてもわかり易い
- チェーンにした場合、何をどう信頼するかは単純な基準がないので今ひとつわかりづらい
 - 単純に多くのリンクが来ているのが信頼されるわけではない(のは、言わなくてもわかるよね)
 - 自分から信頼できるルートを通して信頼できるようになっているかどうかポイント
 - そのルート数が増えれば増えるほど偽造も確率的に難しくなる

サードパーティ署名

- 信頼できる手続きを得てサードパーティに署名してもらう
- サードパーティを信頼する人には有効
 - 気持ち的にX.509のCAだと思う人がいるがあくまでもWeb of Trustの一種

```
sig C368EA31 Boeing CERT <compsec@maple.a1.boeing.com>  
sig 71D5F43A NIIBE Yutaka <gniibe@fsij.org>  
sig 2F99027D Susan B. Levy Haskell <sblh@nts.umn.edu>
```

1stKSMS
2000-05-22

友達づくり

- 親しい友人同士でサインしあう
- キーサインパーティと呼ばれる署名会を開く
 - 全員の鍵と本人確認をして回る
 - $O(n^2)$ 方式: 鍵の情報と本人を確認できるものを持って総当たりしていく
 - 自己紹介もかねているので面白いけどね
 - ササーマン方式
 - $O(n)$ 方式: 最初にリストを作り、本人かどうかの確認を行う担当がいて確認し、リスト上の情報を確認する
 - キーサインパーティーをやったらPGPを開発したPhilip Zimmermannが一番うんざりしていた・「もっと効率の良い方法を考えてくれ」と同行していたL. Sassaman に指示した

チェック・チェック

- Finger printチェック
 - 公開鍵のパラメータにつかっている値にハッシュ値をかけたもの
- 本人チェック
 - 写真つきの公的な身分証明書
 - 運転免許書やパスポート

キーサインパーティーの
オーガナイザーが何をすべきか
指示するのでそれに従うのがベスト

キーサインパーティーの 実際を見て思った個人的感想

- ササーマン方式がベースになっているようだが
 - いまひとつ理解度が足りていない気がする
 - ササーマンというよりはオレ様ルールじゃないかなあ
- そもそも信頼ってなんだろう？
 - そのサインをするコミュニティの署名に対する理解度が、そのコミュニティでの署名関係の信頼性に反映されているような気が...

OpenPGP公開鍵内部構造モデル

自分の検証鍵

ユーザID(任意の文字列)

自分の署名鍵でユーザIDを署名した値

Bさんの署名鍵でユーザIDを署名した値

Cさんの署名鍵でユーザIDを署名した値

Dさんの署名鍵でユーザIDを署名した値

(注) キーサインを行った時にどうユーザIDへ署名されているかに着目して説明しています。OpenPGPのパケット構造はもっと複雑になっています。

OpenPGP公開鍵内部構造モデル2

自分の検証鍵

ユーザIDその1(任意の文字列)

自分の署名鍵でユーザIDその1を署名した値

Cさんの署名鍵でユーザIDその1を署名した値

Dさんの署名鍵でユーザIDその1を署名した値

ユーザIDその2(任意の文字列)

自分の署名鍵でユーザIDその2を署名した値

Bさんの署名鍵でユーザIDその2を署名した値

Dさんの署名鍵でユーザIDその2を署名した値

ユーザIDその3(任意の文字列)

自分の署名鍵でユーザIDその3を署名した値

公開鍵の交換

- メールでやり取りしても良いし
- Webに貼り付けておいてURLを示すのもよい
- でももうちょっとスマートな遣り方があって
もいいんじゃないか？

公開鍵サーバ

- 一々公開鍵をやり取りするのは面倒なので公開鍵サーバをつくり、そこにプールする
 - 公開鍵を保持しておくだけで何の保証もしていない
 - gnupgコマンドは `keyserver`, `--send-key`, `--recv-key`などのオプションでダイレクトに呼び出せる
 - Webブラウザ経由でもアクセスできる
- 日本国内のサーバ
 - `pgp.nic.ad.jp`
 - JPNICに置かせてもらっているサーバ
 - `openpkd.org`
 - Hironobu's OpenPGP Public Keyserver

pgp.nic.ad.jp

PGP KEYSERVER TOP – Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://pgp.nic.ad.jp/

[\[JAPANESE VERSION \]](#) [\[JPNIC HOME PAGE \]](#) [\[PGP HOME PAGE \]](#)

PGP PUBLIC KEYSERVER



[日本語ページへ](#)

[2004-09-23] Keyserver is BACK!!

- [PGP.NIC.AD.JP FAQ](#) [Japanese Language]
- [GO PGP KEYSERVER](#) [[Japanese Language](#)] [[English Language](#)]
- [No cgi-bin command anymore.](#) [Japanese Language]
- GNU Cryptography software ftp site : <ftp.gnupg.org>
- PGP international ftp site : <ftp.pgpi.com>
- [GO related documents about this pgp keyserver.](#)

Done

openpkd.org

Welcome To OpenPKSD.ORG – Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://openpkd.org/

OpenPKSD.ORG

OpenPGP Public Keyserver Development Project Since 2001

Find your public key

KeyID (32bit/64bit) and mail address in UserID are available for public key retrieving .
Example:

- 0x02912C53
- 0xBD58B69202912C53
- hironobu@h2np.net

[Picture](#) of OpenPKSD system development load map.

Statistics of openpkd.org using awstats system.

YOUR EMAIL ADDRESS IS SAFER THAN YOU THOUGHT

2005/Aug/05

I heard some users are afraid of "Spammers got my email address from keyserver because huge email address list could be downloaded from keyserver."

It's not true for OpenPKSD. Spammer can't download email list from OpenPKSD. Everyone can retrieve one public key and its information using an email address as a search key. Also a keyid is available for a search key. If someone search public keys using random keyids or random email addresses, OpenPKSD server goes slow connection for a user who has been trying frequent connections to OpenPKSD keyserver because OpenPKSD has access control for abuse and/or DoS.

Can spammers get 100 email addresses? --- Yes. 1000? --- they could. Over 10000? --- I don't think so. If they try it with bunch of bots, it become a sort of DoS attack to my keyserver site. That is not the matter of spammer thing. THAT IS THE MATTER OF DoS ATTACK. So I have to protect my keyserver against them.

Today, OpenPKSD.org site is not synchronized to any keyserver site. OpenPKSD is able to sync to other OpenPKSD sites but there is no OpenPKSD site except my site ;-)

Anyway, your email address is safer than you thought

Done

Spamの温床？

- 古典的公開鍵サーバはどこかに登録すると世界中の公開鍵サーバに登録される
 - ちなみにopenpkd.orgは外部からは取り込んでいるが、他のサーバへは送っていない
 - 登録、情報参照、鍵の取り出しなどができる
- 公開鍵の内容を自動的にダンプしたページを作っているウェブサイトは少しありがた迷惑
- 公開鍵サーバの検索表示に対しGoogleアタックを防止していない公開鍵サーバサイトがある
 - Googleが出来るということはspam業者も同じことができるはず

かなりありがた迷惑

"Hironobu SUZUKI" site:www.cs.uu.nl - Google 検...

File Edit View Go Bookmarks Tools Help

http://www.google.co.jp/search?h

Google ウェブ イメージ ニュース グループ ディレクトリ more >

"Hironobu SUZUKI" site:www.cs.uu.nl Google 検索

検索オプション
表示設定

ウェブ全体から検索 日本語のページを検索

ウェブ "Hironobu SUZUKI" の検索結果のうち www.cs.uu.nl からの約 188 件中 1 - 10 件目 (0.27 秒)

[PGP trust paths : Hironobu SUZUKI -> Christoph Martin](#) - [[このページを訳す BETA](#)]
PGP trust paths : Hironobu SUZUKI -> Christoph Martin. from, stats Hironobu SUZUKI (Independent Software Consultant) <hironobu.at.h2np.net>. to, stats Christoph Martin <Christoph.Martin.at.Uni-Mainz.DE>. find, reverse path ...
www.cs.uu.nl/people/henkp/henkp/pgp/pathfinder/paths/02912C53/to/F95C2F6D.html - 16k - [キャッシュ](#) - [関連ページ](#)

[PGP trust paths : Hironobu SUZUKI -> Peter Nai Wan](#) - [[このページを訳す BETA](#)]
PGP trust paths : Hironobu SUZUKI -> Peter Nai Wan. from, stats Hironobu SUZUKI (Independent Software Consultant) <hironobu.at.h2np.net>. to, stats Peter Nai Wan <peter.wan.at.cc.gatech.edu>. find, reverse path ...
www.cs.uu.nl/people/henkp/henkp/pgp/pathfinder/paths/02912C53/to/09590CFD.html - 16k - [キャッシュ](#) - [関連ページ](#)

[PGP key statistics : Hironobu SUZUKI \(Independent Software ...](#) - [[このページを訳す BETA](#)]
PGP key statistics : Hironobu SUZUKI <hironobu.at.h2np.net>. msd. ↑ click to enlarge ↓. rank ... 02912C53 - Hironobu SUZUKI (Independent Software Consultant) <hironobu.at.h2np.net>. trust paths :. from to. from to. see also : ...
www.cs.uu.nl/people/henkp/henkp/pgp/pathfinder/stats/02912C53.html - 36k - 2005年10月22日 - [キャッシュ](#) - [関連ページ](#)

Done

公開鍵サーバ不要論

- 段々と公開鍵サーバには置かないユーザも増えてきている
 - でも勝手に他人の公開鍵を公開鍵サーバに置くことができる
- 公開鍵所有者が自分の鍵をコントロールできない
 - 最近ではウェブサーバ上に登録しておき、こちらを使うように指示するユーザも多い
- これはいかん

OpenPKSD Trusted Key Server

- 鍵ペアを持っている
 - 公開鍵を登録する
 - それに対応する署名鍵・秘匿鍵を持っている
 - それがあるならデータを秘匿・検証できる
- だったら公開鍵所有者のみが自分の公開鍵を操作できるようにしてしまえ

OpenPKSD-TKS Coming Soon

- 基本システムは既に完成
 - 平成16年度はIPAから予算獲得
- サーバサイト環境構築にはまだ時間がかかる
 - 運用はプログラムだけじゃないから、まだまだ準備が大変
 - その後、しばらく運用テスト
- OpenPKSD.ORGはTKSベースに移行
 - 新しいタイプの公開鍵サーバとして置き換え
 - 一般に公開するのは2007年4月を目標とする

まとめ

- 完全性に使われる技術の紹介
- 認証性とは何か
- 認証モデルの解説
- キーサインパーティーのお話
- 公開鍵サーバの紹介