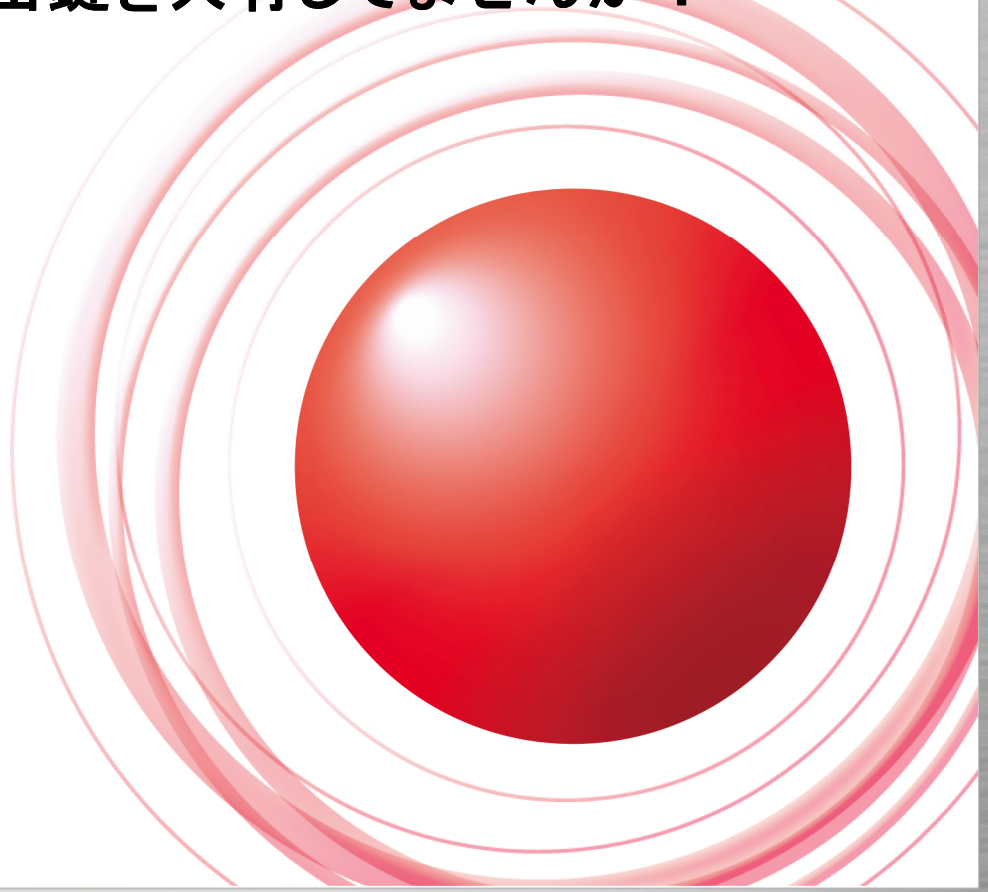
PKI Day 2012
December 13, 2012, ESSAM Green Hall, Tokyo, Japan

公開鍵の多くが意図せず他のサイトと秘密鍵を共有している問題

～いつのまにか他人と秘密鍵を共有してませんか？～



IIJ Internet Initiative Japan

Yuji SUGA
December 13th, 2012

Ongoing Innovation

今回のお話

IIJ Internet Initiative Japan

CONNECTED via IPv6 › お問い合わせ › サイトマップ

ホーム › サービス・ソリューション 企業情報

ホーム > 企業情報 > 研究・開発 > IIJの技術/セキュリティレポート > Internet Infrastructure Review(IIR) > Internet Infrastructure Review(IIR) Vol

Internet Infrastructure Review (IIR) Vol.17
2012年11月13日発行

RSS

今号では、2012年7月から9月までの3ヵ月間を対象として、セキュリティインシデントや迷惑メールなどの観測情報をまとめ、IIJが取り扱ったインシデントと対応について紹介しています。また仮想ネットワークをソフトウェアで自由に構成・制御する技術「SDN」について解説しています。今号のトピックは以下のとおりです。

› 一括ダウンロード[PDF:4.93MB]

› エクゼクティブサマリ[PDF:2.68MB]

› インフラストラクチャセキュリティ「スマートフォンのセキュリティ」[PDF:4.61MB]

今回は、SSL/TLS、SSHで利用されている公開鍵の多くが他のサイトと秘密鍵を共有している問題について解説すると共に、スマートフォンに関するセキュリティ事情と、標的型攻撃対策のための情報提供に関する議論について解説します。

概要

- インターネット上のIPv4アドレスを広範囲にスキャンして, SSL/TLSやSSHで利用されている公開鍵証明書, DSA署名及びPGP鍵を収集したところ, 意図せず他のサイトと秘密鍵を共有していることがLenstraらとHeningerらによる独立した2グループから報告された.

- この問題が発生した要因とその対策について理解して頂くことが本発表の目的.

参考文献

- [RwWr] Arjen K. Lenstra, James P. Hughes, Maxime Augier, Joppe W. Bos, Thorsten Kleinjung, Christophe Wachter, "Ron was wrong, Whit is right"
  - http://eprint.iacr.org/2012/064
- Arjen K. Lenstra, James P. Hughes, Maxime Augier, Joppe W. Bos, Thorsten Kleinjung, Christophe Wachter, "Public Keys“
  - http://www.iacr.org/conferences/crypto2012/abstracts/session11-2.html
- [PsQs] Nadia Heninger, Zakir Durumeric, Eric Wustrow, J. Alex Halderman, "Mining Your Ps and Qs: Detection of Widespread Weak Keys in Network Devices”
  - https://www.usenix.org/conference/usenixsecurity12/mining-your-ps-and-qs-detection-widespread-weak-keys-embedded-devices

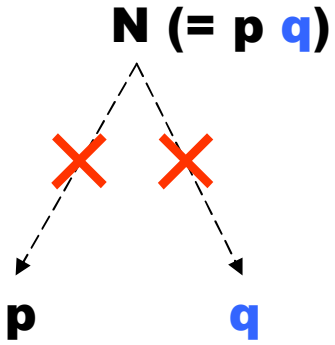
Agenda

- 同じ秘密鍵の利用を外部から同定される仕組みとその一般的な影響
  - RSA鍵生成, DSA署名生成
- Heninger [PsQs] らによる指摘
  - オンライン鍵チェックサービス
- Lenstra [RwWr] らによる指摘
- 対策と今後起こりうる予測

RSA

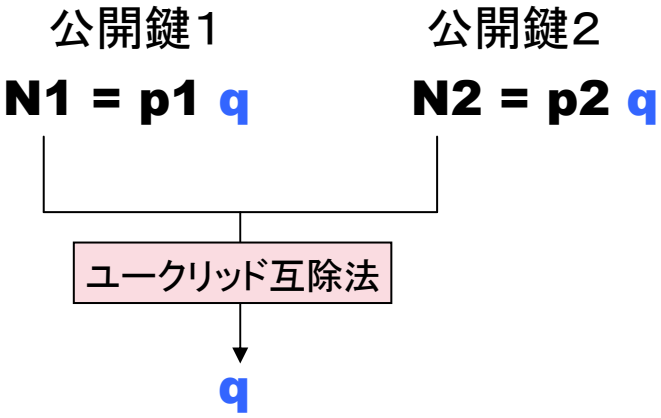
- 鍵生成
  - どでかい素数 p, q を2つ選択
  - N:= pq, e:= 65537(例えば)　←公開鍵
    - Nを素因数分解することは困難
    - eは$\phi(n)=(p-1)(q-1)$と素
  - $d:= e^{-1} \bmod (p-1)(q-1)$　←秘密鍵
    - p か q が分かると d も導出可能
- 暗号化　$c = m^e \bmod N$
- 復号　　$m = c^d \bmod N$

同じ秘密鍵であることが外部から同定される仕組み

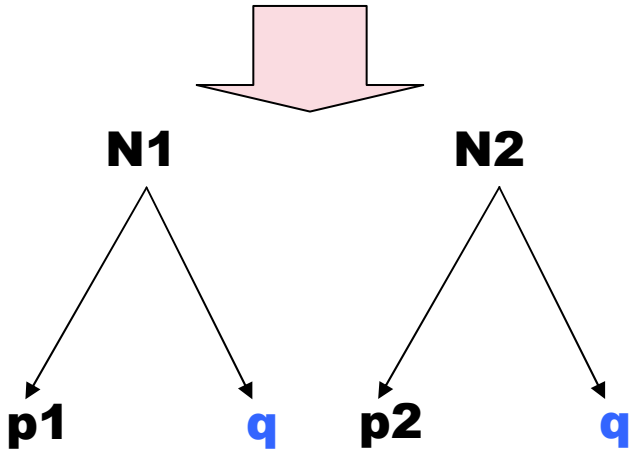


素因数分解は困難

（RSAはこの困難性に
安全の根拠を置いている）



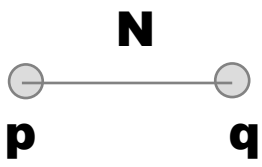
因数を見つけるのは容易

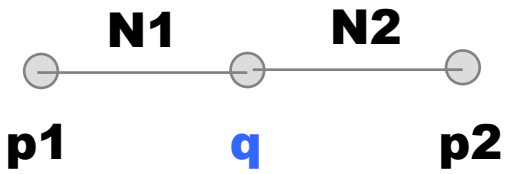


2つの公開鍵が素因数分解できてしまう

この状況をグラフ表現をすると…

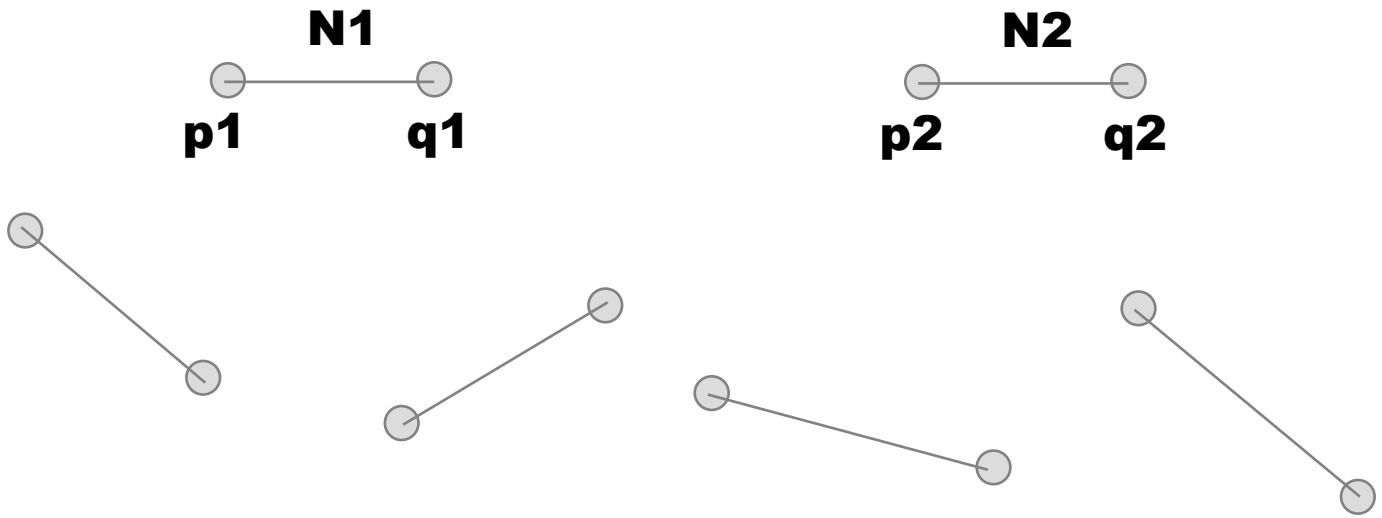
- [RwWr] で提案されたグラフ表現を用いると
  - 頂点：素数
  - 辺：2素数を結ぶ辺の存在＝公開鍵の存在





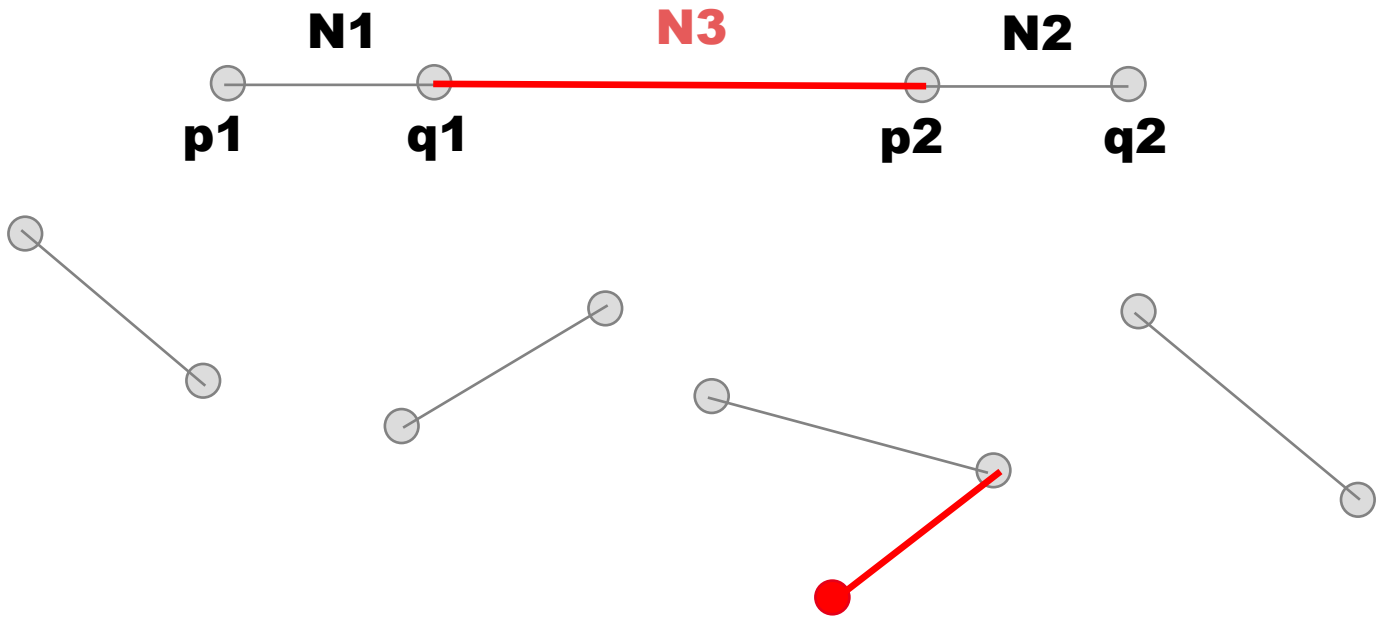
あるべき姿

- 2素数を選んで辺を結んだら（Nを計算したら）ほかのグラフと頂点を共有していない



しかし急にこういうことに…

- ほかのグラフと頂点を共有していなかったのに勝手に結ばれてしまうこともありうる



選んだ素数が偶然重なる可能性

- ランダムに選択したつもりの素数が
  ほかと偶然重なる可能性があることは
  RSAアルゴリズムの根本的な問題

- 素数定理
  - 素数がどのくらい存在するか知る指標

PRNGの問題

- 2048ビットRSAで用いられる1024ビット素数の候補は素数定理から約$2^{1014.53}$個も存在
  - 容易に重複するものではない
- しかし素数生成時のアルゴリズムに偏りがある＝$2^{1014.53}$個の全空間から素数を抽出していない場合にこの問題が起き得る

- 素数生成時に用いられる擬似乱数生成モジュールがまずいと問題が起きる

PRNGが問題である事例

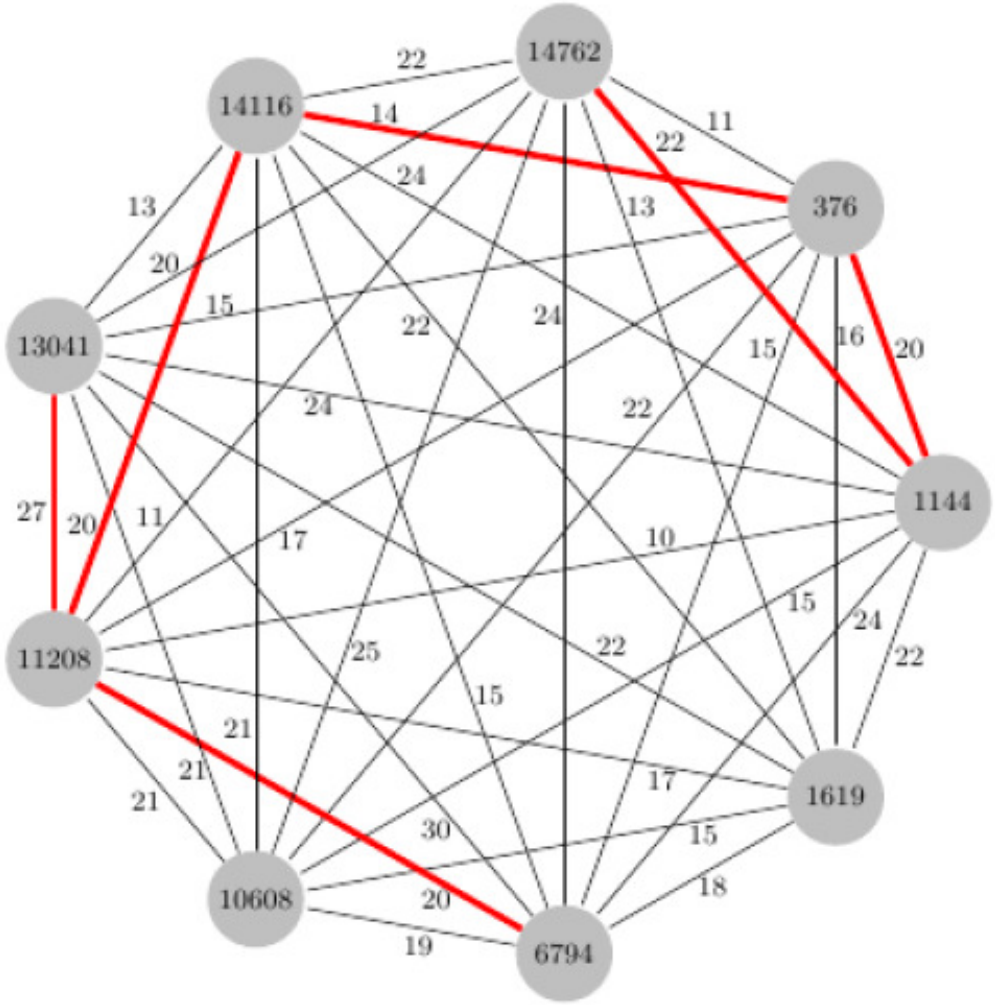
- Heninger[PsQs]らは たった9つの素数しか生成しないデバイスの存在を指摘
- 36の公開鍵は
  異なる9の秘密鍵の組み合わせで構成
  - ${}_9C_2 = 36$通り

DSAでも同じことが起こる

- 秘密鍵生成時ではなく署名生成時に発生

- 共通パラメータ (p, q, g, y)
- 秘密鍵 x, 公開鍵 $y = g^x \bmod p$
- 署名生成
  - 一時鍵(ephemeral key) k をランダムに選択
  - $r := (g^k \bmod p) \bmod q$
  - $s := k^{-1}(H(m)+xr) \bmod q$ (r,s):署名

Lenstra[RwWr] らも観測



[RwWr] Arjen K. Lenstra, James P. Hughes, Maxime Augier, Joppe W. Bos, Thorsten Kleinjung, Christophe Wachter, “Ron was wrong, Whit is right”, http://eprint.iacr.org/2012/064

DSAにおけるk選択時の問題

- もし k がばれると署名(r,s)から秘密鍵xが暴露
  - $x= r^{-1}(ks-H(m)) \bmod q$
    - cf) $s:= k^{-1}(H(m)+xr) \bmod q$

- 同じk を用いて2つの署名を生成すると…

  $r:= (g^k \bmod p) \bmod q$　　← これが共通

  $s1:= k^{-1}(H(m1)+xr) \bmod q$

  $s2:= k^{-1}(H(m2)+xr) \bmod q$

  → $k= (H(m1)-H(m2))(s1-s2)^{-1} \bmod q$

秘密鍵が漏洩したときの影響

影響の分類	詳細
(1)暗号化された通信の暴露	SSL/TLSやSSHサーバとクライアントの通信を傍受できる環境において, 暗号化を行っているにも関わらず通信内容を把握することができる
(2)サーバのなりすまし	DNS詐称やネットワークの乗っ取りが可能な環境において, SSL/TLSやSSHサーバになりすますことができる
(3)不正ログインもしくは クライアントのなりすまし	SSL/TLSでのクライアント認証のように公開鍵証明書を用いたログインが可能になる
(4)不正プログラムへの コード署名	コードサイニング用の公開鍵証明書の秘密鍵が漏洩することで, 当該証明書によって保証されたプログラムであるかのように見せることができる

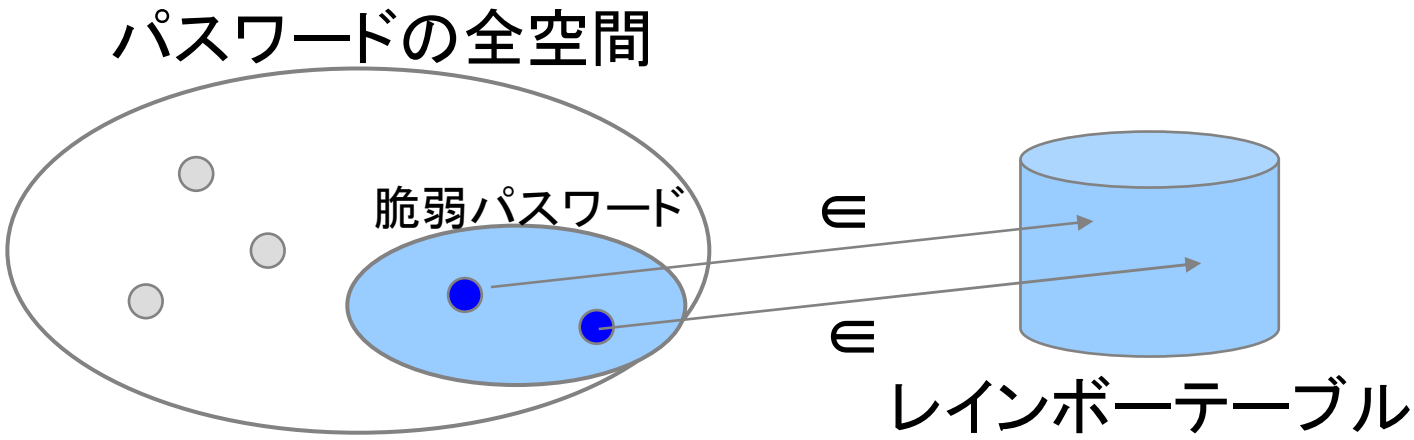
この問題を端的に表現したい

- 本発表のタイトルが長い…
  - **公開鍵の多くが意図せず**
    他のサイトと秘密鍵を共有している問題

- [PsQs] では Repeated keys と表現
- 「繰り返し」鍵？

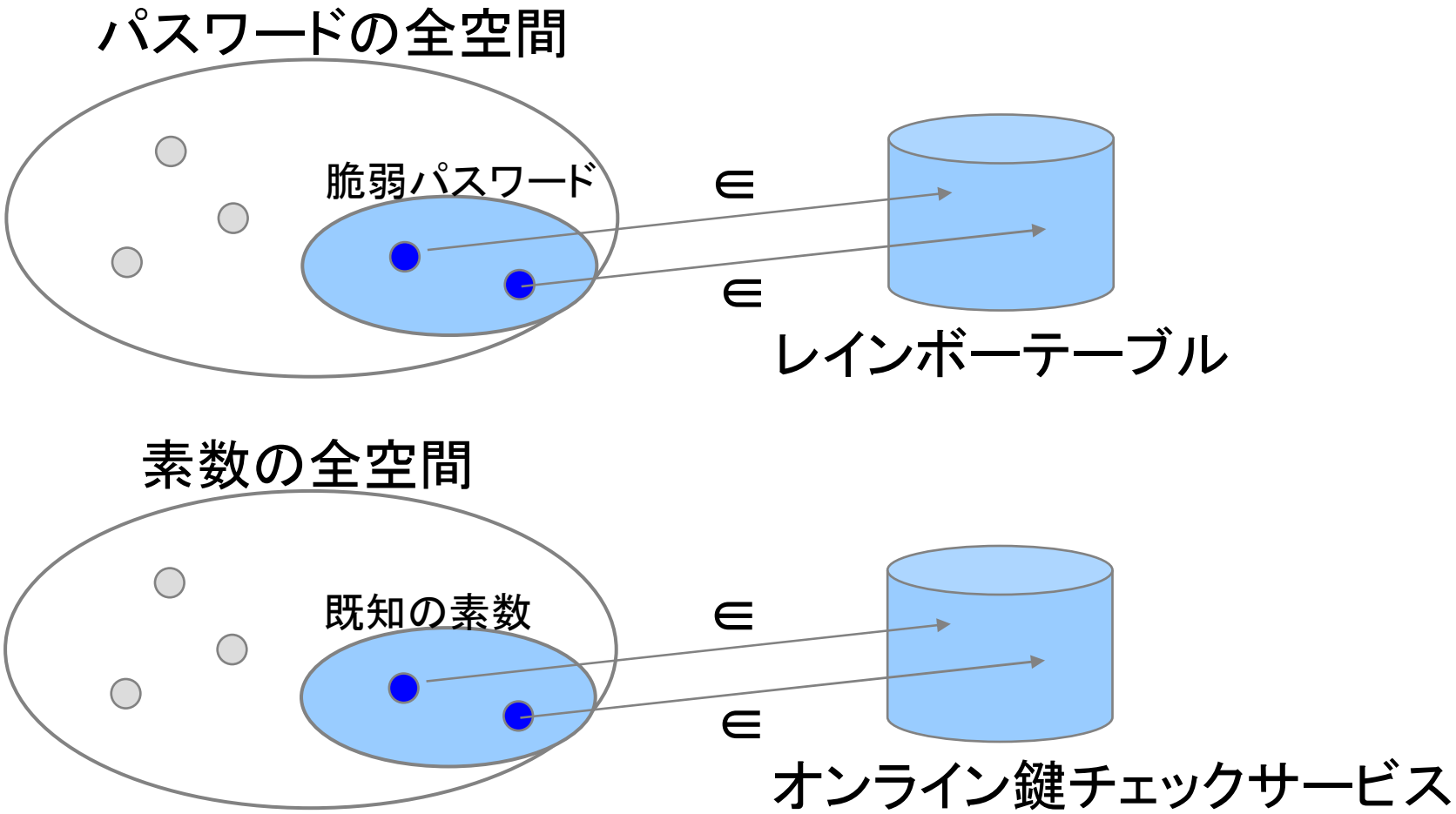
命名「公開鍵使い回し問題」

- 同じような話に「パスワード使い回し問題」



命名「公開鍵使い回し問題」

- 同じような話に「パスワード使い回し問題」



Heninger[PsQs]らによる指摘

- 1,280万のSSL/TLSサーバのうち61%が,
  1,020万のSSHサーバのうち65%が
- 他のいずれかのホストと同じ秘密鍵
  （repeated keys）を利用

一概に危ないとは言えない

- これらのすべてに問題があるわけではなく，意図的に複数IPアドレスで同じ鍵を利用しているケースもあるはず
  - 負荷分散のためDNSラウンドロビンなどの技術を用いて同じFQDNに複数のIPアドレスを割当

けれど…

- SSL/TLSでは5.57%（714,243IPアドレス），SSHでは9.60%（981,166IPアドレス）が
- 脆弱な repeated keys を利用

- 5.57% = 5.23% + 0.34%

5.23%

- 機器の出荷時のデフォルト鍵をSSL/TLSにおいて利用していると思われるケースが，少なくとも5.23%（670,391ホスト）も見つかっている
  - ほとんどがネットワーク機器，組み込み機器
  - 著者らは現在60のベンダーに連絡
  - そのうち20から何らかの回答あり
  - しかしアドバイザリを出したのは3ベンダーのみ

5.23%

- これらの機器以外にもApache WebサーバやCitrixリモートアクセスサーバでもデフォルト鍵を利用している事例が報告
  - そのうち38の証明書がWebブラウザで信頼
  - Fortune 500にリストされている企業, 保険会社, 法律事務所, 公共交通機関, 米国海軍など

0.34%

- 「擬似乱数生成モジュールの問題」に起因

- 鍵生成や署名生成時に利用する擬似乱数生成モジュールのエントロピーが不足しているために, 十分な鍵空間から鍵が生成されていないため, 同じ秘密鍵を共有してしまっているという主張

Debian OpenSSL

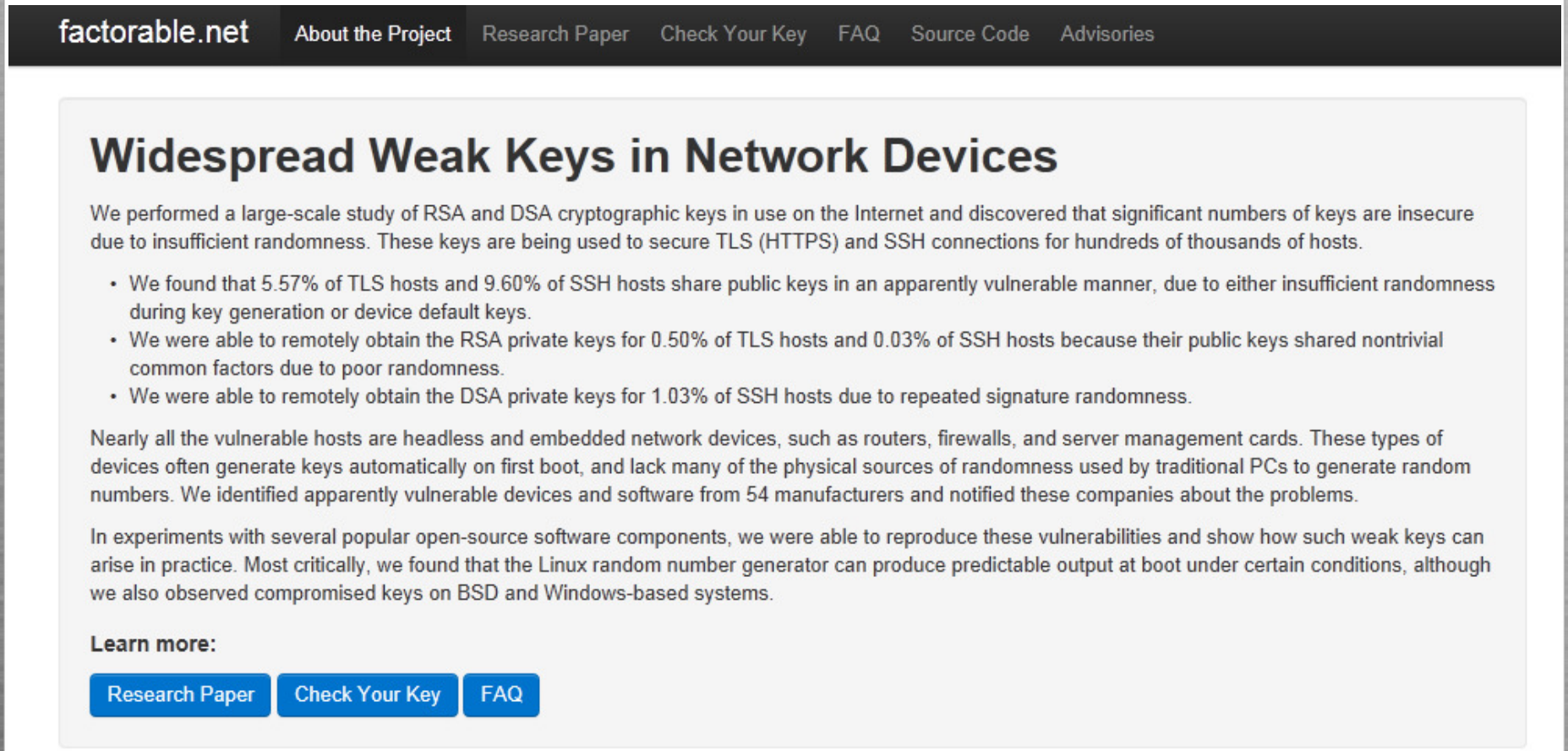
- 過去にも同じ問題に起因する事例が存在
- Debianの特定バージョンにおけるOpenSSLを使って鍵生成を行った場合，極端に少ない鍵空間からしか秘密鍵を導出していないという問題
- 2008年にアドバイザリが発行されていたにも関わらず，現在でもその脆弱な鍵を利用しているサイトがSSL/TLSでは0.03%（4,147ホスト），SSHでは0.52%（53,141ホスト）存在していることが報告

推奨事項

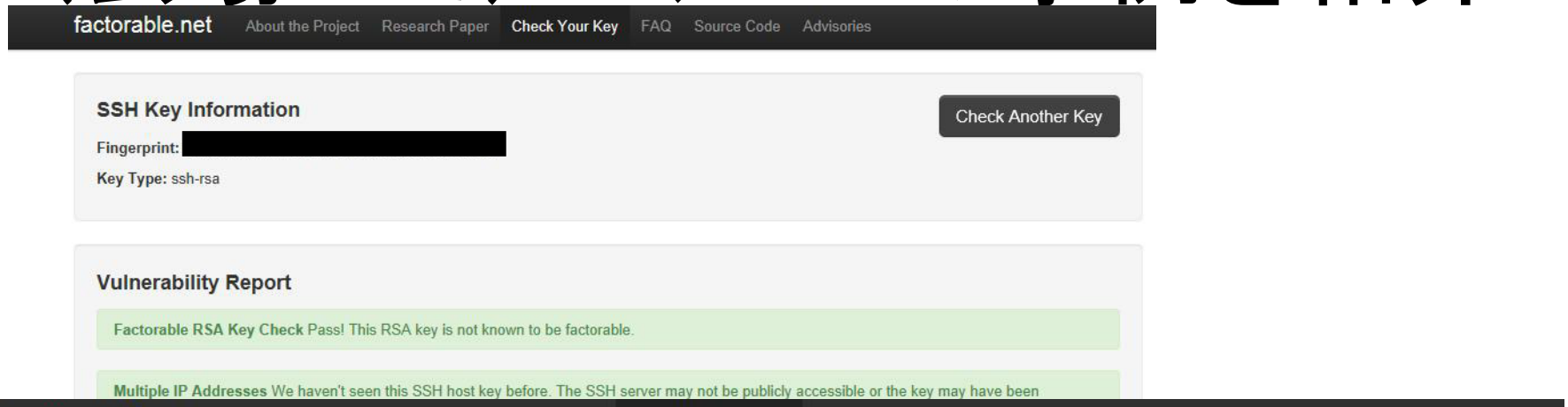
- デバイス製造社向け
  - ビルトインされたデフォルト鍵や証明書をユーザが利用できる状態にしない。
  - 十分なエントロピーを確保するためにハードウエアによる擬似乱数生成器を利用する。
- エンドユーザ向け
  - デバイスが出荷されたときに格納されているデフォルト鍵や最初にブートされた際に生成される鍵を利用せず，十分なエントロピーを確保できる他の環境下で生成した鍵を利用する。
  - 自ら生成した鍵が脆弱かどうか，つまり既に他のユーザに利用されている鍵かどうかをチェックする。
- CA向け
  - カスタマーが提示した公開鍵が脆弱かどうかチェックし，脆弱な場合には証明書を発行しない。

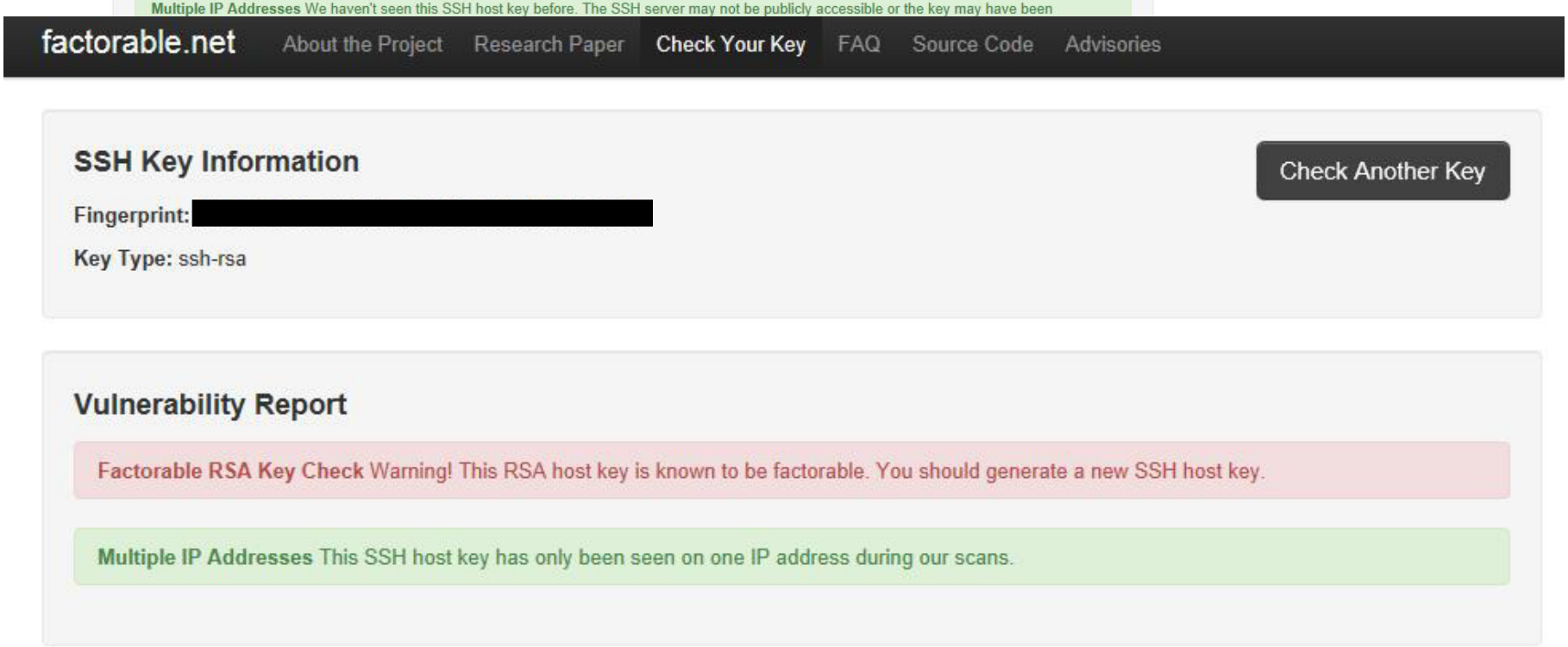
オンライン鍵チェックサーバ

- https://factorable.net/



（会場にて）いくつかの事例を紹介



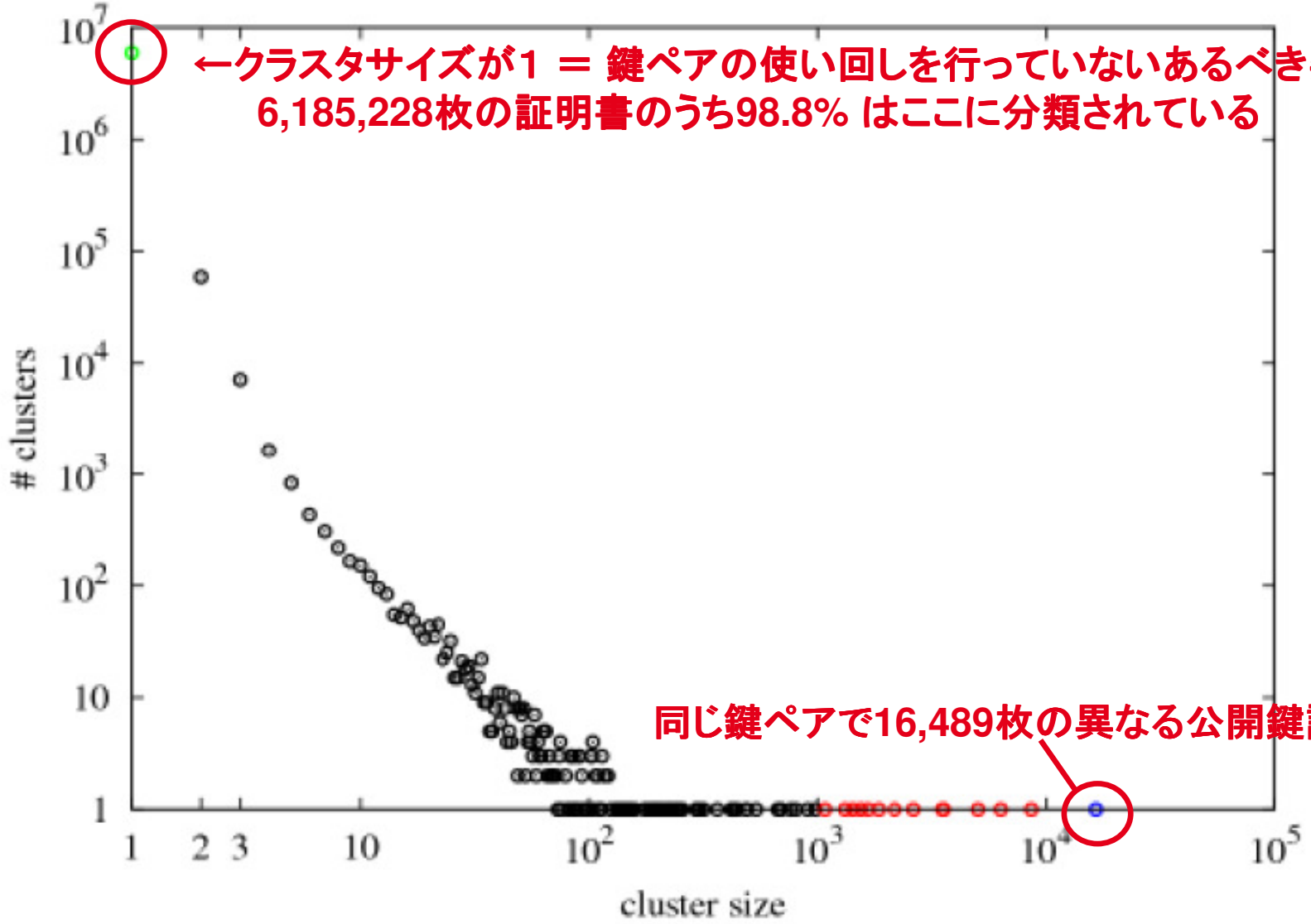


Lenstraらによる指摘

- The EFFSSL Observatoryなど, 公開されている複数の公開鍵証明書データベースから, 6,185,372の相異なるX.509証明書と, 5,481,332のPGP公開鍵を収集

RSA証明書のクラスタリング

- 6,185,228のRSA公開鍵を含むX.509証明書のうち, 266,729(4.3%)の証明書において他の証明書と同じRSA公開鍵を包含していることを指摘
- 同じ公開鍵を持つ証明書をクラスタリングして5,989,523のクラスタに分類



[RwWr] Arjen K. Lenstra, James P. Hughes, Maxime Augier, Joppe W. Bos, Thorsten Kleinjung, Christophe Wachter, “Ron was wrong, Whit is right”, http://eprint.iacr.org/2012/064

意図的に同じ鍵を利用しているケース？

- 証明書の期限切れのあと同じ鍵ペアで公開鍵証明書を新たに申請
  - 問題ないと認識 or 鍵を更新すべきと知らない？
- 同じ組織の異なるFQDNの証明書に同じ鍵ペアを利用している
  - 管理が楽だから？

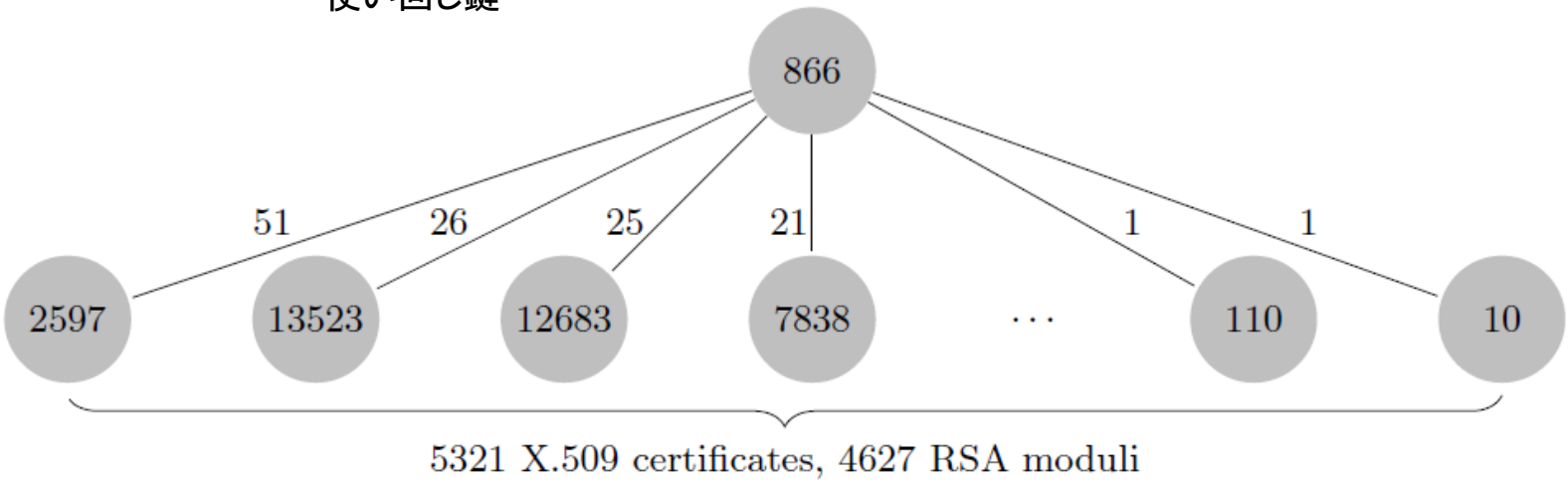
などのケースがあると考えられる（須賀私見）

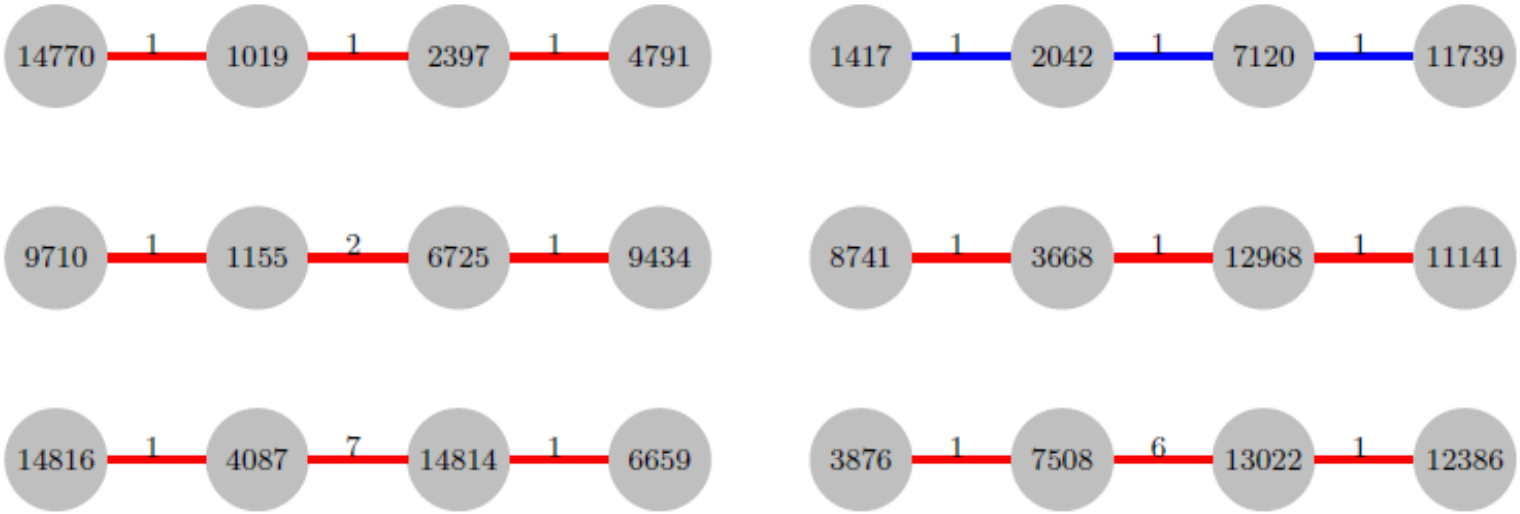
相異なる公開鍵の分類

- X.509証明書から得られた5,989,523の異なるRSA公開鍵と，同様にしてPGP公開鍵から得られたRSA公開鍵から合計で6,386,984の異なるRSA公開鍵を得たのち，同じ秘密鍵を共有してしまっているケースを調べたところ12,934の公開鍵で秘密鍵を共有していることが明らかに

脆弱な鍵を使っている事例

使い回し鍵





[RwWr] Arjen K. Lenstra, James P. Hughes, Maxime Augier, Joppe W. Bos, Thorsten Kleinjung, Christophe Wachter, “Ron was wrong, Whit is right”, http://eprint.iacr.org/2012/064

対策

- 十分なエントロピーを確保できる環境かどうか確認
  - とはいうものの… それを実現するのは難しい
- 鍵データベースを構築して，これまでに利用された「使い回し鍵」を使っていないか確認できる仕組み
  - factorable.net と同様のサービス
  - 事例：Debian OpenSSL鍵生成問題時に配布されたDB
- 鍵長を特殊なものにする
  - 例えば N が 2012 ビットになるような合成数など
    - この鍵を受け入れないデバイスもあるかもしれないので注意

技術的要因以外の問題

- 問題の本質は技術的なことだけではない
  - コスト負担の問題
    - お互いに押し付けあわないで
      それぞれの立ち位置でカバーしあうことが大切
  - まだまだ啓蒙活動が必要なのかもしれない
    - 一例）暗号学者と実装者との理解の乖離

Predictions

- アルゴリズムの横展開
  - 楕円暗号などほかのプリミティブでも起こる
- Low entropy を利用した攻撃
  - DSA署名生成時のように一時鍵・セッション鍵の鍵空間の狭さを利用した脆弱性の発見
  - CAや主要サイトの鍵生成時の環境を特定し鍵生成アルゴリズムをぶん回して鍵を特定

IIJ Internet Initiative Japan

インターネットの先にいます。

IIJはこれまで，日本のインターネットはどうあるべきかを考え，
つねに先駆者として，インターネットの可能性を切り拓いてきました。
インターネットの未来を想い，イノベーションに挑戦し続けることで，世界を塗り変えていく。
それは，これからも変わることのない姿勢です。
IIJの真ん中のIはイニシアティブ ―――――― IIJはいつもはじまりであり，未来です。

Ongoing Innovation

お問い合わせ先　IIJインフォメーションセンター
TEL：03-5205-4466（9：30～17：30 土/日/祝日除く）
info@iij.ad.jp
http://www.iij.ad.jp/