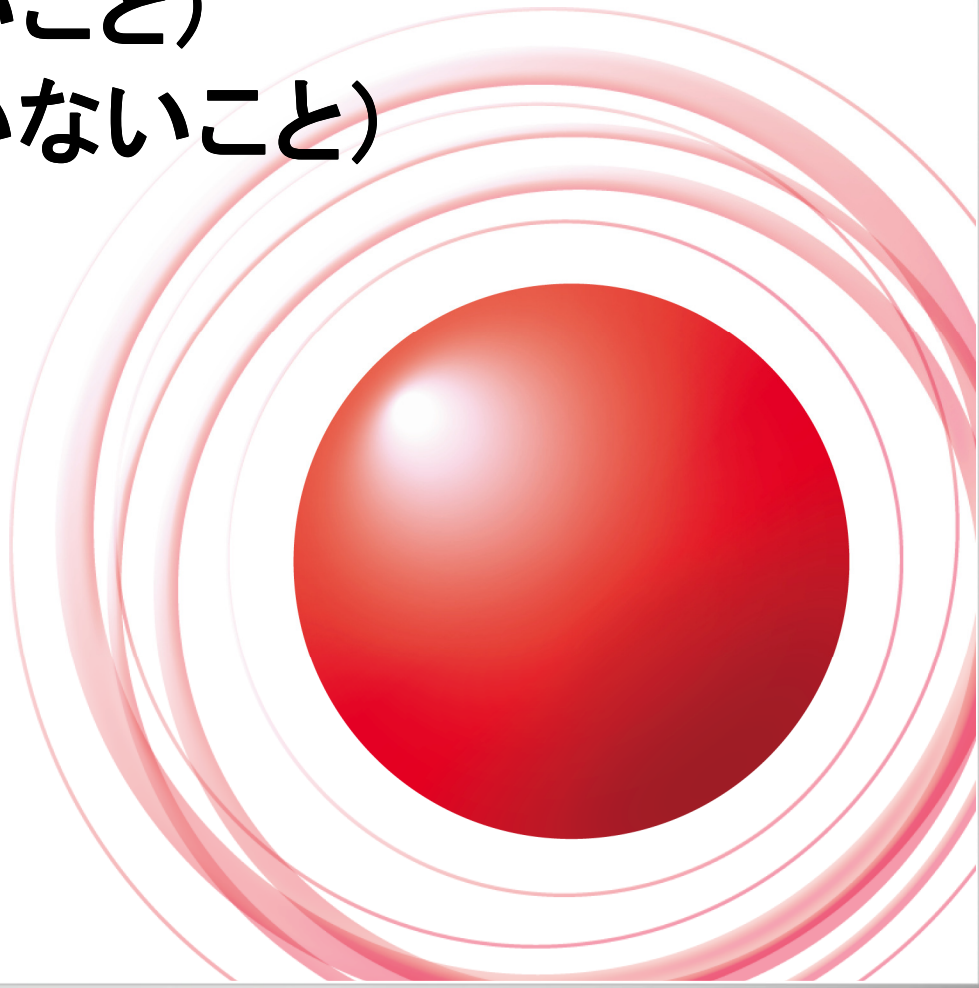
CRYPTREC Symposium 2015, March 20th, 2015, Tokyo, Japan

ISPから見た暗号技術（仮題）

株式会社インターネットイニシアティブ
セキュリティ情報統括室
須賀祐治
2015-03-20

CRYPTREC Symposium 2015, March 20th, 2015, Tokyo, Japan

ISPから見た暗号技術
（に期待したいこと）
（に期待していないこと）

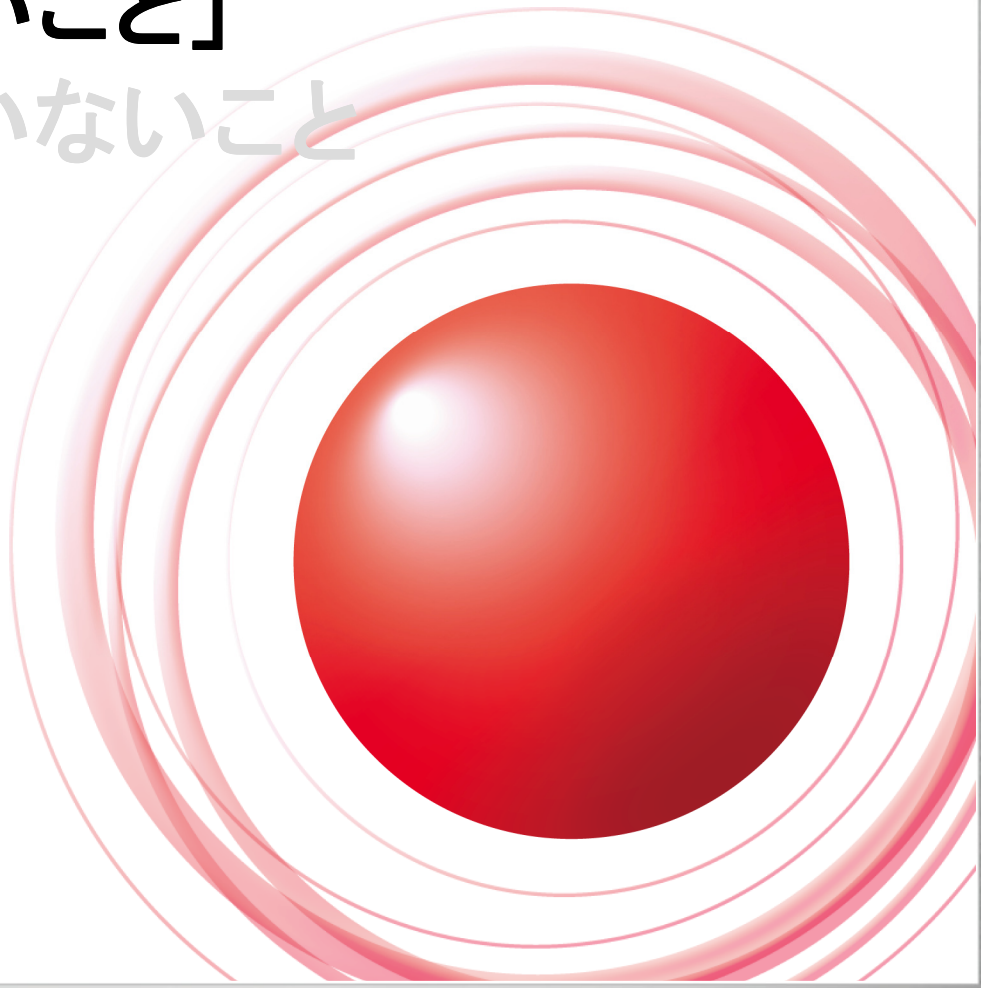


株式会社インターネットイニシアティブ
セキュリティ情報統括室
須賀祐治
2015-03-20

CRYPTREC Symposium 2015, March 20th, 2015, Tokyo, Japan

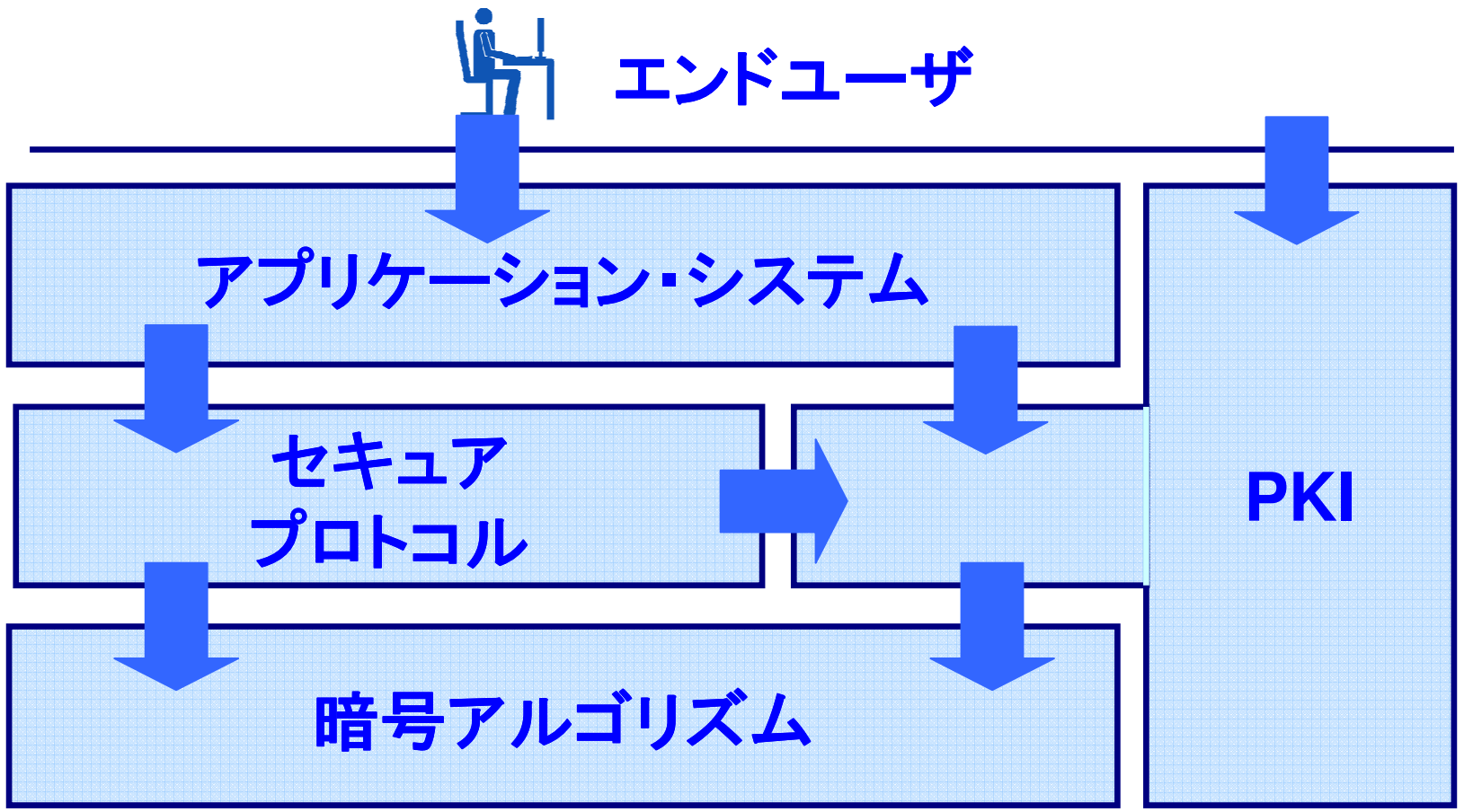
ISPから見た「暗号技術に期待したいこと」

に期待していないこと



株式会社インターネットイニシアティブ
セキュリティ情報統括室
須賀祐治
2015-03-20

ISPが暗号技術を使う場面って？



ISPからCRYPTRECの活動に期待したいこと

- どの暗号アルゴリズムを使ってよいか？
  - リスト：1軍/2軍/戦力外通知
- 暗号を使う際に気をつける点はないか？
  - 各種ガイドラインで補足されているけれど…
  - RSAでの素数被り，（EC）DSAでのパラメータ被り

- もっと端的に言えば…
  - こういう製品・サービスを利用していればOKというお墨付き
  - かつ… こういう設定をしていればOK
  - どのようなサイクルで運用すればよいかBest Practice

期待したいこと(続)

- その暗号がヤバくなったときに教えて欲しい
  - 即時性のあるものか？
  - どのレイヤがヤバいのか？
  - どう塞ぐのか？
- OpenSSLよりいいライブラリを教えて欲しい
- 暗号そのものだけではなく，その周辺で起きていることも教えて欲しい
  - (同上：即時性/レイヤ/塞ぎ方)
  - そしてこっちの方が圧倒的に事例が多く，ときにはアカデミアからのインプットで分かりにくい

穴を塞げと言われても

- どのように穴を塞げる？それは簡単ですか？
  - 設定を変更するだけでよい（ないといけない）
  - Patchをあてるだけでよい（ないといけない）
  - ライブラリを差し替えればよい（ないといけない）
  - それを捨てて代替物を買い直さないといけない
- できれば稼動しているものを止めずに対応したい
  - 可用性：24hours 365days

脆弱性を紐解くフェーズ1

- 「なに」に対する脆弱性・問題なのか？
  - 製品・サービス or 仕様・フォーマットそのもの
- サーバサイドの影響
  - バックエンドシステムの対処
- クライアントサイドの影響
  - 顧客からの問い合わせへの対応

脆弱性を紐解くフェーズ2

- リスクの見積もり
- スケジュールの見積もり

- 特殊環境での利用かどうか
- S/C片方だけの対処でうまくいくケースか

「塞ぐ」意味を理解しているか？

- その脆弱性を咀嚼して行動に移せるのは理想だけど，管理者・オペレータにそこまでの知識・力量はあるか？（人材育成の問題）

- 有識者曰く「これさえやればOK」を信じる？
  - HeartBleed バグに起因する証明書再発行
- ベンダーからの情報を信じる？（信じられる？）
  - Superfish問題：アンインストールだけでOK？

評価機関としてのCRYPTREC

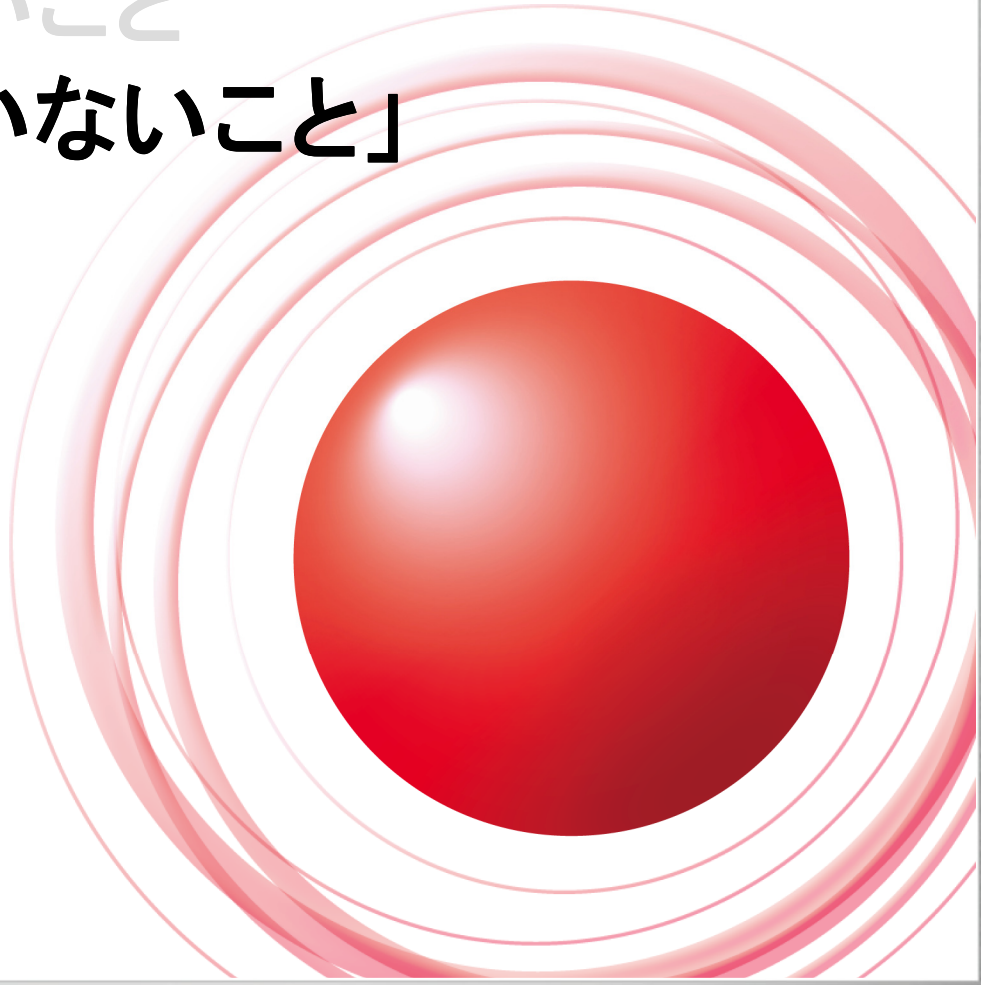
- 暗号アルゴリズムの評価だけに留まる必要はありますか？
  - （他のプリミティブへの拡張も含めて）
- 暗号技術を使うプロトコル・フォーマット仕様
  - 実装の問題だけではなく，仕様そのものの問題をどう考えていくか
- パーソナルデータ利活用時には
  - 「匿名化」技術のお墨付き制度ができるのでは？

そのほかISPから見て解せないこと

- SSL/TLSでCTRモードが使えない
  - CBCの一連の問題も（もし使えたならば）CTRに移行する対策が楽ではなかったか？
    - 2009年に起きたSSHの問題は対策が容易だった
- SHA-2移行終わっていないのにSHA-3要る？

CRYPTREC Symposium 2015, March 20th, 2015, Tokyo, Japan

ISPから見た「暗号技術
に期待したいこと
に期待していないこと」



株式会社インターネットイニシアティブ
セキュリティ情報統括室
須賀祐治
2015-03-20

ひとつの真理

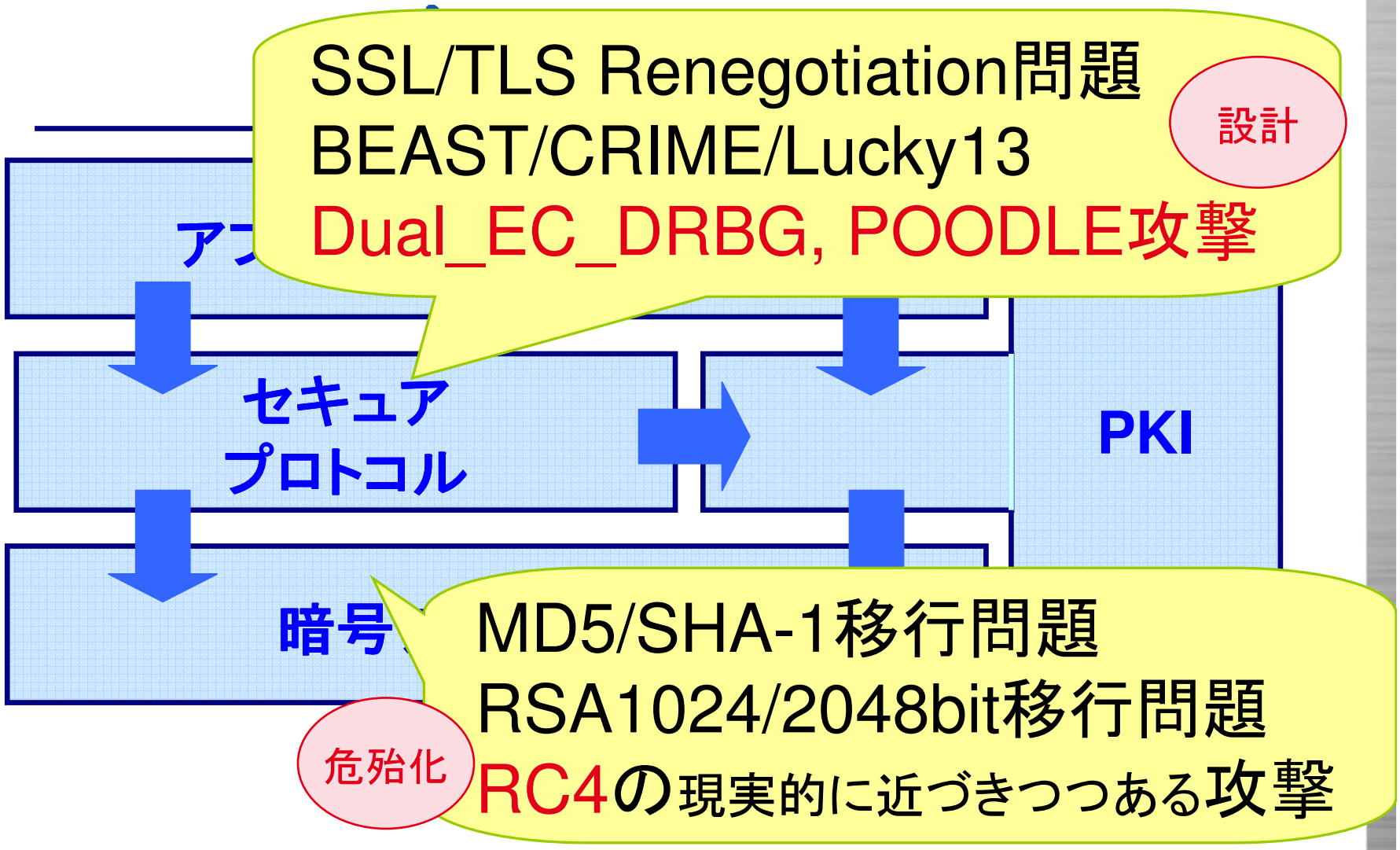
ひとつの真理

暗号を解くよりも
他をハック
する方が楽

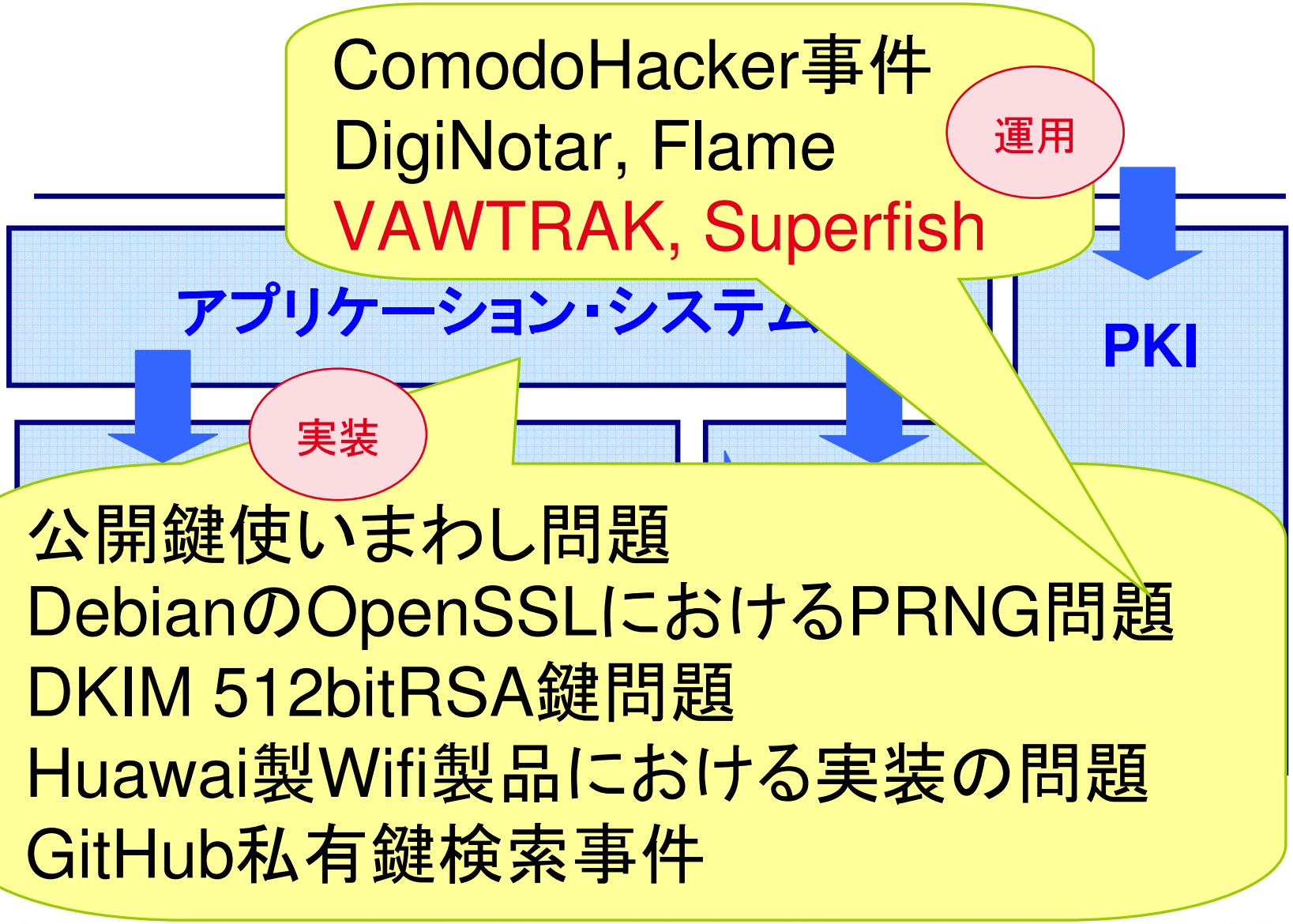
攻撃者の心理

エンドユーザと「暗号技術」の距離

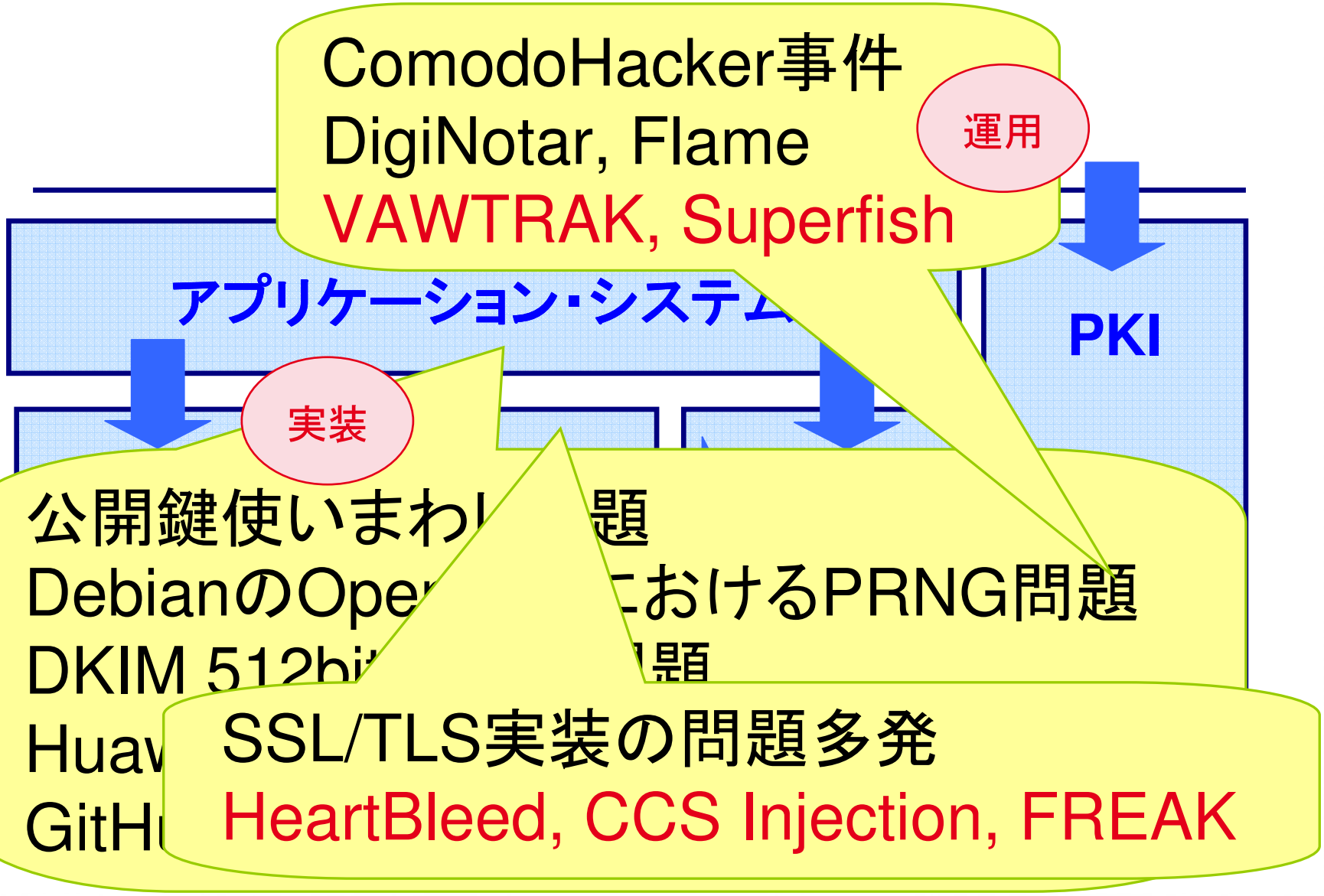
赤字はここ2年で顕在化した問題



エンドユーザと「暗号技術」の距離



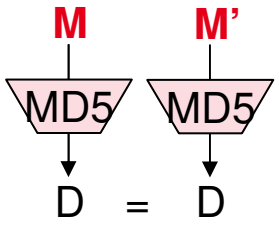
エンドユーザと「暗号技術」の距離



危殆化

暗号危殆化による間接的な影響

- APOP, SIP, HTTP Authentication における パスワードリカバリ攻撃
  - Yu Sasaki, Lei Wang, Kazuo Ohta and Noboru Kunihiro, "Extended Password Recovery Attacks against APOP, SIP, and Digest Authentication," IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, Vol.E92-A, No. 1, pp. 96-104, 2009.
- X.509中間CA証明書偽造攻撃
  - MD5 considered harmful today
    - http://www.win.tue.nl/hashclash/rogue-ca/
- ともにMD5コリジョン攻撃を利用



設計

2011年はCBCモードの当たり年

- 9月：BEAST攻撃（CVE-2011-3389）
  - SSL 3.0/TLS 1.0 を使用しているブラウザの CBC モードに対して選択平文攻撃を行うことでブラウザ内の Cookie を入手するツールを公開
  - ブロックごとではなく**バイトごとの全数検索**だとうまくいく例を示し，実際にPayPalからのセキュアなCookieを奪取してログイン権限を不正に得るというデモを公開
- 10月：XML暗号化仕様
  - Webサービスの実装物をplaintext validity oracle として利用
  - XML Parser のエラーの意味を解釈しながらトライ＆エラー
- 12月：TLS1.2における Truncated HMAC利用時の問題
  - RFC6066で規定された拡張機能のひとつであるTruncated HMACを用いたTLS1.2通信における脆弱性が公開
  - 通常のHMACではなく、80ビットに切り詰めたデータをMAC(データの完全性を保証する認証子)として利用する拡張方式の原理的な問題

設計

CRIME攻撃（2012年9月）

- SSL/TLSでCompression(圧縮)機能を有効にしているケースでCookie を搾取するデモが公開
- 例え同じ長さのデータを圧縮したとしても，圧縮前に同じ文字を含むかどうかで辞書の長さが変わるという事実を用いてトライ＆エラーで暗号化データを復元する

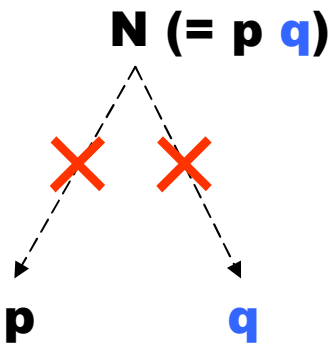
Lucky13攻撃（2013年2月）

- SSL/TLSへのタイミング攻撃．演算速度の違いから情報を搾取するサイドチャネル攻撃の１種をネットを介して行う手法
- CBCモードを使わない，もしくはMACとしてHMAC-SHA1などではなくAEAD(暗号化と認証子付与を同時に行う方式)を用いる．例えば GCMモードやCCMモードなど．

実装

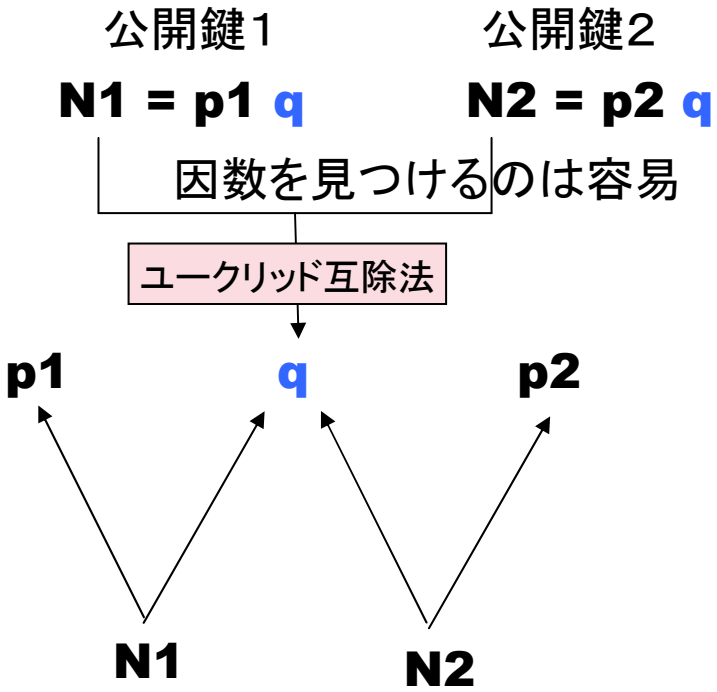
公開鍵使いまわし問題

- SSL/TLSやSSHで利用されている公開鍵証明書を収集
  → 意図せず他のサイトと秘密鍵を共有している事例
  - 機器の出荷時の鍵を利用: 5.23%（670,391ホスト）
  - 十分な鍵空間から鍵生成せず同じ秘密鍵を共有 : 0.34%
- RSAにて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）



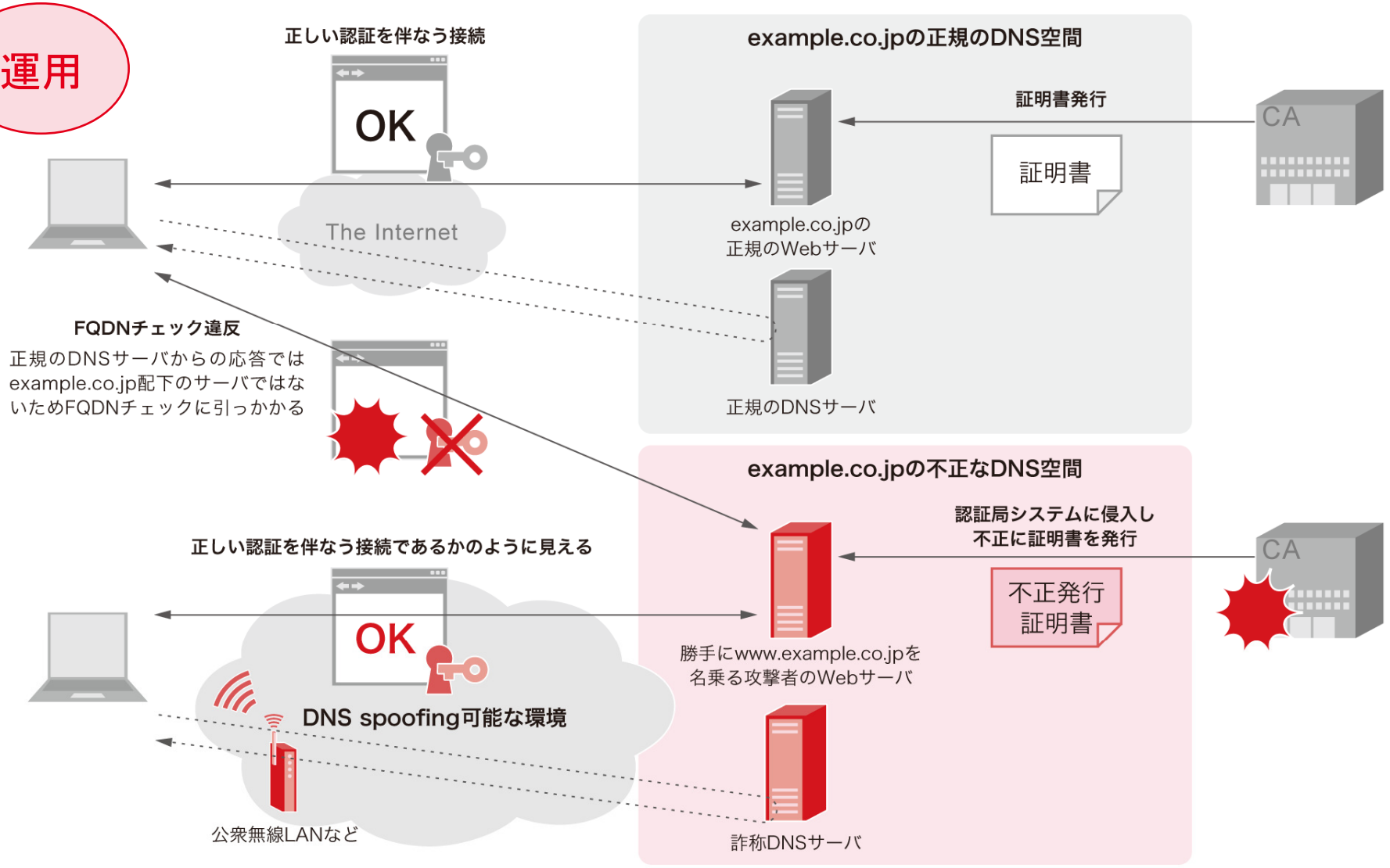
運用

PKIへの一連の攻撃

- ComodoHacker事件
  - 2011年3月 Comodo社の委託登録局(RA)のアカウントハッキングによる証明書の不正発行
  - Gmailなど著名なドメインに対するMITM攻撃
- DigiNotar認証局事件
  - 2011年8月　DigiNotar社自体への不正侵入による大量の不正な証明書発行
  - 本事件の影響により同社は翌9月に倒産
- Flame事件
  - 2012年5月 Microsoftの認証局に対するMD5選択平文攻撃による証明書の偽造
  - 未知の暗号解析手法が用いられたとの意見も
- その後も TRUKTRUSTなどPKI信頼失墜の事例が...

不正な証明書発行による影響（IIR13 1.4.3より）

運用



認証局システムに侵入され、www.example.co.jpについて不正証明書が発行されたとしても、攻撃対象ドメインの配下にサーバを実際に設置するか、何らかのDNSの不正操作と併用しないと攻撃は成立しない。この図はDNS spoofingによりFQDNチェックを迂回する例。

結局ルート証明書ストアを やられている

- 2014年に観測された VAWTRAK
  - オンラインバンキングへの中間者攻撃
  - 何しようが，もうあかんという状況

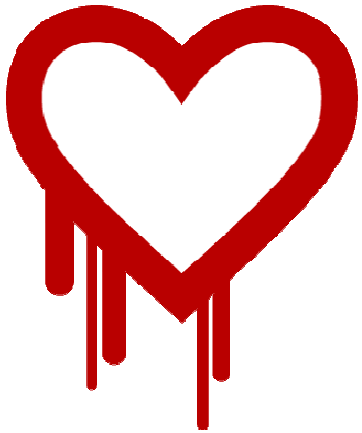
- Superfish事件
  - ○○系ベンダーだから… という政治的な話ではなくて，その製品のやっていることを技術的な観点で紐解くとやっぱ
    「ダメなもんはダメ」じゃないでしょうか？

暗号技術に関わるここ2年の話題

- 2013年 3月/8月 RC4における複数の攻撃
- 2013年 6月 NSAによる諜報活動の報道
- 2013年 9月 Dual_EC_DRBG 問題
- 2013年11月 IETF-88 にてPervasive Surveillance（広域監視）がメイントピックに
- 2014年 4月 OpenSSLにHeartbleed 発覚
- 2014年 6月 OpenSSLにCCS Injection 発覚
- 2014年 9月 Mozilla NSSに署名検証不備の脆弱性
- 2014年10月 POODLE attack SSLv3仕様そのものの問題
- 2015年 3月 FREAK攻撃

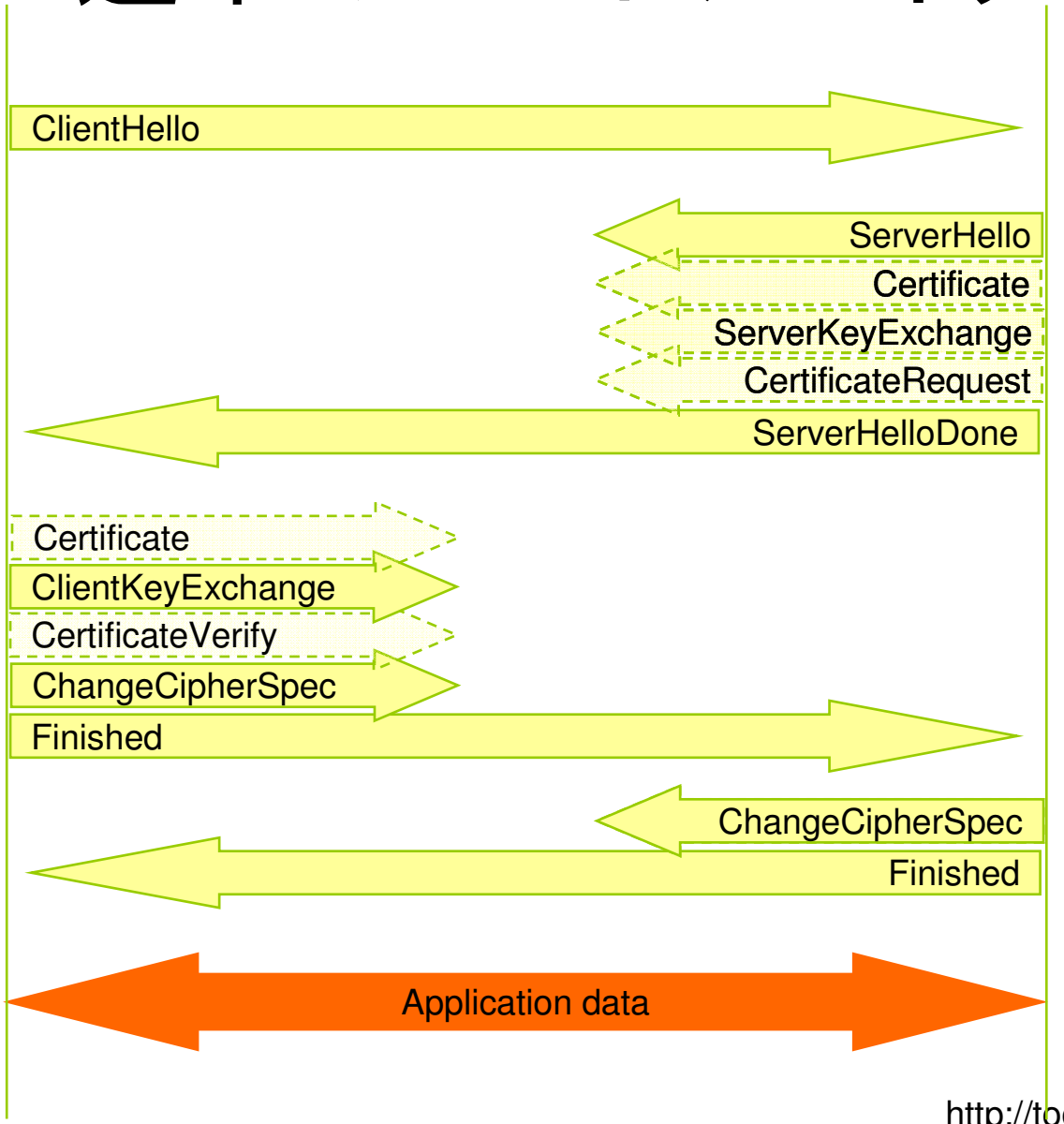
Heartbleed Bug 概要

2014年4月



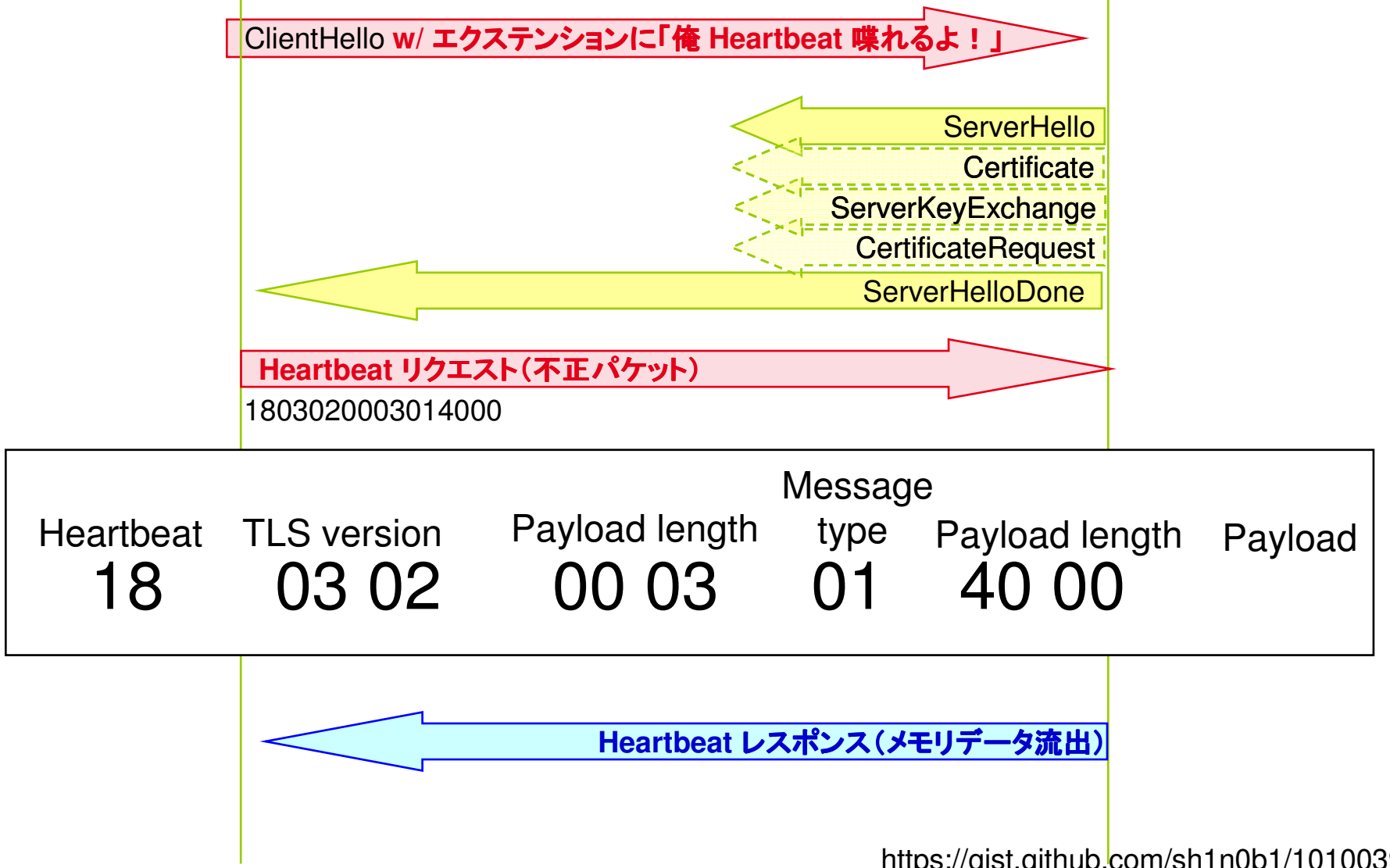
- CVE-2014-0160
- 日本時間4月8日未明に公開
  - http://heartbleed.com/
- 脆弱なバージョン：OpenSSL
  1.0.1 から 1.0.1f および 1.0.2beta1
- 問題：OpenSSL が動作しているマシンの
  メモリ情報を取得可能な状態にあった
- 対策：(1) 1.0.1g にアップデート or
  (2) Heartbeat 無効にして再コンパイル

通常のハンドシェイク



http://tools.ietf.org/html/rfc5246

ssltest.py (攻撃ツール)



https://gist.github.com/sh1n0b1/10100394

メモリへの不正アクセスの深刻度

- システム上のメモリ領域データ奪取により

ID/パスワード	成りすましによる不正アクセス
セッションID (例：Cookie)	セッションハイジャック
サーバ証明書の秘密鍵	暗号化通信の復元
	本物と判断可能な偽サーバへの誘導

が可能になっていた

- 秘密鍵の更新と証明書の再発行が推奨された

CCS Injection 概要

2014年6月

- CVE-2014-0224
- 日本時間6月6日に公開
  - http://ccsinjection.lepidum.co.jp/ja.html
- 脆弱なバージョン:
  - サーバ OpenSSL 1.0.1 系列:1.0.1g以下
  - クライアント

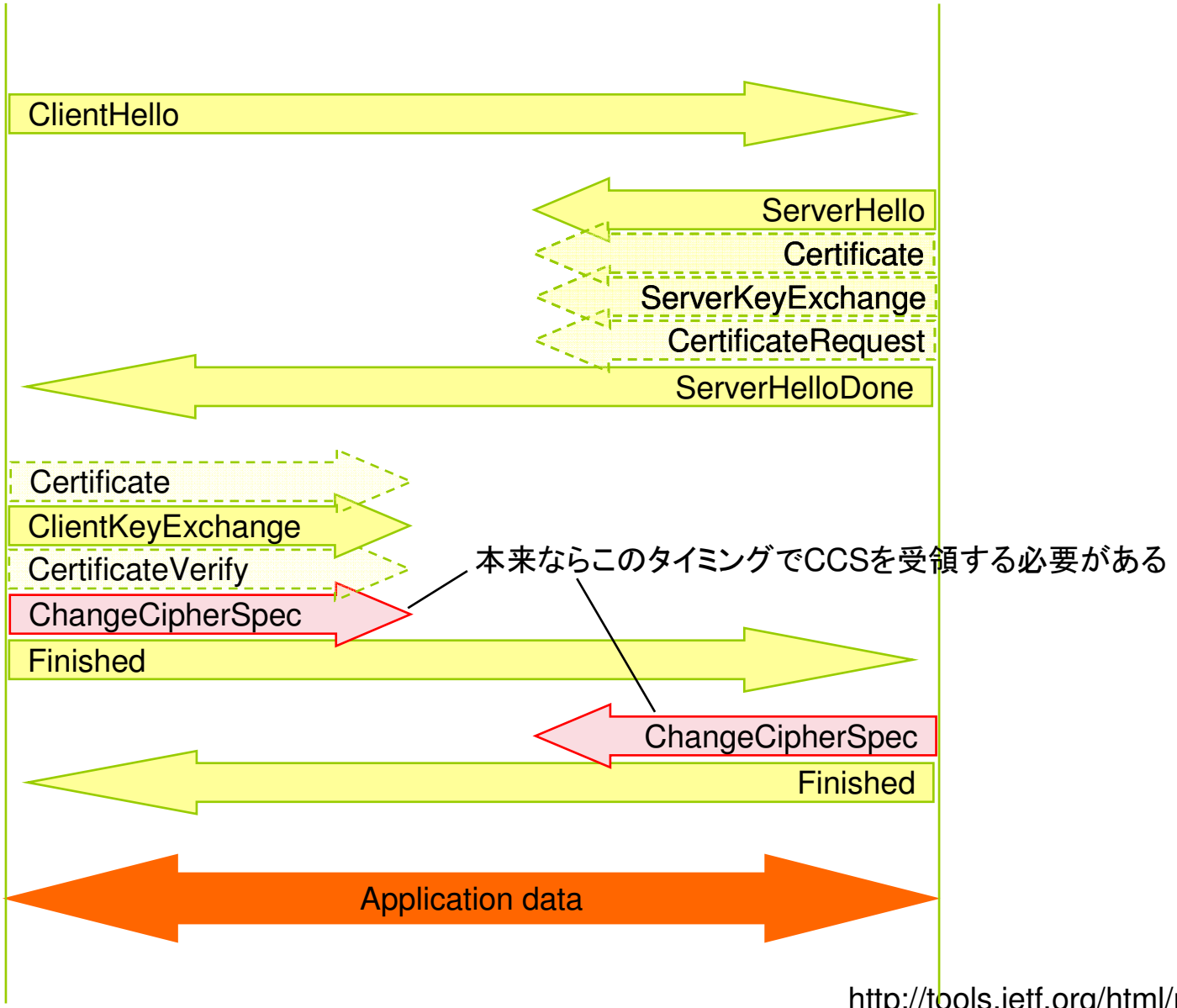
    各系列 1.0.1g以下, 1.0.0以下, 0.9.8y 以下
- 問題:ChangeCipherSpecメッセージの処理の欠陥により暗号化データの漏洩

ポイント

- 受け入れてはいけないタイミングでCCSメッセージを受領してしまう
  - CCS＝「セッション鍵をリフレッシュしようぜ！」
- サーバだけでなくクライアントもOpenSSLを利用している際にだけ起こる

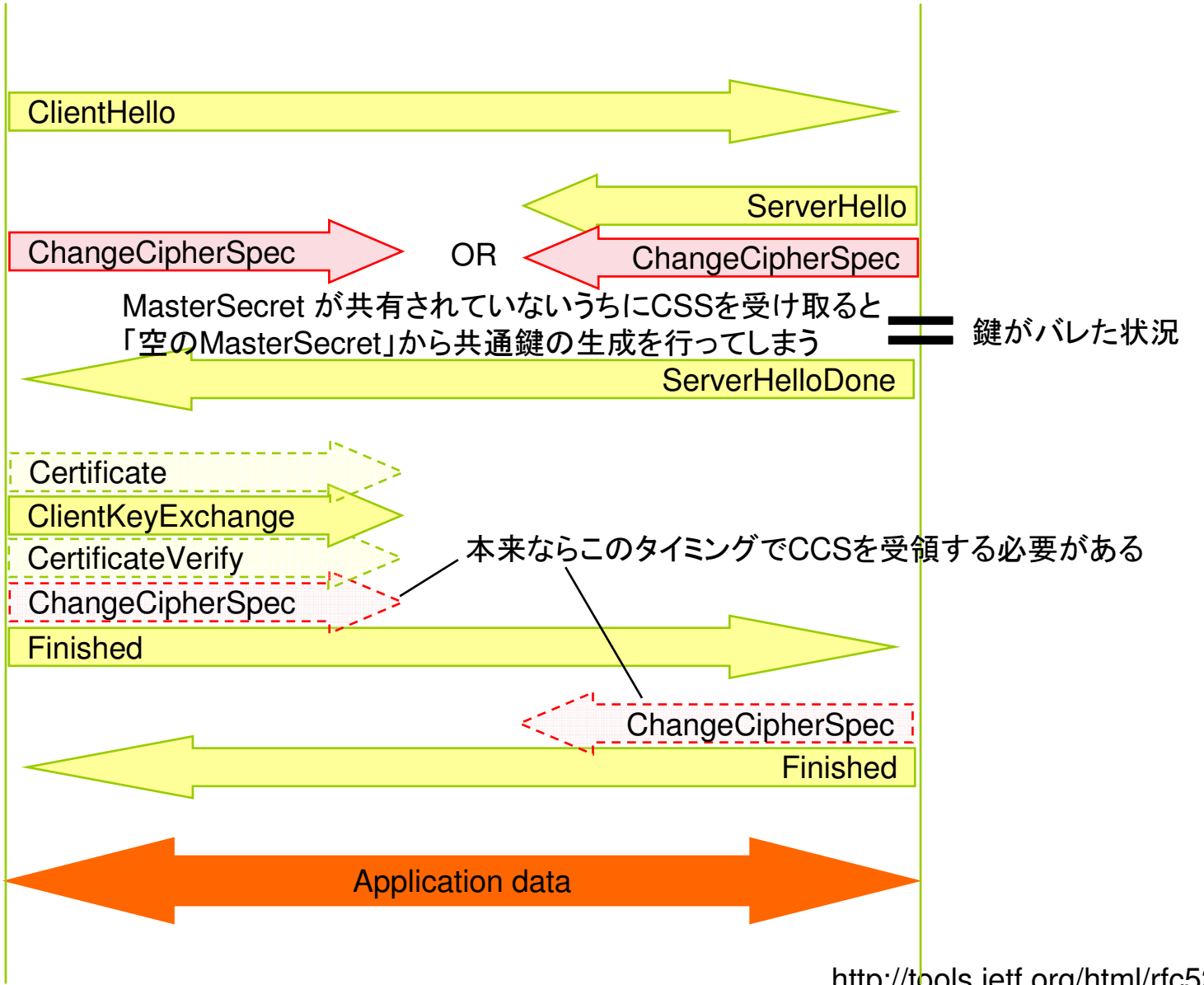
  → 環境によっては放置しても大丈夫な状況もありえる
- 枯れた技術神話が崩壊した
  - 「0.98系は大丈夫だろう」 v.s 意識の高い管理者（パッチ信者）
  - 1998年12月からエンバグしていた

SSL/TLS handshake



http://tools.ietf.org/html/rfc5246

SSL/TLS handshake



http://tools.ietf.org/html/rfc5246

2014年9月

NSSにおける署名検証不備

- CVE-2014-1568
- 日本時間9月24日に公開
  - http://www.mozilla-japan.org/security/announce/2014/mfsa2014-73.html

- 問題：NSSにおけるパーサの問題によりRSA署名検証をすり抜けて，偽造された文書が正当な署名であると返却される場合がある

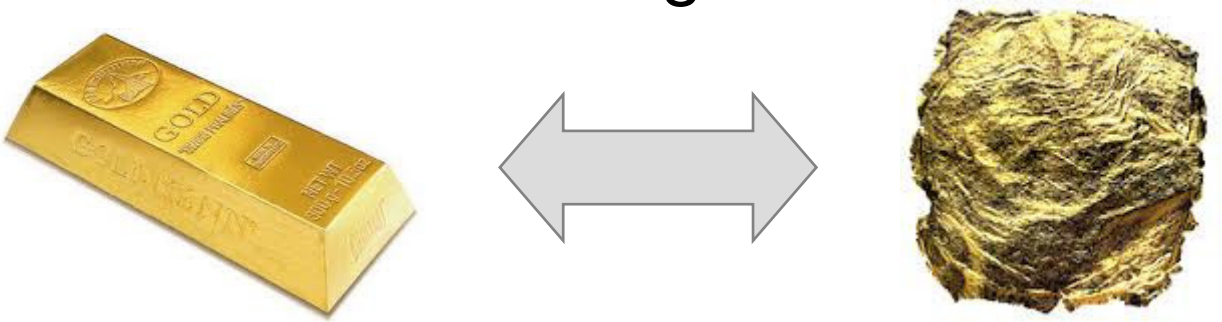
malleability

- Bitcoin 交換所Mt.Goxへの攻撃でも利用

http://arxiv.org/abs/1403.6676

  - Christian Decker, Roger Wattenhofer, “Bitcoin Transaction Malleability and MtGox”

- セマンティカリに同じ内容のデータをエンコーディングしても複数のデジタルデータで表現可能になるという「揺れ」を利用
  - BER v.s DER encoding



POODLE attack 概要

2014年10月

- CVE-2014-3566
- 日本時間10月15日に公開
  - https://www.openssl.org/~bodo/ssl-poodle.pdf
- 仕様そのものの問題
  - SSLv3にてCBC暗号モード利用時のみ影響
    - SSLv2は以前から脆弱
- 問題：Padding Oracle Attack の一種.
  サーバのパディングチェック機能を悪用し
  ブラウザから大量のリクエストをサーバに
  送りつけてトライ＆エラーを繰り返し，暗号化
  された攻撃対象データを1バイトずつ復号

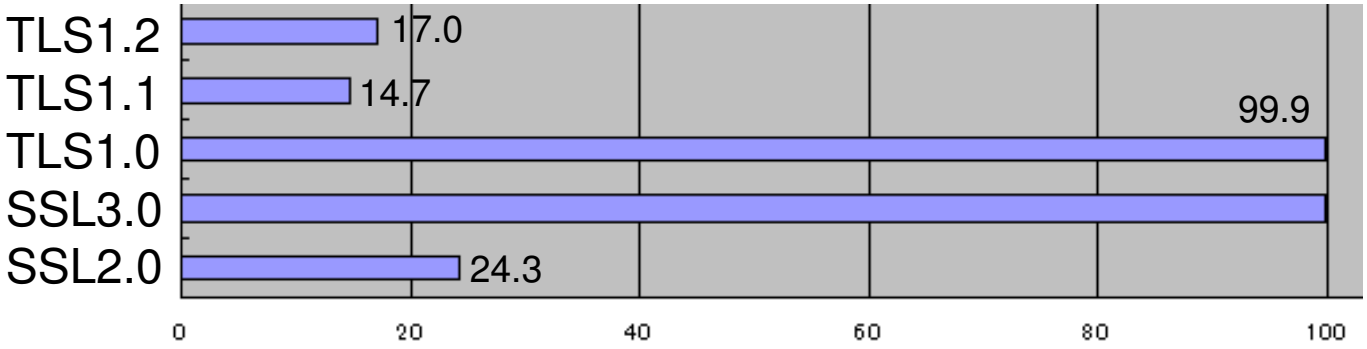
POODLE attack への対策

- （1）SSLv3を捨てる [機会損失]
  - Twitterなどで即座に対応が行われた

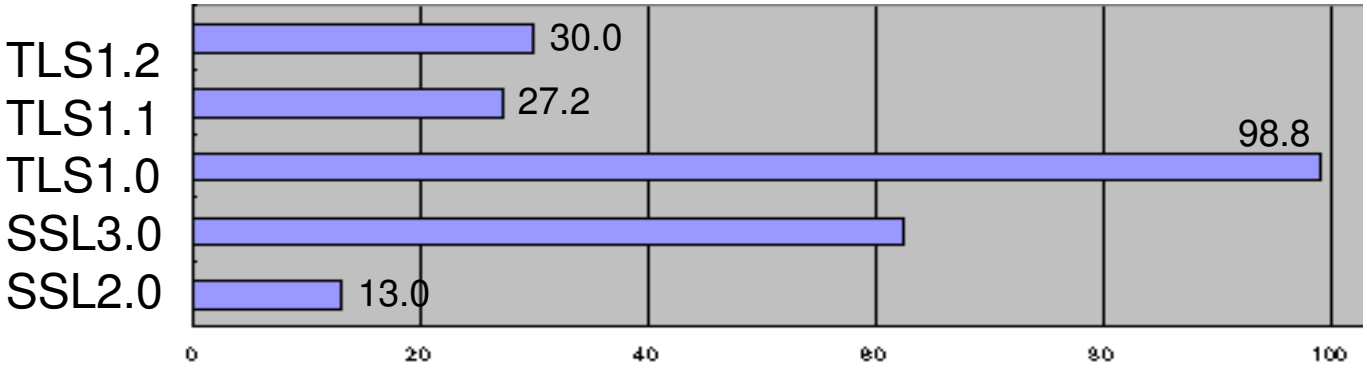
https://tools.ietf.org/html/draft-ietf-tls-downgrade-scsv-02

SSL/TLSサーバのバージョン移行状況

- 4月15日 SSL-enable sites=5677



- 11月26日 SSL-enable sites=5620



99.9%

SSLv3を無効にするサイトが大幅に増加している

62.3%

Alexa top 100M sites に記載されている .jp ドメイン17988サイトを調査
両日ともに同じURLリストを利用

POODLE attack への対策

- （1）SSLv3を捨てる [機会損失]
  – Twitterなどで即座に対応が行われた
- （2）TLS_FALLBACK_ SCSVの導入
  – OpenSSL 10月アップデートで実装済

https://tools.ietf.org/html/draft-ietf-tls-downgrade-scsv-02

POODLE attack への対策

- **（1）SSLv3を捨てる** [機会損失]
  - Twitterなどで即座に対応が行われた
- （2）TLS_FALLBACK_ SCSVの導入
  - OpenSSL 10月アップデートで実装済
- 両方の対策ともレガシーな製品（特にフィーチャーフォンやゲーム機器など）からサイトが閲覧できなくなったりするケースも考えられる

https://tools.ietf.org/html/draft-ietf-tls-downgrade-scsv-02

SSLv3 に延命技術はないのか？

- BEASTのときには 1/n-1 分割法で回避
  - SSL3.0, TLS1.0 でCBC利用時に影響
- SSLv3 で利用できる CipherSuites

SSL_NULL_WITH_NULL_NULL
SSL_RSA_WITH_NULL_MD5
SSL_RSA_WITH_NULL_SHA
SSL_RSA_EXPORT_WITH_RC4_40_MD5
SSL_RSA_WITH_RC4_128_MD5
SSL_RSA_WITH_RC4_128_SHA
SSL_RSA_EXPORT_WITH_RC2_CBC_40_MD5
SSL_RSA_WITH_IDEA_CBC_SHA
SSL_RSA_EXPORT_WITH_DES40_CBC_SHA
SSL_RSA_WITH_DES_CBC_SHA
SSL_RSA_WITH_3DES_EDE_CBC_SHA
SSL_DH_DSS_EXPORT_WITH_DES40_CBC_SHA
SSL_DH_DSS_WITH_DES_CBC_SHA
SSL_DH_DSS_WITH_3DES_EDE_CBC_SHA
SSL_DH_RSA_EXPORT_WITH_DES40_CBC_SHA
SSL_DH_RSA_WITH_DES_CBC_SHA
SSL_DH_RSA_WITH_3DES_EDE_CBC_SHA
SSL_DHE_DSS_EXPORT_WITH_DES40_CBC_SHA
SSL_DHE_DSS_WITH_DES_CBC_SHA
SSL_DHE_DSS_WITH_3DES_EDE_CBC_SHA
SSL_DHE_RSA_EXPORT_WITH_DES40_CBC_SHA
SSL_DHE_RSA_WITH_DES_CBC_SHA
SSL_DHE_RSA_WITH_3DES_EDE_CBC_SHA
SSL_DH_anon_EXPORT_WITH_RC4_40_MD5
SSL_DH_anon_WITH_RC4_128_MD5
SSL_DH_anon_EXPORT_WITH_DES40_CBC_SHA
SSL_DH_anon_WITH_DES_CBC_SHA
SSL_DH_anon_WITH_3DES_EDE_CBC_SHA
SSL_FORTEZZA_KEA_WITH_NULL_SHA
SSL_FORTEZZA_KEA_WITH_FORTEZZA_CBC_SHA
SSL_FORTEZZA_KEA_WITH_RC4_128_SHA

https://www.tools.ietf.org/html/rfc6101

SSLv3 に延命技術はないのか？

- BEASTのときには 1/n-1 分割法で回避
  - SSL3.0, TLS1.0 でCBC利用時に影響
- SSLv3 で利用できる CipherSuites

SSL_NULL_WITH_NULL_NULL
SSL_RSA_WITH_NULL_MD5
SSL_RSA_WITH_NULL_SHA
SSL_RSA_EXPORT_WITH_RC4_40_MD5
SSL_RSA_WITH_RC4_128_MD5
SSL_RSA_WITH_RC4_128_SHA
SSL_RSA_EXPORT_WITH_RC2_CBC_40_MD5
SSL_RSA_WITH_IDEA_CBC_SHA
SSL_RSA_EXPORT_WITH_DES40_CBC_SHA
SSL_RSA_WITH_DES_CBC_SHA
SSL_RSA_WITH_3DES_EDE_CBC_SHA
SSL_DH_DSS_EXPORT_WITH_DES40_CBC_SHA
SSL_DH_DSS_WITH_DES_CBC_SHA
SSL_DH_DSS_WITH_3DES_EDE_CBC_SHA
SSL_DH_RSA_EXPORT_WITH_DES40_CBC_SHA
SSL_DH_RSA_WITH_DES_CBC_SHA
SSL_DH_RSA_WITH_3DES_EDE_CBC_SHA
SSL_DHE_DSS_EXPORT_WITH_DES40_CBC_SHA
SSL_DHE_DSS_WITH_DES_CBC_SHA
SSL_DHE_DSS_WITH_3DES_EDE_CBC_SHA
SSL_DHE_RSA_EXPORT_WITH_DES40_CBC_SHA
SSL_DHE_RSA_WITH_DES_CBC_SHA
SSL_DHE_RSA_WITH_3DES_EDE_CBC_SHA
SSL_DH_anon_EXPORT_WITH_RC4_40_MD5
SSL_DH_anon_WITH_RC4_128_MD5
SSL_DH_anon_EXPORT_WITH_DES40_CBC_SHA

RC4

3DES-CBC

https://www.tools.ietf.org/html/rfc6101

RC4を受け入れるサイト

- あえて「RC4利用時のリスク」を受容しているサイトもある
  - これは暗号リストやガイドラインとは相反する
- 正しく技術を理解してポリシー決めした結果
  - 設定の不備を一概に責めることができない

FREAK attack 概要

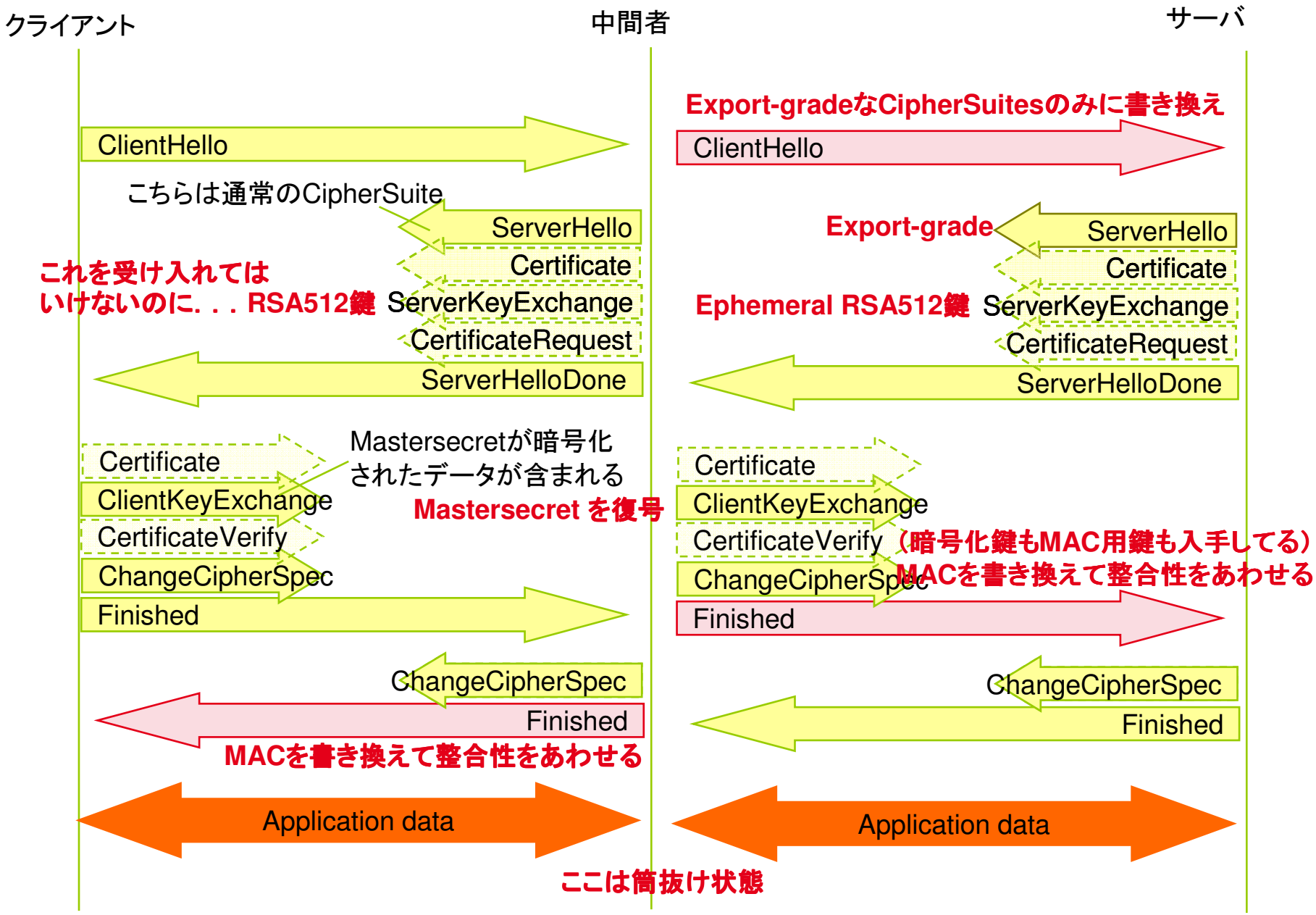
2015年3月

- CVE-2015-0204
- 2015年 日本時間3月3日にサイト公開（再認識）
  - https://freakattack.com/
- 脆弱なバージョン：
  - 2015年1月のUpdateで修正済
  - クライアント

    各系列 1.0.1j以下，1.0.0o以下，0.9.8zd 以下
- 問題：クライアントの指定したCipherSuitesではなくExport-grade（輸出可能な弱い）暗号を意図せず利用されてしまう

攻撃成功のための条件

- 以下の両方が必要
  逆に考えるとサーバサイドの設定変更だけで対策は十分
  - クライアント: OpenSSLの1.0.1k 以前のバージョン（CVE-2015-0204の脆弱性を持つ）
  - サーバ: EXP_* （Export-grade）なCipherSuitesをしゃべる設定がされている
    - Ephemeral RSA鍵が使いまわされている
      （実際，起動時に同じ鍵を使う実装が存在）
- 事前にサーバのRSA512鍵を取得（誰でも可能）→ごり押しで素因数分解して秘密鍵取得
- 公衆無線LANなどを利用してDNS詐称して本サーバを攻撃者サーバにすげ替える



Freakattack.com

- これまでにも「サーバ設定の不備が風評被害になりうる」と言ってた→今回わりと現実的に
  - 技術的に理解していない方でもブラックリストのように見えるわけで，実際Twitterなどでいくつかの日本のサーバが名指しされている.
- さすがにもうこの時期に輸出規制時代のCipherSuitesをサポートすることは後方互換性の確保のため，という理由にはならないか

OpenSSL 2015-03-19 Updates

- 事前に予告があり「そのときを」備えることができた
  - 日本時間 3/19 20:00-24:00 →23時ちょい過ぎ
- その日のうちに「解散宣言」しましたよね

OpenSSL 2015-03-19 Updates

- FREAK攻撃:Low→High Severity に格上げ
  - 間違いを素直に正している点は評価
  - でも Low に分類してたのはなぜ？
    - ここでもアカデミアとインダストリの考え方の違いか？
- いくつかの観点で分類してみると
  - 1.0.2 だけで起こるケース
  - DoSを誘発するケース（サーバサイドだけの問題）
  - Low severity

今回こう考えてみた

- 「意識の高い」管理者が使っていると思われる 1.0.2 は速やかにパッチあてられるはず
  - 一方でレガシー環境で古い枯れたライブラリを使っている製品の方が移行は困難
- サーバサイドで対応できるものを対応していない場合は「風評被害」を受けてもしようがない
- そうすると OpenSSL を使っているクライアントサイドで，どのくらい即時性を持って対処されるかにかかってくるのではないか？
  - 主要ブラウザの更新頻度は相当早い

まとめ

（その場でいいます）