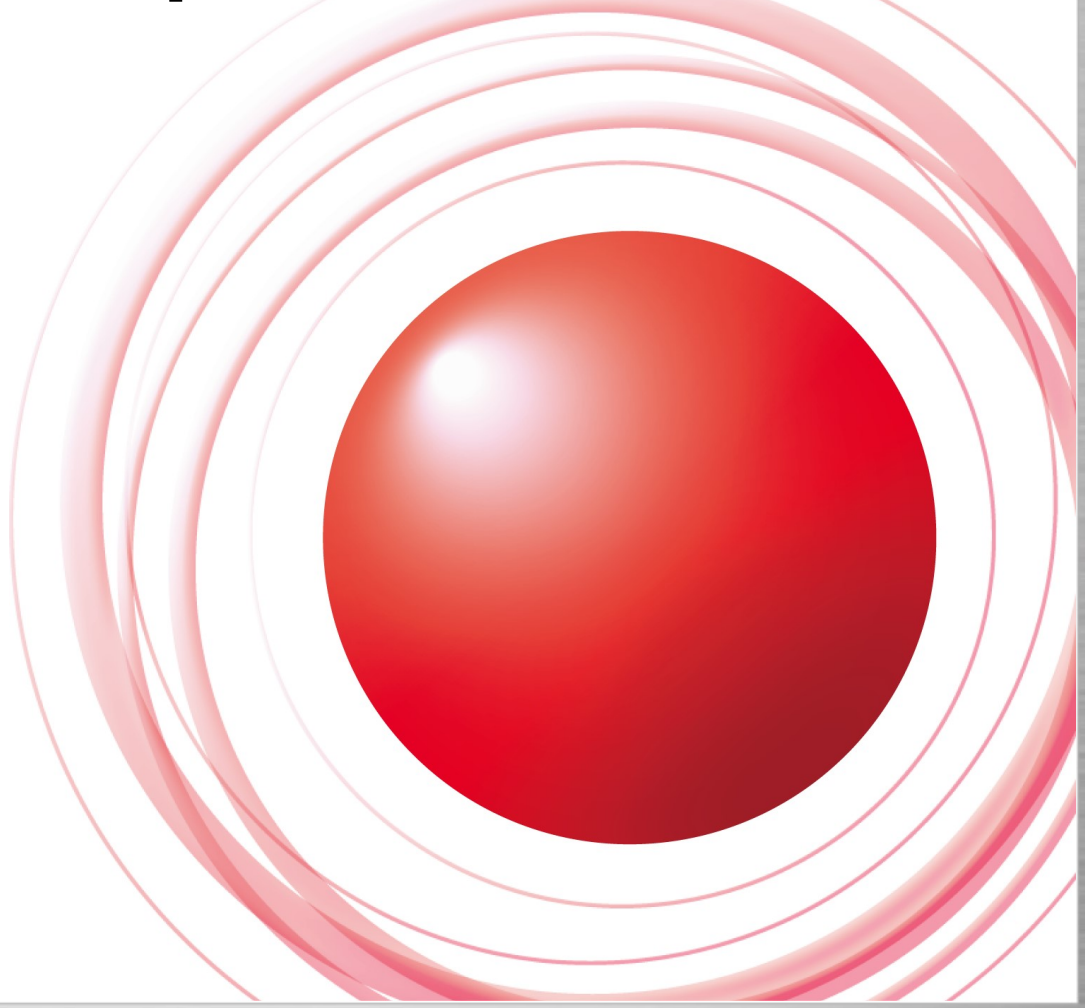
CRYPTRECシンポジウム2019, Tokyo, Japan

www.iij.ad.jp
安全な接続

ガイドライン改訂に伴う論点整理と TLS1.3ディプロイメントについて



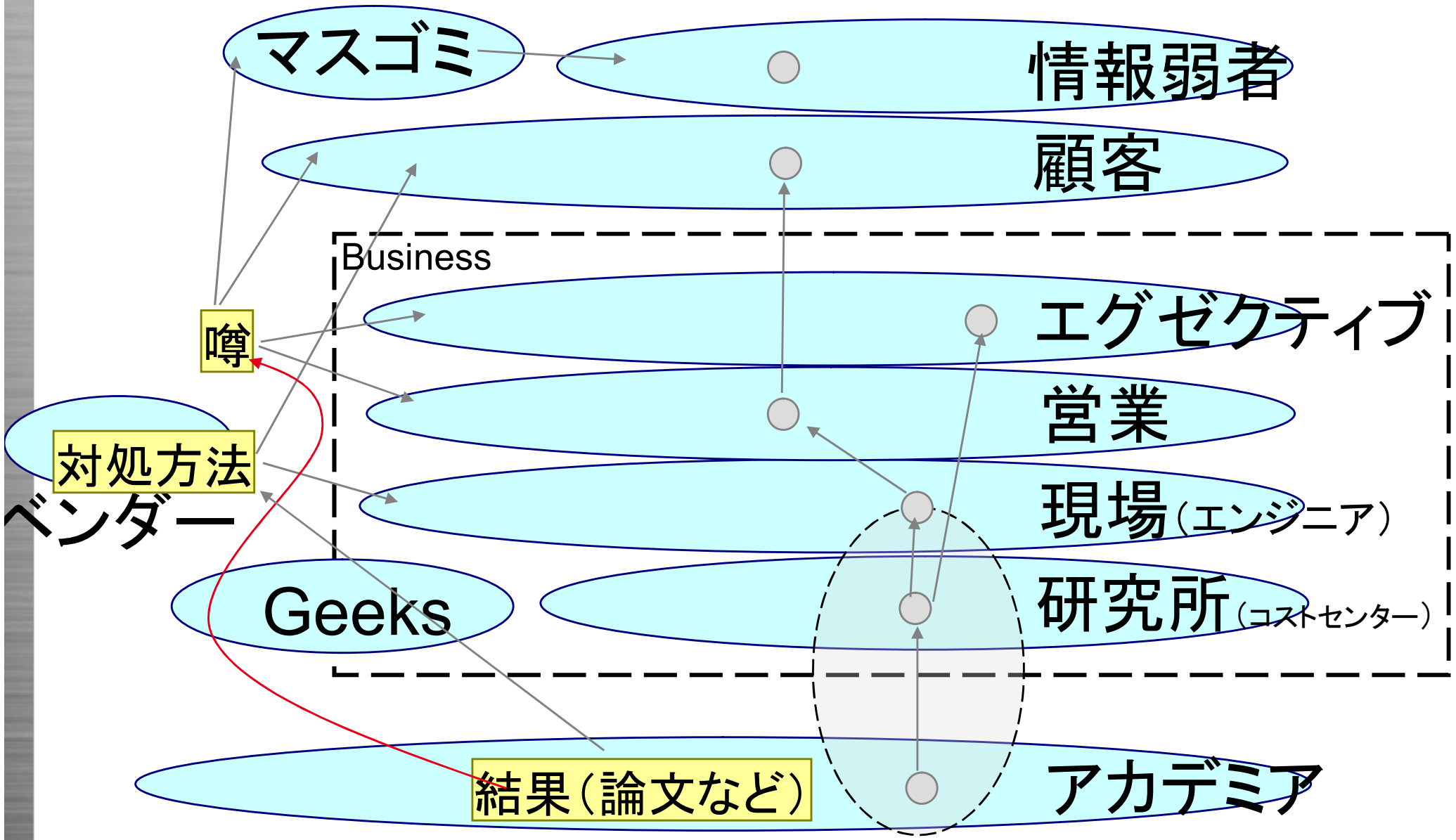
Internet Initiative Japan

wizSafe

セキュリティ本部セキュリティ情報統括室
須賀祐治
2019-07-12

自己紹介？
～昔の発表資料から～

「技術翻訳者」連携の必要性



2012年3月総合大会「セキュリティプロトコル仕様の脆弱性事例と世代交代困難性」

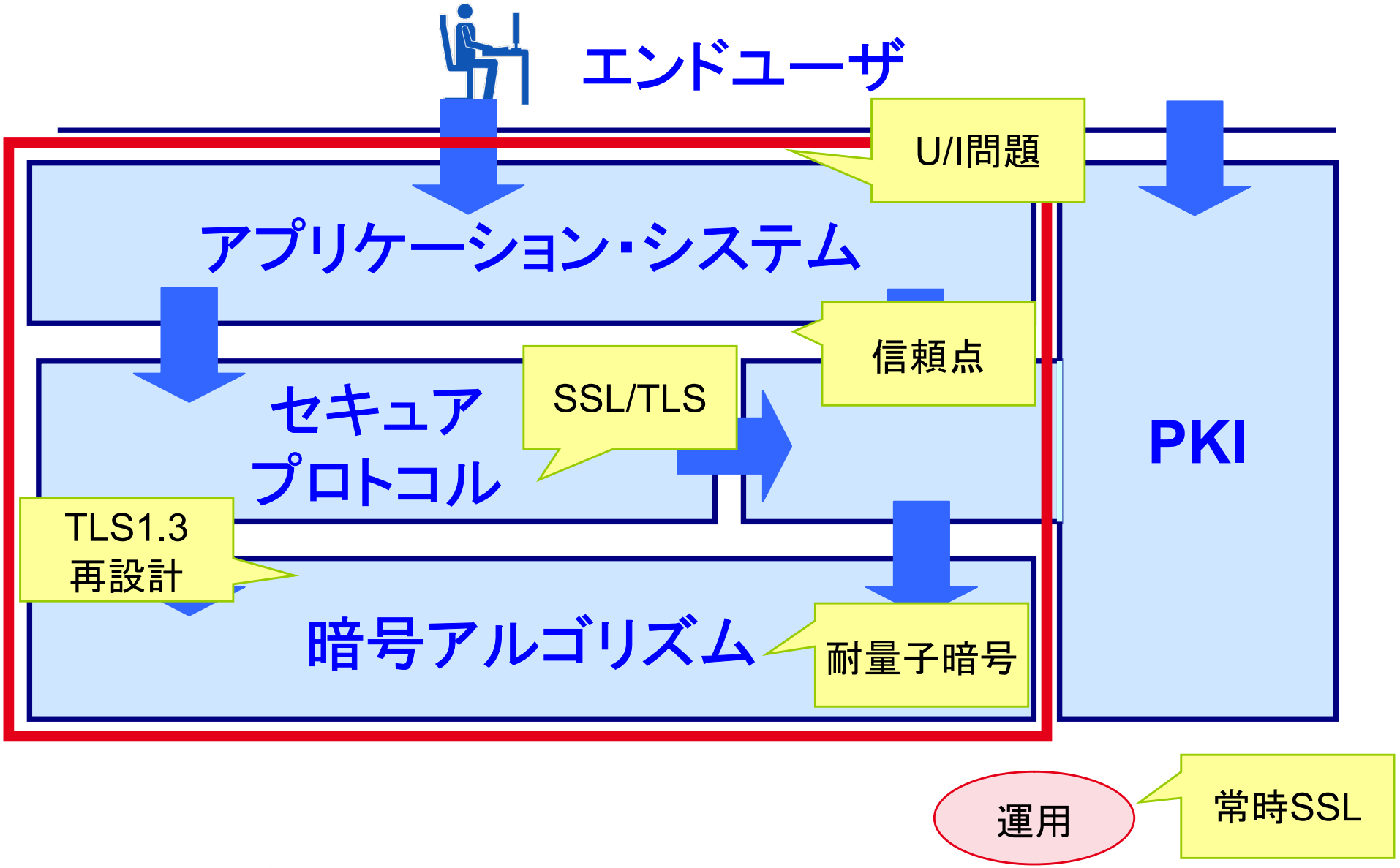
「技術翻訳者」の役割

- 正しい情報をわかりやすく「上」に伝える
  - 落としてよい情報と肝の情報
- 誤りがあればそれを正す
  - 噂が広まるのは早い
- どう解釈しているのか「横」に伝える

2012年3月総合大会「セキュリティプロトコル仕様の脆弱性事例と世代交代困難性」

SSL/TLS バージョンと
移行に関する外的要因

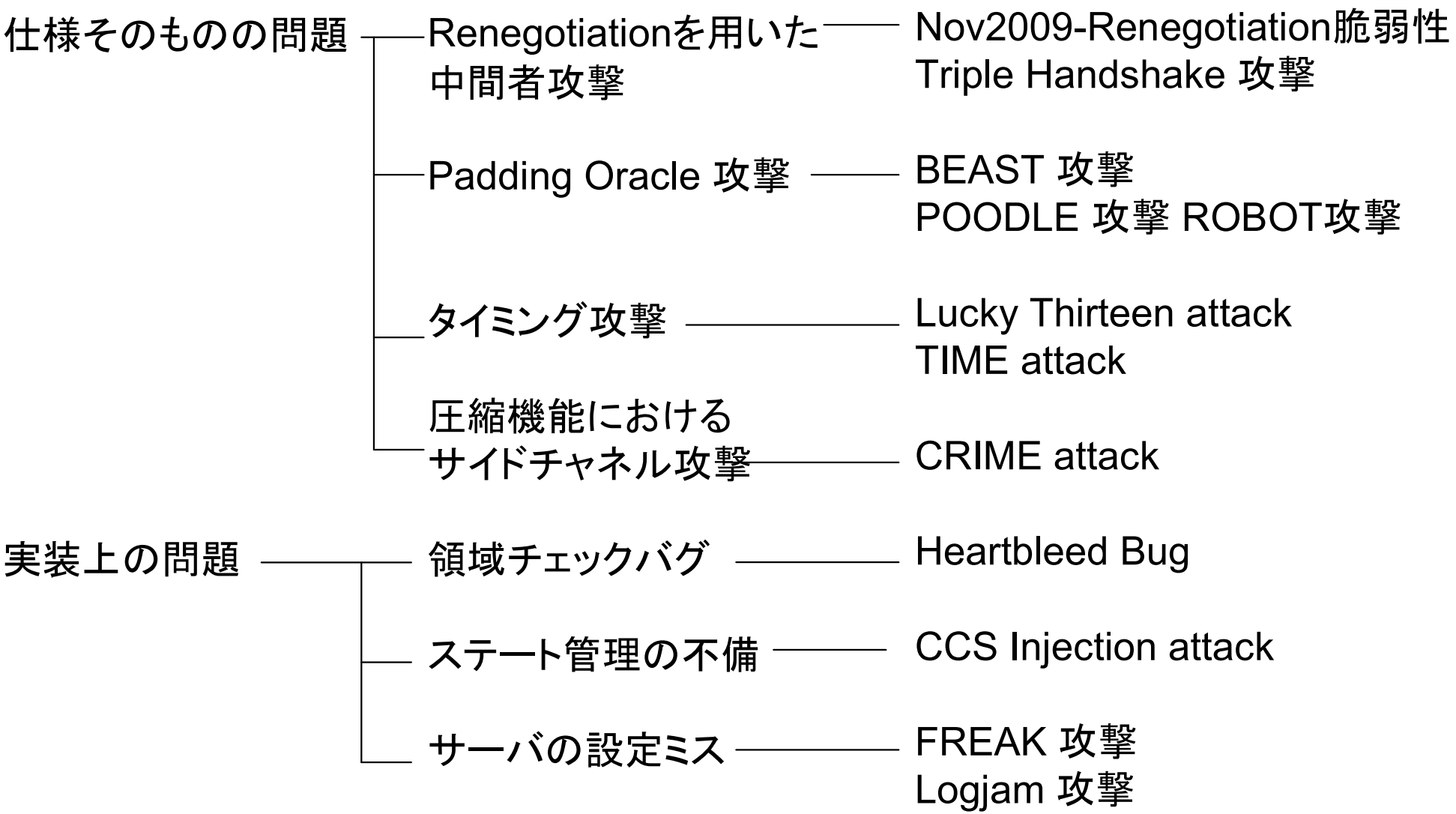
今回取り扱う技術の範囲



SSL/TLSの経緯

- Netscape Communicationsから1995年にInternet draft が公開された時期と同じくして当時のブラウザNetscape NavigatorにSSL2.0が実装
- 2014年10月に発表されたPOODLE攻撃はSSL3.0における根本的な問題を露呈
- SSLの後継でありIETFで策定されたTLSは1.0/1.1/1.2が1999/2006/2008年にそれぞれRFCとして公開
- 2014年4月からTLS1.3ドラフト策定開始．28版の改訂後，2018年8月ついにTLS1.3がRFCとして公開

SSL/TLSに対する攻撃の分類



SSL/TLSバージョンごとの状況

プロトコル	バージョン	ステータス	ワークアラウンド	根拠
SSL	2.0	脆弱	なし	RFC6167
	3.0	脆弱	なし	RFC7568
				POODLE攻撃

IETFがSSL2.0/3.0を無効化

（トップダウン的な措置）

- 2011年3月
  - RFC 6176：Prohibiting Secure Sockets Layer（SSL）Version 2.0
- 2015年6月
  - RFC 7568：Deprecating Secure Sockets Layer Version 3.0
  - http://disablessl3.com/ by Michele Spagnuolo

SSL/TLSバージョンごとの状況

プロトコル	バージョン	ステータス	ワークアラウンド	根拠
SSL	2.0	脆弱	なし	RFC6167
	3.0	脆弱	なし	RFC7568
				POODLE攻撃
TLS	1.0	問題はあるが回避策あり （ただし回避策がないものもある）	Renegotiation機能を利用しない	RFC5746
			圧縮機能を利用しない	CRIME攻撃
			1/n-1分割法	BEAST攻撃
			リスク受容	Lucky13攻撃
	1.1	問題はあるが回避策あり （ただし回避策がないものもある）	圧縮機能を利用しない	CRIME攻撃
			リスク受容	Lucky13攻撃
	1.2	問題はあるが回避策あり	圧縮機能を利用しない	CRIME攻撃
			暗号モードとしてGCM, CCMのみを利用	Lucky13攻撃
	1.3	安全に設計したと信じられている		

2018年9月 TLS1.0/1.1排除の初動

- SSL3.0などと同様の die-die-die ドラフト発行
  - https://tools.ietf.org/html/draft-ietf-tls-oldversions-deprecate
  - “Deprecating TLSv1.0 and TLSv1.1”
  - 移行状況に関する数値的根拠も紹介されている（-00 draft：Webサーバの対応状況は以下の通り）

Name/Ref	Date	SSLv3	TLSv1.0	TLSv1.1	TLSv1.2	TLSv1.3
Alexa [1]	20180226	-	2.0	<0.1	97.9	-
Cloudflare [2]	20180518	0.0	9.3	0.2	84.9	5.5
Firefox [3]	20180709	-	1.0	-	94.0	5.0
Chrome [4]	20180711	-	0.4	<0.1	-	-

世の中のバージョン移行動向

https://www.ssllabs.com/ssl-pulse/

- 毎月レポートが発出
  - プロトコル種別などの先月との差分表示

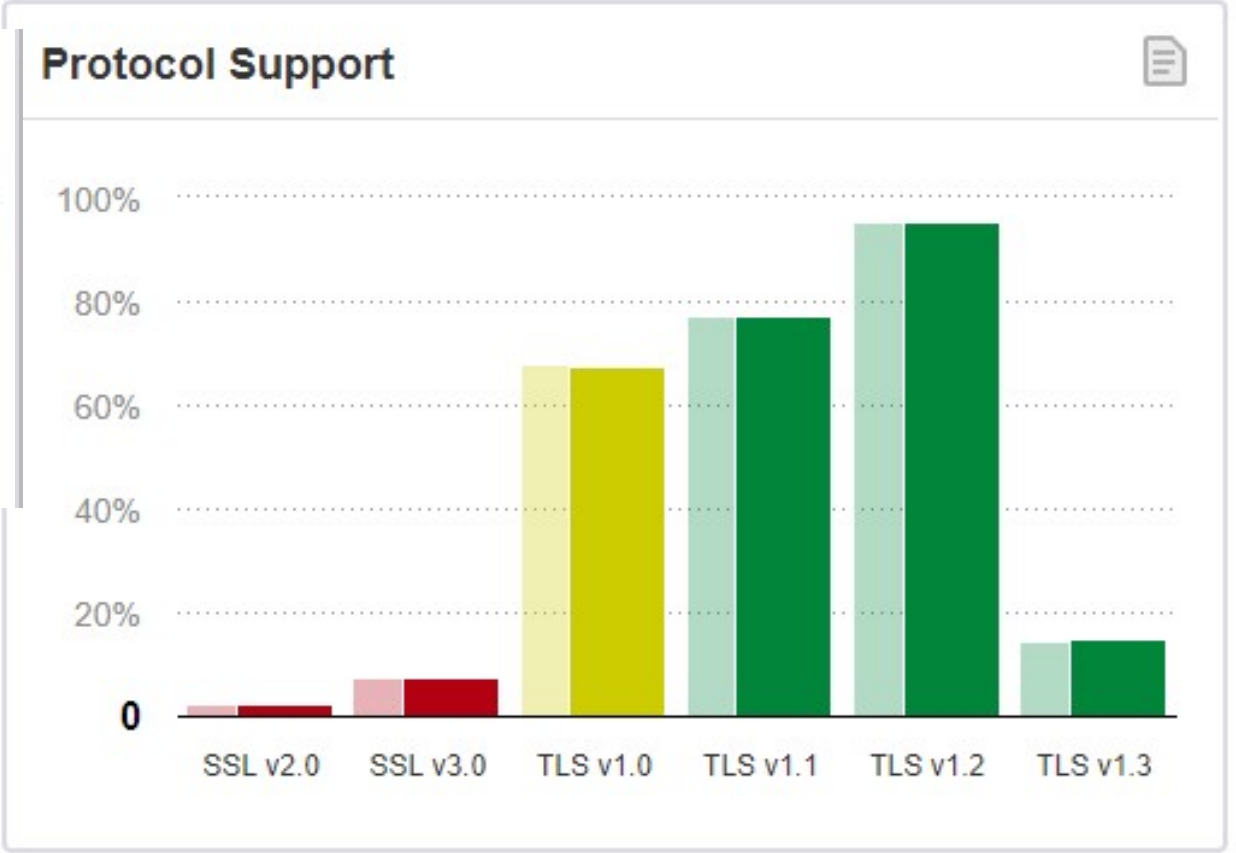
June 2019

14.8 % of sites surveyed support the **TLS v1.3** protocol

20,661 sites + 0.6 %

May 2019

14.2 % (19,821 sites)



参考：K. Urushimaさん http://blog.livedoor.jp/k_urushima/archives/1786599.html

2018年10月 TLS1.0/1.1排除の本格化

- 主要ブラウザベンダーが同時に
  2020年前半に無効化するアナウンス
  - Chrome, Edge/IE, Firefox, Safari

2020 年 IE, Edge で TLS 1.0, 1.1 での接続無効化。確認を！

Rate this article ★★★★★

YURIKAM 2018/10/16

Share 139 | 0 | in 0

こんにちは、垣内ゆりかです。

マイクロソフトでは、Transport Layer Security (TLS) 1.0, 1.1 の利用を廃止し、より安全なプロトコルである TLS 1.2 以降への移行を推奨しています。(参考: 過去ブログ [IT 管理者向け] TLS 1.2 への移行を推奨しています)

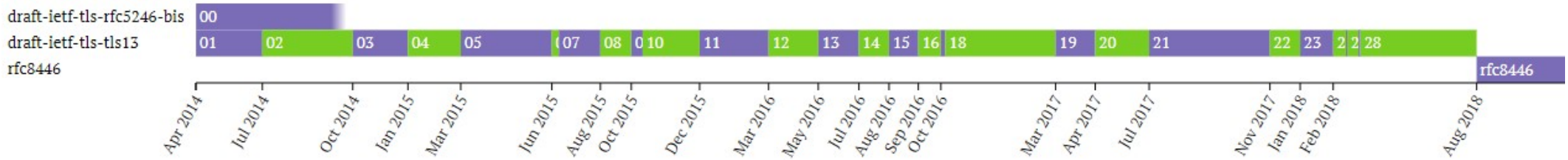
2020 年 前半、Internet Explorer 11, Microsoft Edge にて、 TLS 1.0 および TLS 1.1 を既定で無効化する措置を行う予定です。

https://blogs.technet.microsoft.com/jpsecurity/2018/10/16/tlsdeprecation/

TLS1.3の登場

- メジャーアップデート
  - 新しい暗号アルゴリズムへの移行
  - 0-RTTによる高速処理可能なシーケンス
- ガイドラインとの整合性を再検討
  - 3つのシチュエーションを想定するも
    ブラウザ-サーバという環境のみを想定
- TLS1.3実装状況と
  - TLS1.1以下の排除が加速する
    →考えていたよりも早い移行が予想される
    - 緩やかなSHA-3移行とはタイムラインが異なる

TLS1.3 仕様の過去の参照先



- ドラフト更新の経緯を参照可能
  - https://tools.ietf.org/html/draft-ietf-tls-tls13
- IETFとは異なるリポジトリで頻繁に更新
  - https://tlswg.github.io/tls13-spec/

過去の議論を観るには？

- Github
  - https://github.com/tlswg/tls13-spec/issues
- IETF meeting
  - https://tools.ietf.org/agenda/105/
    - 年3回のIETF meeting. 次回は7月末のIETF-105
    - TLS WGだけでなくsecdispatch WGやQUIC WG さらにセキュリティ横断的にCFRG等でもTLS関連技術が議論されるため要チェック

フォーマルメソッドで設計中のプロトコルの穴を見つける

- 米山一樹，セキュリティプロトコル安全性検証の理想と現実，CRYPTRECシンポジウム2016
  - https://www.cryptrec.go.jp/symposium/20160627_invited1.pdf
- レピダムBlog, 祝RFC！Transport Layer Security (TLS) 1.3 発行の軌跡 ～熟成された4年間の安全性解析～
  - https://lepidum.co.jp/blog/2018-10-01/tls1_3security/

TLS1.3 における再設計

懐かしいね
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

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．

2013年の問い「CBCを捨てますか？」

- 単純な対策方法：CBCを使わない
  - Lucky13（2月5日）の風潮：RC4使おう！
    - Lucky Thirteen: Breaking the TLS and DTLS Record Protocols
    - http://www.isg.rhul.ac.uk/tls/Lucky13.html
- しかしRC4も死亡（3月13日）
    - Breaking the TLS and DTLS Record Protocols
    - http://www.isg.rhul.ac.uk/tls
- AEADの普及率が課題…

代替手段があるかどうか？＋相互接続性を確保できるか？

2013年3月 第3回神保町暗号ワークショップ「CBCモードの現在（いま）～CBCモードに何が起こっているか？」

ちょっと面白そうな論文

https://www.usenix.org/conference/usenixsecurity19/presentation-21

Scalable Scanning and Automatic Classification of TLS Padding Oracle Vulnerabilities

Authors:
Robert Merget and Juraj Somorovsky, *Ruhr University Bochum;* Nimrod Aviram, *Tel Aviv University;* Craig Young, *Tripwire VERT;* Janis Fliegenschmidt and Jörg Schwenk, *Ruhr University Bochum;* Yuval Shavitt, *Tel Aviv University*

Abstract:
This paper is under embargo and will be released to the public on the first day of the symposium, August 14, 2019.

The TLS protocol provides encryption, data integrity, and authentication on the modern Internet. Despite the protocol's importance, currently-deployed TLS versions use obsolete cryptographic algorithms which have been broken using various attacks. One prominent class of such attacks is CBC padding oracle attacks. These attacks allow an adversary to decrypt TLS traffic by observing different server behaviors which depend on the validity of CBC padding.

We present the first large-scale scan for CBC padding oracle vulnerabilities in TLS implementations on the modern Internet. Our scan revealed vulnerabilities in 1.83% of the Alexa Top Million websites, detecting nearly 100 different vulnerabilities. Our scanner observes subtle differences in server behavior, such as responding with different TLS alerts, or with different TCP header flags.

RUB-NDS / TLS-Padding-Oracles

Watch 14 | Star 95 | Fork 8

Code | Issues 0 | Pull requests 0 | Projects 0 | Wiki | Security | Insights

Branch: master ▾ TLS-Padding-Oracles / TlsPaddingOracleScanning.pdf

Find file | Copy path

jurajsomorovsky Add files via upload efddd9c 21 days ago

1 contributor

806 KB

Download | History

Scalable Scanning and Automatic Classification of TLS Padding Oracle Vulnerabilities

Robert Merget[1], Juraj Somorovsky[1], Nimrod Aviram[2], Craig Young[3], Janis Fliegenschmidt[1], Jörg Schwenk[1], and Yuval Shavitt[2]

[1]Ruhr University Bochum
[2]Department of Electrical Engineering, Tel Aviv University
[3]Tripwire VERT

https://github.com/RUB-NDS/TLS-Padding-Oracles/blob/master/TlsPaddingOracleScanning.pdf

TLS Padding Oracles

The TLS protocol provides encryption, data integrity, and authentication on the modern Internet. Despite the protocol's importance, currently-deployed TLS versions use obsolete cryptographic algorithms which have been broken using various attacks. One prominent class of such attacks is CBC padding oracle attacks. These attacks allow an adversary to decrypt TLS traffic by observing different server behaviors which depend on the validity of CBC padding.

We evaluated the Alexa Top Million Websites for CBC padding oracle vulnerabilities in TLS implementations and revealed vulnerabilities in 1.83% of them, detecting nearly 100 different vulnerabilities. These padding oracles stem from subtle differences in server behavior, such as responding with different TLS alerts, or with different TCP header flags. We suspect the subtlety of different server responses is the reason these padding oracles were not detected previously.

The currently identified and fixed vulnerabilities are:

- OpenSSL. CVE-2019-1559. OpenSSL Security Advisory: 0-byte record padding oracle
- Citrix. CVE-2019-6485. TLS Padding Oracle Vulnerability in Citrix Application Delivery Controller (ADC) and NetScaler Gateway.
- F5. CVE-2019-6593. TMM TLS virtual server vulnerability CVE-2019-6593.
- SonicWall SonicOs. CVE-2019-7477. SonicOS & SonicOSv CBC Cipher TLS Padding Vulnerability.

The disclosure process is still running with a handful of vendors. Some of them consider to disable or even completely remove CBC cipher suites from their products.

https://github.com/RUB-NDS/TLS-Padding-Oracles

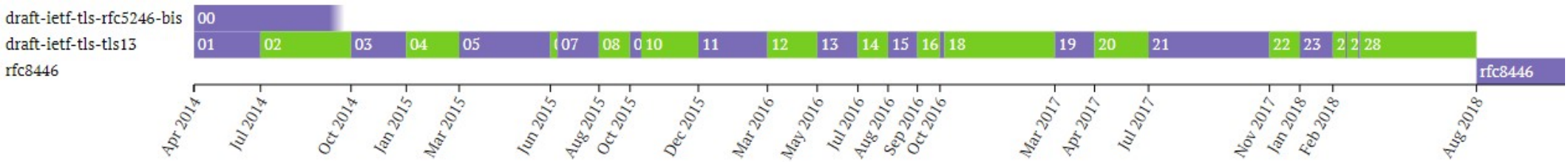
これあれやん（CVE-2019-1559）

- OpenSSL Security Advisory [26 February 2019]
  0-byte record padding oracle (CVE-2019-1559)
  ==========
- Severity: Moderate
- If an application encounters a fatal protocol error and then calls SSL_shutdown() twice (once to send a close_notify, and once to receive one) then OpenSSL can respond differently to the calling application if a 0 byte record is received with invalid padding compared to if a 0 byte record is received with an invalid MAC. If the application then behaves differently based on that in a way that is detectable to the remote peer, then this amounts to a padding oracle that could be used to decrypt data.
- In order for this to be exploitable "non-stitched" ciphersuites must be in use. Stitched ciphersuites are optimised implementations of certain commonly used ciphersuites. Also the application must call SSL_shutdown() twice even if a protocol error has occurred (applications should not do this but some do anyway).

AEAD ciphersuites are not impacted.

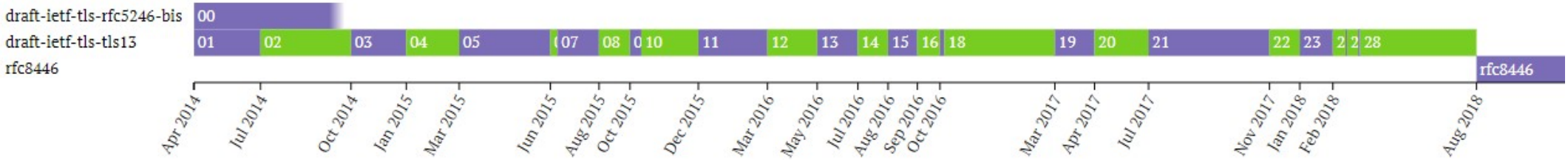
https://www.openssl.org/news/secadv/20190226.txt

Rev重ねるごとに落選していく

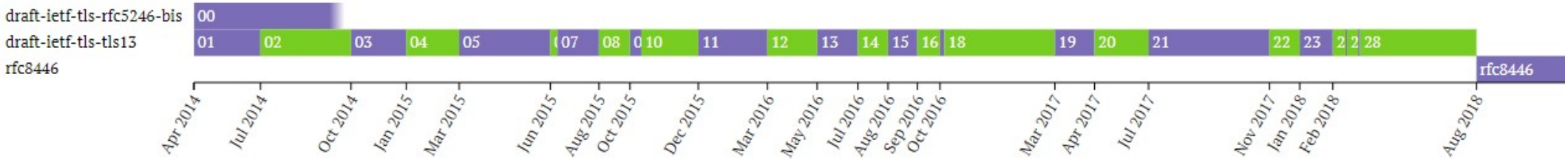


- draft-02
  - Remove custom DHE groups.
  - Remove support for compression.
  - Remove support for static RSA and DH key exchange.
  - Remove support for non-AEAD ciphers.
- draft-03
  - Remove the unnecessary length field from the AD input to AEAD ciphers.
- draft-06
  - Prohibit RC4 negotiation for backwards compatibility.

https://tlswg.org/tls13-spec/draft-ietf-tls-tls13.html



- draft-07
  - Integration of semi-ephemeral DH proposal.
  - Remove resumption and replace with PSK + tickets.
  - Move to HKDF.
- draft-08
  - Remove support for weak and lesser used named curves.
  - Remove support for MD5 and SHA-224 hashes with signatures.



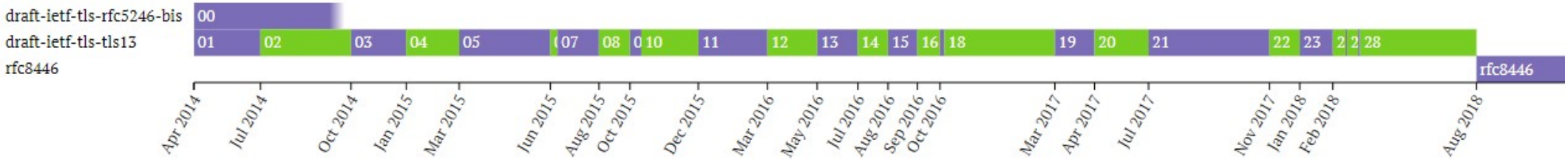
- draft-09
  - Change to RSA-PSS signatures for handshake messages.
  - Remove support for DSA.
  - Deprecate SHA-1 with signatures.
- draft-11
  - Port the CFRG curves & signatures work from RFC4492bis.

2016年1月の時点の様子

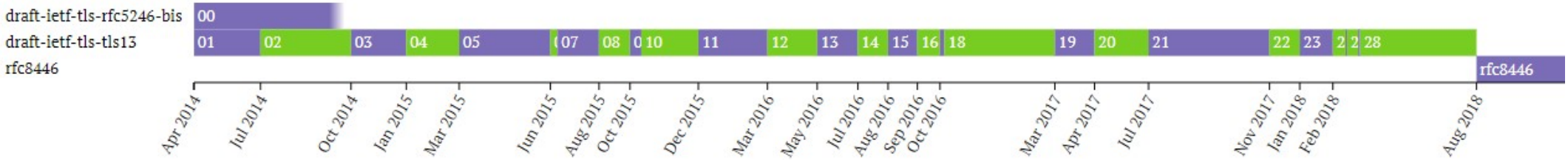
- https://tlswg.github.io/tls13-spec/#rfc.appendix.A.4

Cipher Suite Name	Value	Specification
TLS_DHE_RSA_WITH_AES_128_GCM_SHA256	{0x00,0x9E}	[RFC5288]
TLS_DHE_RSA_WITH_AES_256_GCM_SHA384	{0x00,0x9F}	[RFC5288]
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	{0xC0,0x2B}	[RFC5289]
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	{0xC0,0x2C}	[RFC5289]
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	{0xC0,0x2F}	[RFC5289]
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	{0xC0,0x30}	[RFC5289]
TLS_DHE_RSA_WITH_AES_128_CCM	{0xC0,0x9E}	[RFC6655]
TLS_DHE_RSA_WITH_AES_256_CCM	{0xC0,0x9F}	[RFC6655]
TLS_DHE_RSA_WITH_AES_128_CCM_8	{0xC0,0xA2}	[RFC6655]
TLS_DHE_RSA_WITH_AES_256_CCM_8	{0xC0,0xA3}	[RFC6655]
TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256	{TBD,TBD}	[I-D.ietf-tls-chacha20-poly1305]
TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256	{TBD,TBD}	[I-D.ietf-tls-chacha20-poly1305]
TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256	{TBD,TBD}	[I-D.ietf-tls-chacha20-poly1305]

TLS_（鍵共有）_（署名）_（**AEADのアルゴリズム**）_（ハッシュ関数）



- draft-13
  - Require DH public keys and secrets to be zero-padded to the size of the group.
- draft-14
  - Define ecdsa_sha1 (*).
- draft-15
  - Remove old PRNG text.



- draft-16
  - Change RSASSA-PSS and EdDSA SignatureScheme codepoints for better backwards compatibility (*)
- draft-23
  - Add some text on the security of static RSA.

共通鍵暗号

- DESはTLS1.2で既に排除済み
- RC4についてもRFC7465が2015年2月に発行されたことを受けて排除済み

- CBC暗号モード排除．共通鍵暗号としてはAEAD（Authenticated Encryption with Associated Data）のみの利用に統一
  - ChaCha20-Poly1305がAES-GCMと並んで実装必須（Mandatory Algorithms）

ハッシュ関数

- MD5, SHA-224 を排除（MUST NOT）
- SHA-1 は SHOULD NOT
  - 後方互換性のためのSHA-1サポート
    - 特にSHA-1署名での証明書検証はサポート

CipherSuitesの簡略化

- 5つのCipherSuitesのみを規定
  - （これまでのCipherSuitesと同じ呼び方でいいのか？）
- これまでの反省のもと「記載」をかなり簡略化

Description	Value	
TLS_AES_128_GCM_SHA256	{0x13,0x01}	**MUST**
TLS_AES_256_GCM_SHA384	{0x13,0x02}	**SHOULD**
TLS_CHACHA20_POLY1305_SHA256	{0x13,0x03}	**SHOULD**
TLS_AES_128_CCM_SHA256	{0x13,0x04}	
TLS_AES_128_CCM_8_SHA256	{0x13,0x05}	

TLS_（**AEADのアルゴリズム**）_（**ハッシュ関数**）

IANAで規定の各種パラメータ

Transport Layer Security (TLS) Parameters

Created
2005-08-23
Last Updated
2019-04-22
Available Formats

Registries included below

- TLS ClientCertificateType Identifiers
- TLS Cipher Suites
- TLS ContentType
- TLS Alerts
- TLS HandshakeType
- TLS Supported Groups
- TLS EC Point Formats
- TLS EC Curve Types
- TLS Supplemental Data Formats (SupplementalDataType)
- TLS UserMappingType Values
- TLS SignatureAlgorithm
- TLS HashAlgorithm
- TLS Exporter Labels
- TLS Authorization Data Formats
- TLS Heartbeat Message Types
- TLS Heartbeat Modes
- TLS SignatureScheme
- TLS PskKeyExchangeMode

https://www.iana.org/assignments/tls-parameters/tls-parameters.xhtml

CCM_8_SHA256 推奨の変化

Value	Description	DTLS-OK	Recommended	Reference
0x13,0x01	TLS_AES_128_GCM_SHA256	Y	Y	[RFC8446]
0x13,0x02	TLS_AES_256_GCM_SHA384	Y	Y	[RFC8446]
0x13,0x03	TLS_CHACHA20_POLY1305_SHA256	Y	Y	[RFC8446]
0x13,0x04	TLS_AES_128_CCM_SHA256	Y	Y	[RFC8446]
0x13,0x05	TLS_AES_128_CCM_8_SHA256	Y	N	[RFC8446][IES

6.3 TLS registry updates (Benjamin Kaduk)

The management issue was discussed. The IESG agreed to use IESG Action to effect a "Y"->"N" change in the value of the "Recommended" column for the following TLS registry entries:

- o TLS_AES_128_CCM_8_SHA256 in the TLS Cipher Suites registry
- o truncated_hmac in the TLS ExtensionType Values registry

Internet Engineering Steering Group
Area Director of Security Area

https://www6.ietf.org/iesg/minutes/2018/minutes-2018-08-16.txt

署名・鍵交換

- DSA排除（ただし現時点で脆弱ではない）して
  ECDSA利用に完全シフト
- もちろん署名としてはECDSAだけではなく
  RSA-PSSも利用可能

- 一方で鍵交換に使われるDHは排除されることなく
  ECDHと共に残留
- 毎回異なる鍵（Ephemeral keys）を生成する
  Forward secrecy を満たす方式のみ

1軍(電子政府推奨暗号リスト)

電子政府推奨暗号リスト

暗号技術検討会[1]及び関連委員会(以下、「CRYPTREC」という。)により安全性及び実装性能が確認された暗号技術[2]について、市場における利用実績が十分であるか今後の普及が見込まれると判断され、当該技術の利用を推奨するもののリスト。

技術分類		名称
公開鍵暗号	署名	DSA
		ECDSA
		RSA-PSS(注1)
		RSASSA-PKCS1-v1_5(注1)
	守秘	RSA-OAEP(注1)
	鍵共有	DH
		ECDH

- いずれもリストに掲載されており安全であると認識されている
- レガシーな暗号であるという論理だけで移行が進められているという問題がある(私見)
  - 傷もついていないのに…

B.3.1.3. Signature Algorithm Extension

```
enum {
    /* RSASSA-PKCS1-v1_5 algorithms */
    rsa_pkcs1_sha256(0x0401),
    rsa_pkcs1_sha384(0x0501),
    rsa_pkcs1_sha512(0x0601),

    /* ECDSA algorithms */
    ecdsa_secp256r1_sha256(0x0403),
    ecdsa_secp384r1_sha384(0x0503),
    ecdsa_secp521r1_sha512(0x0603),

    /* RSASSA-PSS algorithms with public key OID rsaEncryption */
    rsa_pss_rsae_sha256(0x0804),
    rsa_pss_rsae_sha384(0x0805),
    rsa_pss_rsae_sha512(0x0806),

    /* EdDSA algorithms */
    ed25519(0x0807),
    ed448(0x0808),

    /* RSASSA-PSS algorithms with public key OID RSASSA-PSS */
    rsa_pss_pss_sha256(0x0809),
    rsa_pss_pss_sha384(0x080a),
    rsa_pss_pss_sha512(0x080b),

    /* Legacy algorithms */
    rsa_pkcs1_sha1(0x0201),
    ecdsa_sha1(0x0203),

    [redacted]

} SignatureScheme;
```

MUST for certificates

MUST

MUST for CertificateVerify and certificates

Edwards曲線上の Schnorr 署名ライクな EdDSA

4.2.7. Supported Groups

When sent by the client, the "supported_groups" extension indicates the named groups which the client supports for key exchange, ordered from most preferred to least preferred.

```
enum {

    /* Elliptic Curve Groups (ECDHE) */
    secp256r1(0x0017), secp384r1(0x0018), secp521r1(0x0019),
    x25519(0x001D), x448(0x001E),

    /* Finite Field Groups (DHE) */
    ffdhe2048(0x0100), ffdhe3072(0x0101), ffdhe4096(0x0102),
    ffdhe6144(0x0103), ffdhe8192(0x0104),

    /* Reserved Code Points */
    ffdhe_private_use(0x01FC..0x01FF),
    ecdhe_private_use(0xFE00..0xFEFF),
    (0xFFFF)
} NamedGroup;
```

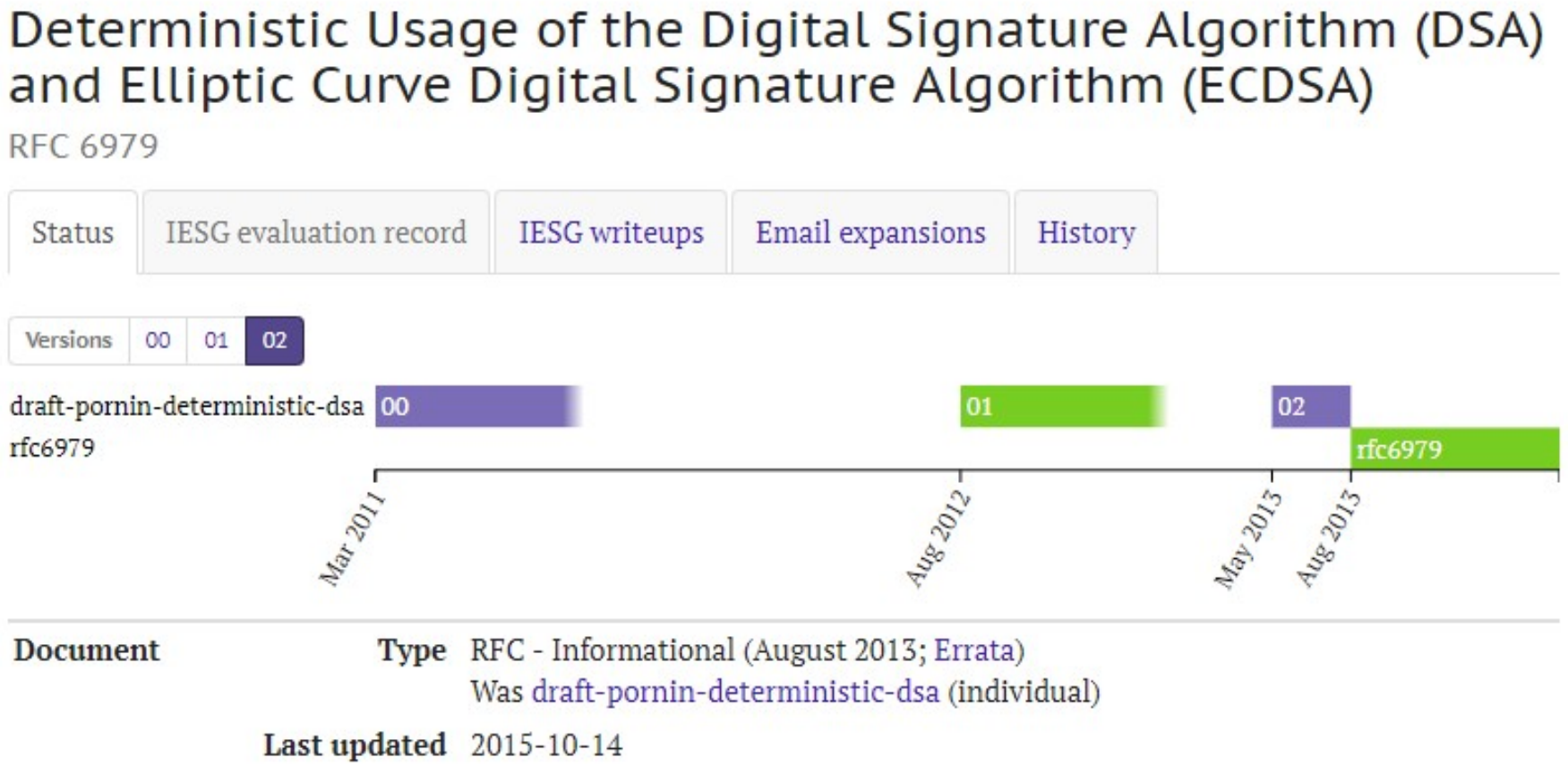
MUST
SHOULD

secp256r1, secp384r1, secp521r1: Standards for Efficient Cryptography Group, SEC 2: "Recommended Elliptic Curve Domain Parameters“ http://www.secg.org/sec2-v2.pdf

x25519, x448：”Elliptic Curves for Security”, https://datatracker.ietf.org/doc/rfc7748/

脱線（本日3回目）

- これ使っていいかよーわからん
- Bitcoin署名時のPRNG空間の狭さでノンス使いまわして奪取問題．これが対策のひとつ．



Post Quantum な暗号ライブラリ

いくつかの先駆的な取り組み

- Microsoft Research, Open Quantum Safe library
  - https://github.com/open-quantum-safe/
  - https://qtesla.org/
- Google, Post-quantum confidentiality for TLS
  - https://www.imperialviolet.org/2018/04/11/pqconftls.html
- CloudFlare, Introducing CIRCL: An Advanced Cryptographic Library
  - https://blog.cloudflare.com/introducing-circl/

PQC Standardization Process: Second Round Candidate Announcement

January 30, 2019

After over a year of evaluation, NIST would like to announce the candidates that will be moving on to the 2nd round of the NIST PQC Standardization Process.

The 17 Second-Round Candidate public-key encryption and key-establishment algorithms are:

- BIKE
- Classic McEliece
- CRYSTALS-KYBER
- FrodoKEM
- HQC
- LAC
- LEDAcrypt (merger of LEDAkem/LEDApkc)
- NewHope
- NTRU (merger of NTRUEncrypt/NTRU-HRSS-KEM)
- NTRU Prime
- NTS-KEM
- ROLLO (merger of LAKE/LOCKER/Ouroboros-R)
- Round5 (merger of Hila5/Round2)
- RQC
- SABER
- SIKE
- Three Bears

The 9 Second Round Candidates for digital signatures are:

- CRYSTALS-DILITHIUM
- FALCON
- GeMSS
- LUOV
- MQDSS
- Picnic
- qTESLA
- Rainbow
- SPHINCS+

https://csrc.nist.gov/news/2019/pqc-standardization-process-2nd-round-candidates

じゃあ軽量暗号はTLSに載る？

- 現時点では正直わからんけど…
  「最小実装」プロファイルとかなら？

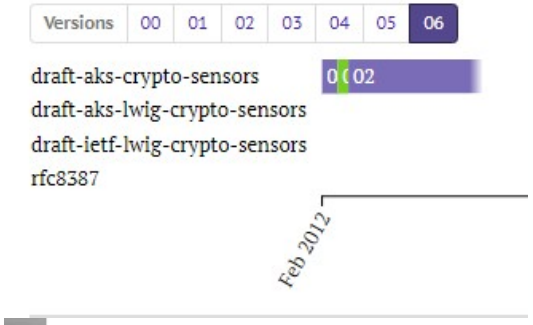
- Light-Weight Implementation Guidance (LWIG)

RFCs (4 hits)		
RFC 7228 *(was draft-ietf-lwig-terminology)* **Terminology for Constrained-Node Networks**	2014-05 17 pages	Informational RFC
RFC 7815 *(was draft-ietf-lwig-ikev2-minimal)* **Minimal Internet Key Exchange Version 2 (IKEv2) Initiator Implementation**	2016-03 41 pages	Informational RFC
RFC 8352 *(was draft-ietf-lwig-energy-efficient)* **Energy-Efficient Features of Internet of Things Protocols**	2018-04 24 pages	Informational RFC
RFC 8387 *(was draft-ietf-lwig-crypto-sensors)* **Practical Considerations and Implementation Experiences in Securing Smart Object Networks**	2018-05 33 pages	Informational RFC

あとはRFC8387は読もうかね

Practical Considerations and Implementation Experiences in Securing Smart Object Networks

RFC 8387



Light-Weight Implementation Guidance
Internet-Draft
Intended status: Informational
Expires: March 8, 2017

M. Sethi
J. Arkko
A. Keranen
Ericsson
H. Back
Comptel
September 4, 2016

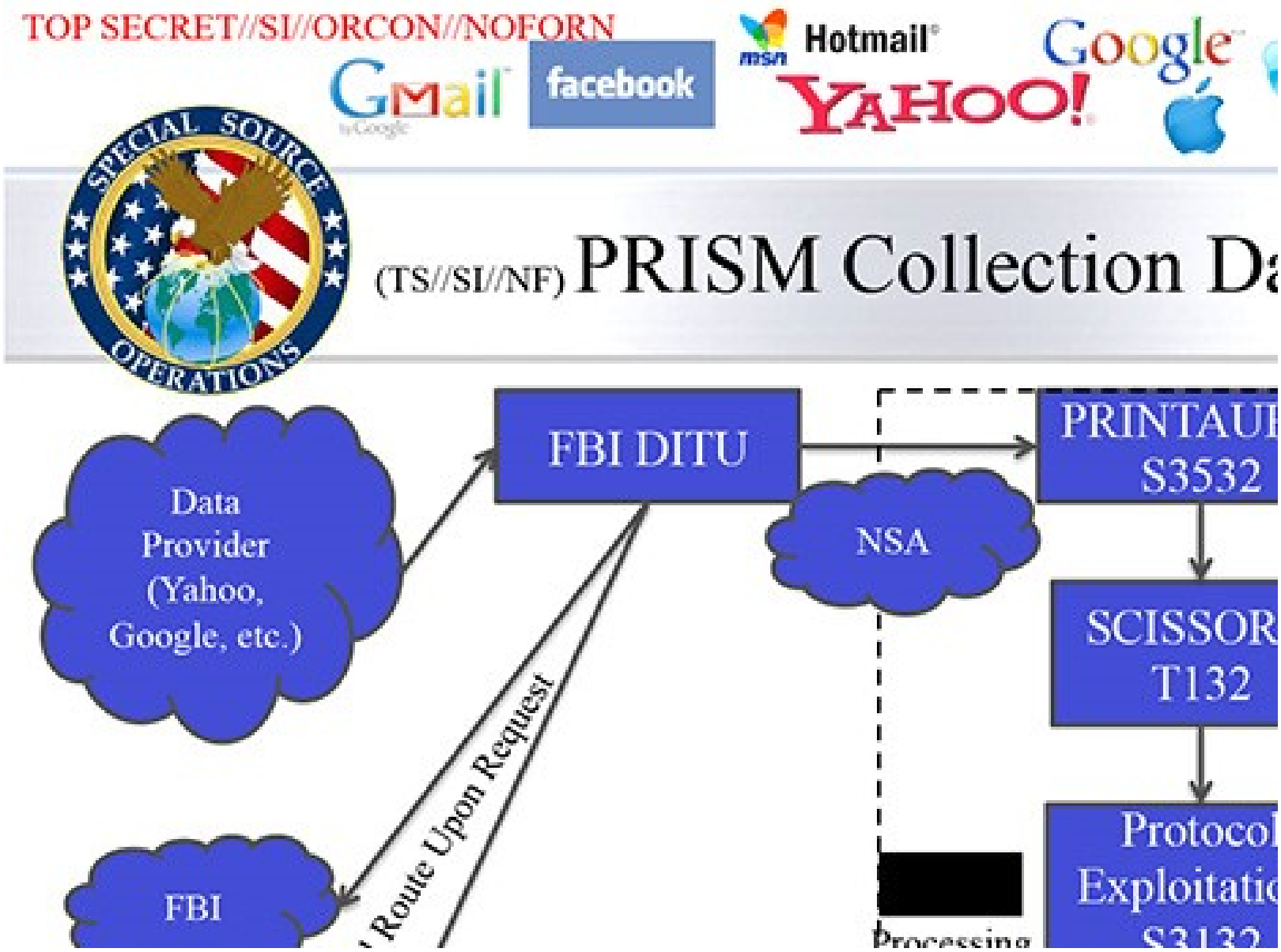
Practical Considerations and Implementation Experiences in Securing Smart Object Networks
draft-ietf-lwig-crypto-sensors-00

The implementation difficulties are important, but they should not be overemphasized. It is important to select the right security mechanisms and avoid duplicated or unnecessary functionality. But at the end of the day, if strong cryptographic security is needed, the implementations have to support that. Also, the use of the most lightweight algorithms and cryptographic primitives is useful, but should not be the only consideration in the design. Interoperability is also important, and often other parts of the system, such as key management protocols or certificate formats are heavier to implement than the algorithms themselves.

常時SSL（Always On SSL）への対応

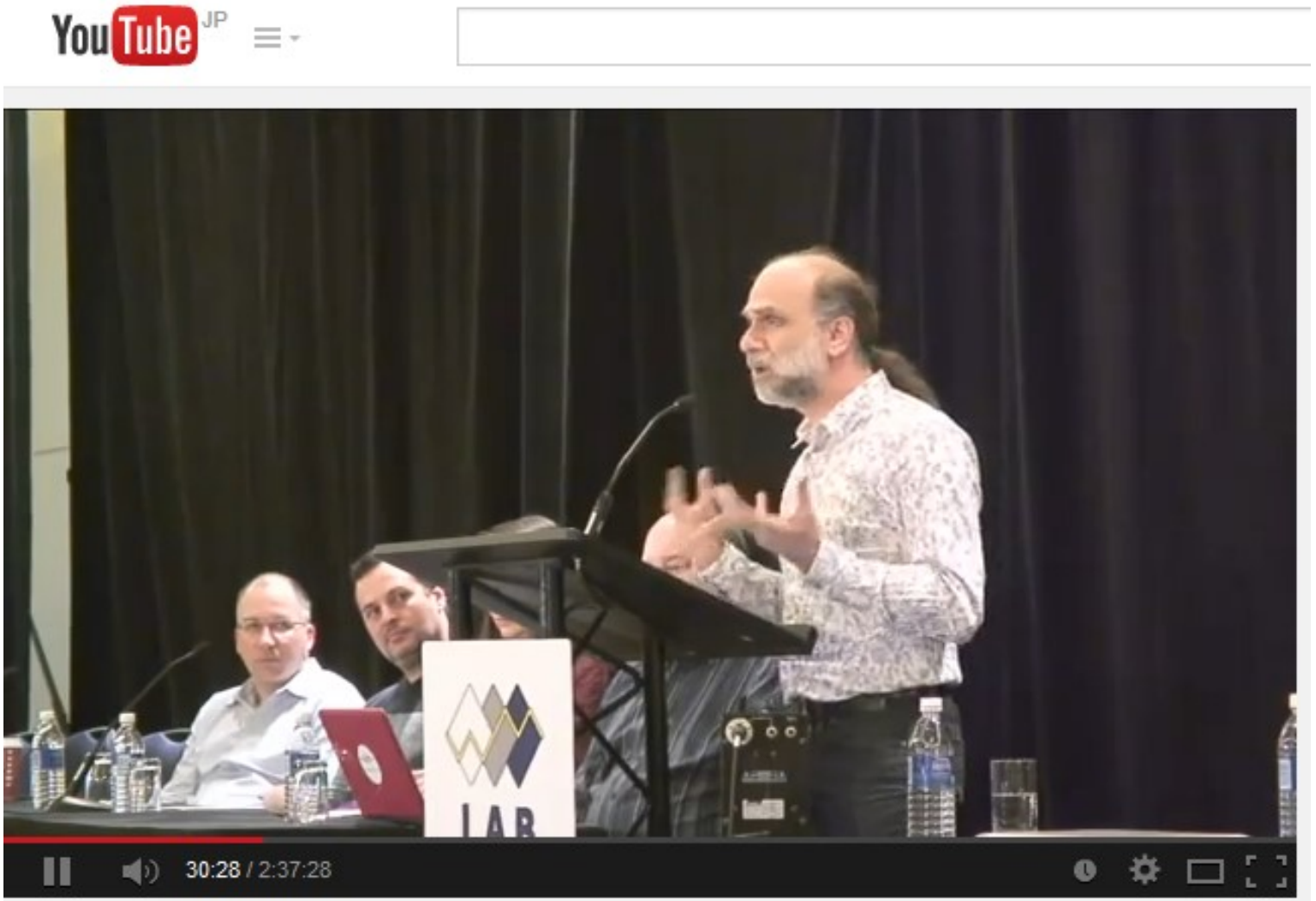
マーケティング用語？

PRISM



http://en.wikipedia.org/wiki/Global_surveillance_disclosures_(2013%E2%80%93present)

IETF 88 – Pervasive Surveillance



http://www.youtube.com/watch?v=oV71hhEpQ20

起点はおそらくスノーデン

- 全部暗号せな なにもかも抜かれてしまうで.
- せや！全ての通信を暗号化したらええやん.

- でも，それTLSじゃなくてよくね？
  - E2E的）IPsecなどの より下のレイヤでカバー
  - C2C的）S/MIME, GnuPGなどでカバー

日本がフルボッコにあったから？

（もしそうなら消極的では？）

USENIX Security '17 - Measuring HTTPS Adoption on the Web

358 回視聴

https://www.youtube.com/watch?v=toaR0uSabK8

（私見）無理しないでいいよ

suga@iij.ad.jp

私もかつてはSSL/TLS化すべきと考えている時期がありました.

[redacted]に遺恨を残しているという特殊な背景はありますので話半分で聞いて貰えればと.
私が正しい考えを持っているのかどうかも分からないです.

政治的な部分の問題が多いと思います．Symantec証明書排除などPKIのプレイヤー構造は大きく変化してしまいブラウザベンダーが強い意見を持つようになったと感じます．ブラウザベンダー様のいうことを聞かざるをえないのはおかしいんじゃない？とも.

・今回の移行計画は十分な余白を持って踏み切っているとは思えない（準備不足）
・HTTPS化した途端にHTTPコンテンツが残ったままのサイトが反応するのもまずい（Mixed Contentsエラーが出ても気が付かない）
・鍵管理のことわからんまま証明書導入している事例がたくさんありそう，いやある

最後の[redacted]が悪いと思う．クラウド証明書（＝複数の組織で秘密鍵を共有している証明書）が多すぎ．これブラウザで警告したりしないし，横の業者さんがHTTPSの中身見えてしまうリスクを受容しているとは到底思えない.
SnakeOil証明書（サンプルで公開されてた秘密鍵を利用）使っているところも実はまだある.

だから無理しないでいいよと言いたい.

特に日本はやり玉に挙げられてる（SSL/TLS対応比率が世界的に見て低い）のはなんとかしたい気持ちは分かるけど，徐々にでいいと考えてしまってます．少し消極的な意見ですかね.

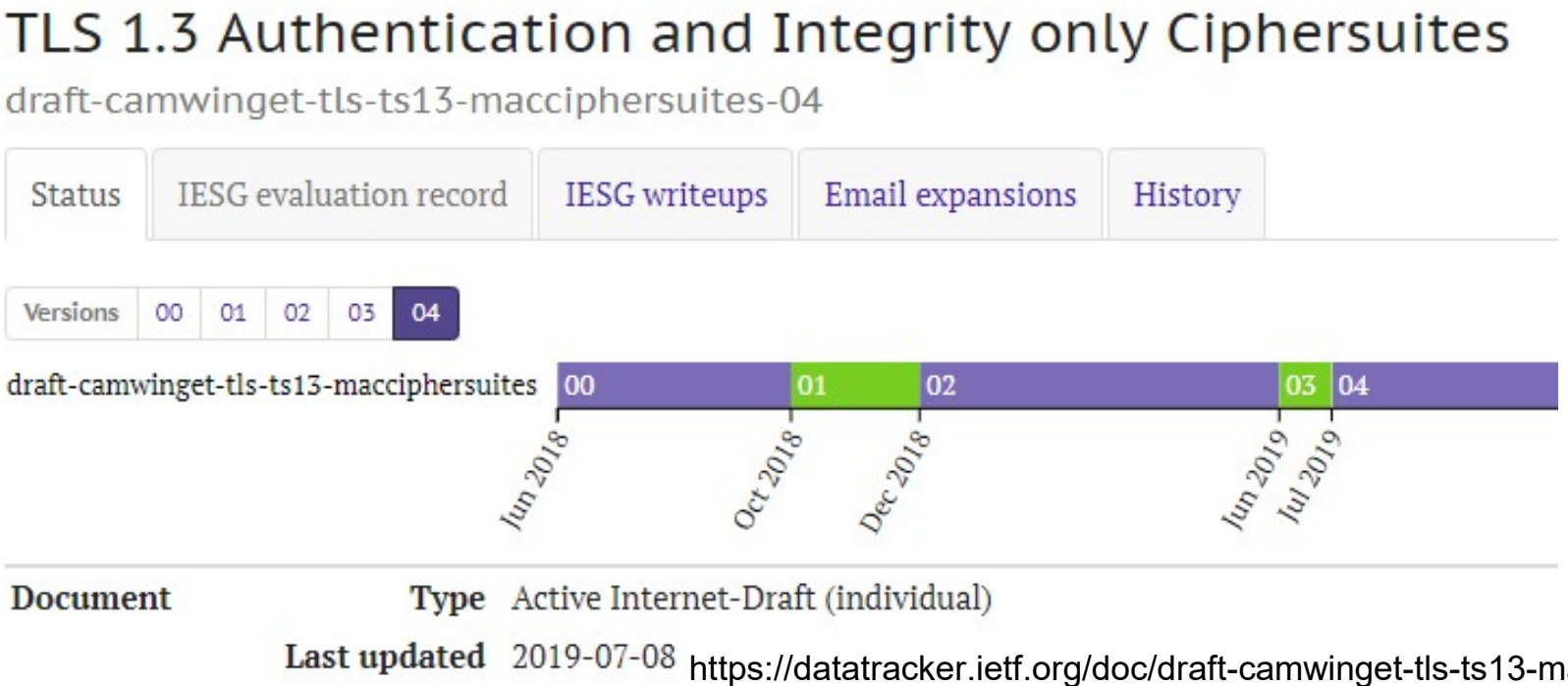
返信・編集・削除・いいね!・👍8 名がいいね! しています・2018/07/31

思うところは…

- 日本的な「しなやかさ」で対応できないか
  - 必要なところだけ使っているサイトが多い
    - そういう意味では数値は**実態**を表していない
    - つまりアンフェア
- 別観点ではサプライチェーンが信用できないのであれば「国産」であることも導入要因に？

TLS_*_anon_WITH_*

- 暗号化ONLY CipherSuitesの復活は？
  - TLS_DH_anon_WITH_CAMELLIA_256_CBC_SHA
- と思ったら同じこと考えてるヒト居た（嬉）



https://datatracker.ietf.org/doc/draft-camwinget-tls-ts13-macciphersuites

証明書・PKIとユーザインターフェイス

1-to-1ではない証明書の是非

- ワイルドカード証明書
  - ネットワークトポロジによっては管理コスト低下
- マンション証明書
  - 異なる組織のFQDNがSANに共存している
- IPアドレス証明書
  - SANに含めた証明書事例：1.1.1.1 (DNS界隈)

マンション証明書

Subject Alternative Name

- 異なるドメイン名がSANに共存している証明書

Subject DN	OU=Domain Control Validated, OU=PositiveSSL Multi-Domain, CN=sni178105.cloudflaressl.com
Issuer DN	C=GB, ST=Greater Manchester, L=Salford, O=COMODO CA Limited, CN=COMODO ECC Domain Validation Secure Server CA 2
Serial	374962007578661058562359913471005634O8
Validity	2018-02-22 00:00:00 **to** 2018-08-31 23:59:59 (190 days, 23:59:59)

VPSでお隣のVMに入居してる人の顔を知らない状況とよく似てる

https://censys.io/certificates/2843400699468ff9439c3ec89c3ee475d8d6ae9d1ca4107cad94d54b112c5e8f

VPS: Virtual private server

多様な接続形態・NWトポロジー

- VPSの利用
- ロードバランサによる負荷分散
  - 鍵の配備位置問題
  - 複数の鍵を別に置くことのリスク
- IoTなどミニマムセットでの利用

マンション証明書のリスク

- 異なる組織が同じ鍵ペアを利用している
  - 理論的にはお隣さんのTLSトラフィック読める？
    - SNIの仕組みとかVMの独立性に依存しそう
  - そもそも VPSベンダーが鍵生成してるから…
  - go.jp でもマンション証明書を保持してるFQDNあり
- 自分で調べたサービスは秘密鍵が
  ぶっこ抜けないようになってた
  - （いや，そこは本質じゃない）

リモート鍵とかHSMとかで解決？

- ネットワークトポロジで分類できそう
- リモート：鍵が必要になった時点で問い合わせ
  - レスポンスできる？
  - そもそもVPS借りてる意味は？
- 物理的なHSMを預ければいいのかい？
  - コスト高いわ，物理配送が必要で即時開通できん

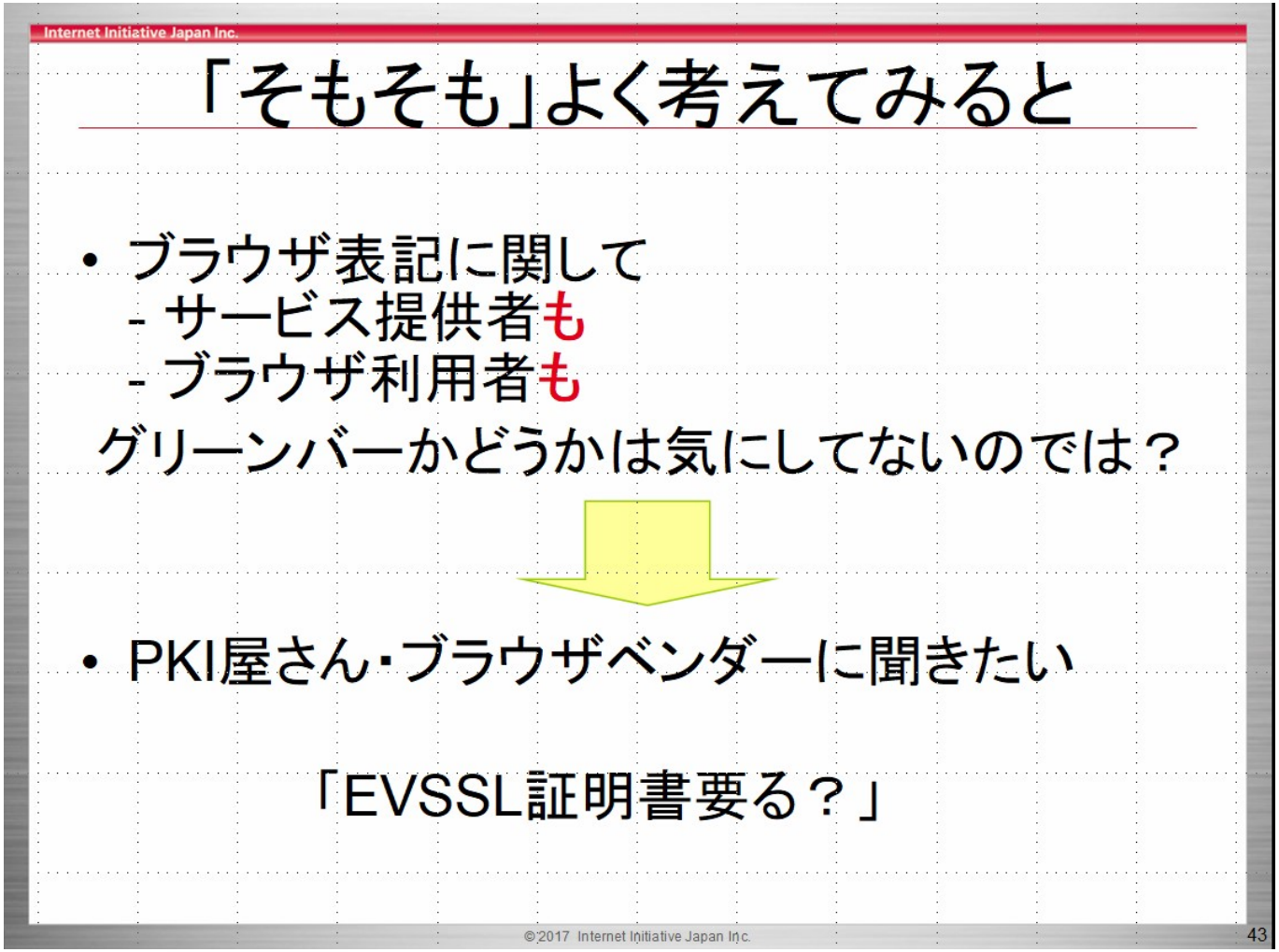
あらゆることが原因で EVSSL証明書が泣いている事例たち

- Mixed-Content
- 証明書OID処理バグ

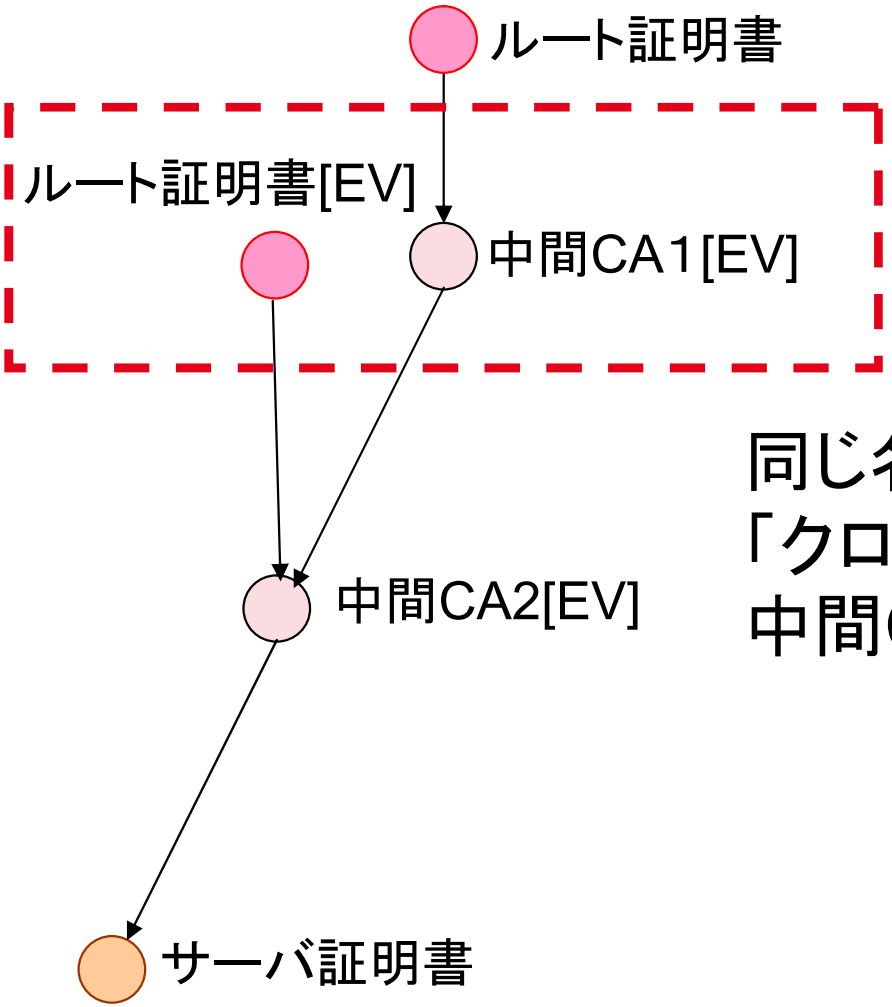
- クロスルート証明書 ← **new!**
  - フィーチャーフォン等のレガシー対応とも関連

EVSSL証明書の再考

- 2017年7月 Janog40 meeting の資料から
  - 当時はまだグリーンバーによるメリットがあった



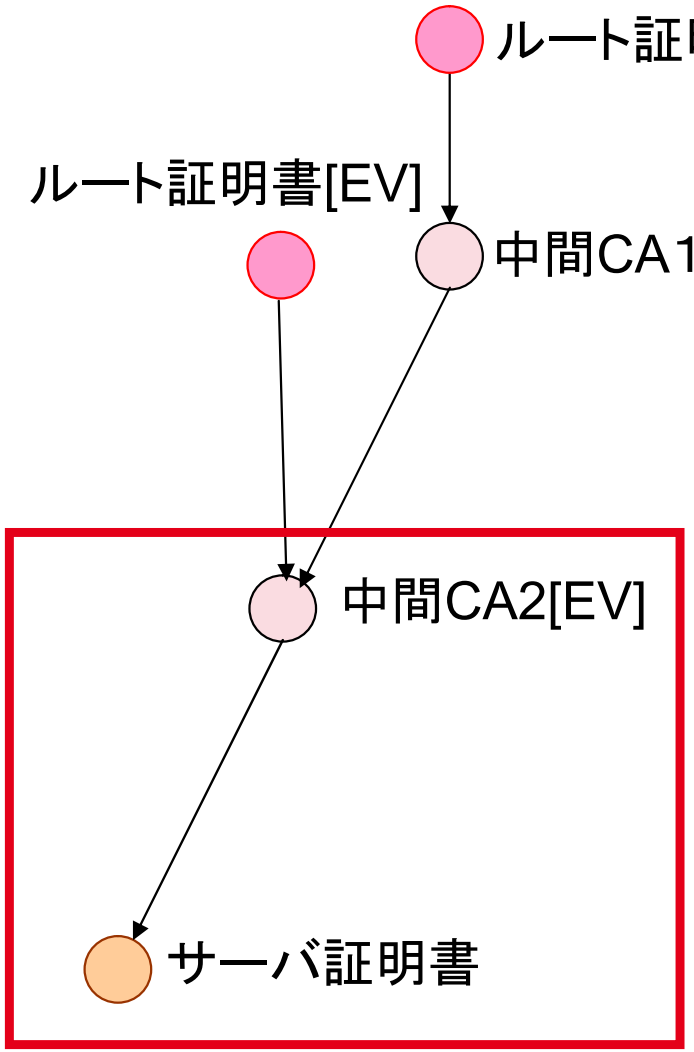
別のEVSSL証明書 不活性問題



ブラウザのトランスアンカーに
入っている

同じ名前でルート証明書でない
「クロスルート証明書」と呼ばれる
中間CA証明書(1)が存在するという状況

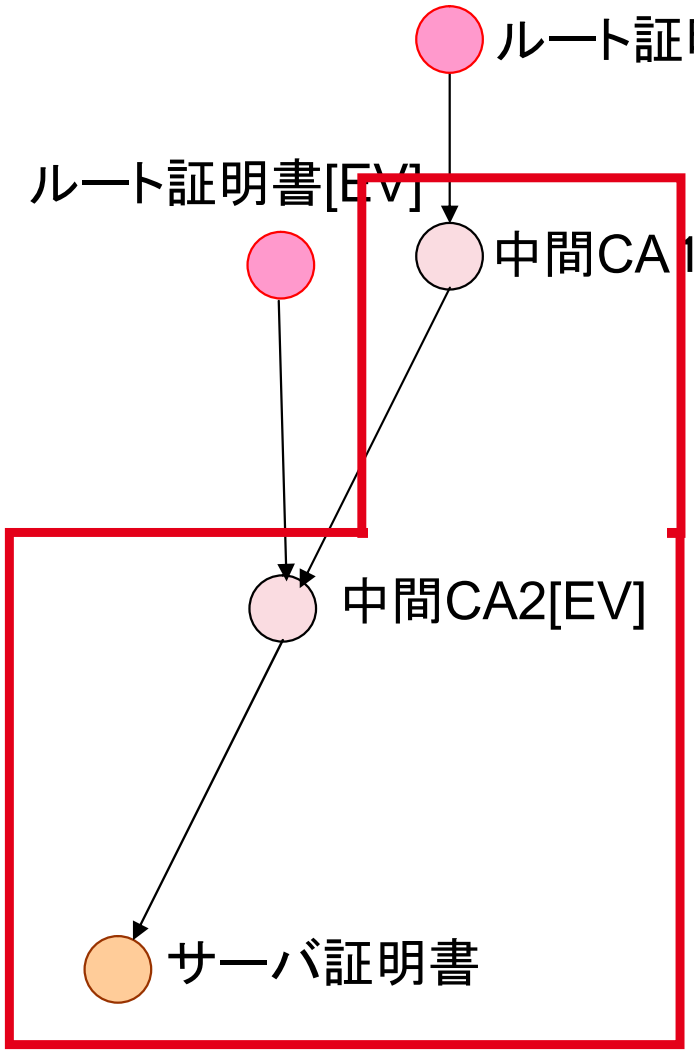
別のEVSSL証明書 不活性問題



ブラウザのトランスアンカーに
入っている

SSL/TLS接続した際にサーバから
この2枚の証明書を送りつける場合は
セーフ

別のEVSSL証明書 不活性問題

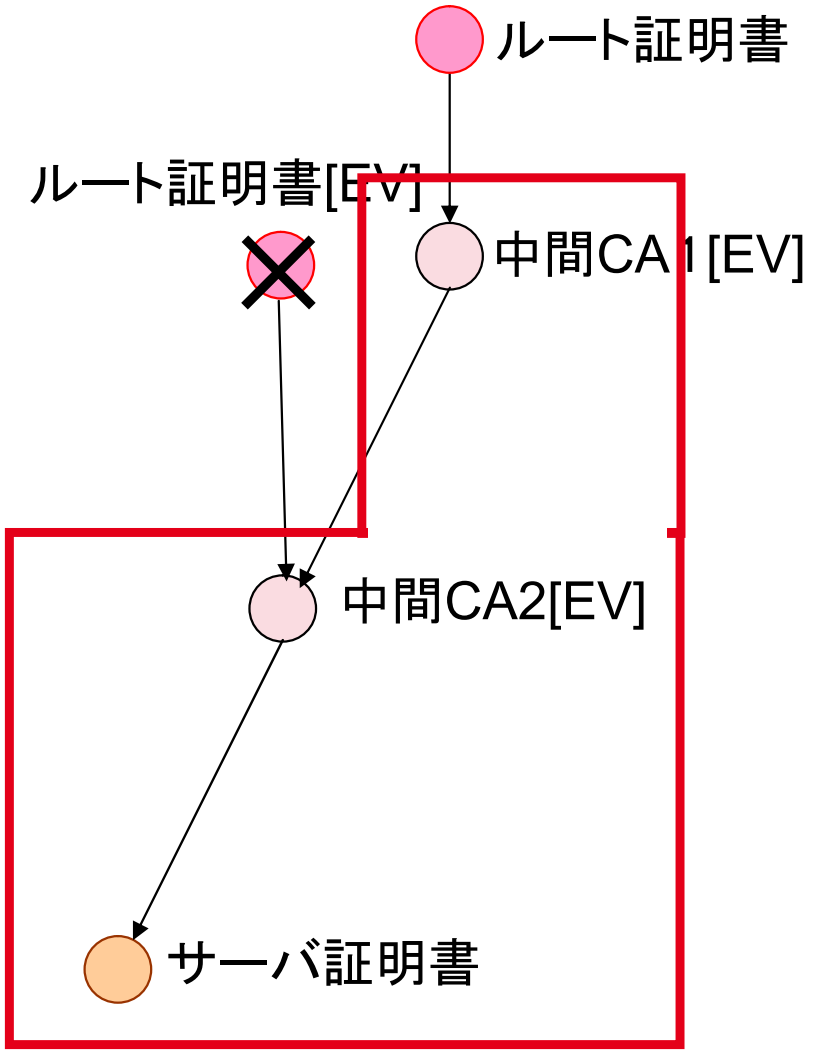


ブラウザのトランスアンカーに
入っている

SSL/TLS接続した際にサーバから
この3枚の証明書を送りつける場合は
アウト

本来な左の に
パスが伸びる実装が素直では？

これフィーチャーフォン対策でした



- 中間CA1を送らないとルート証明書までのパスを辿れない
- 泣く泣くEV不活性にしている実情

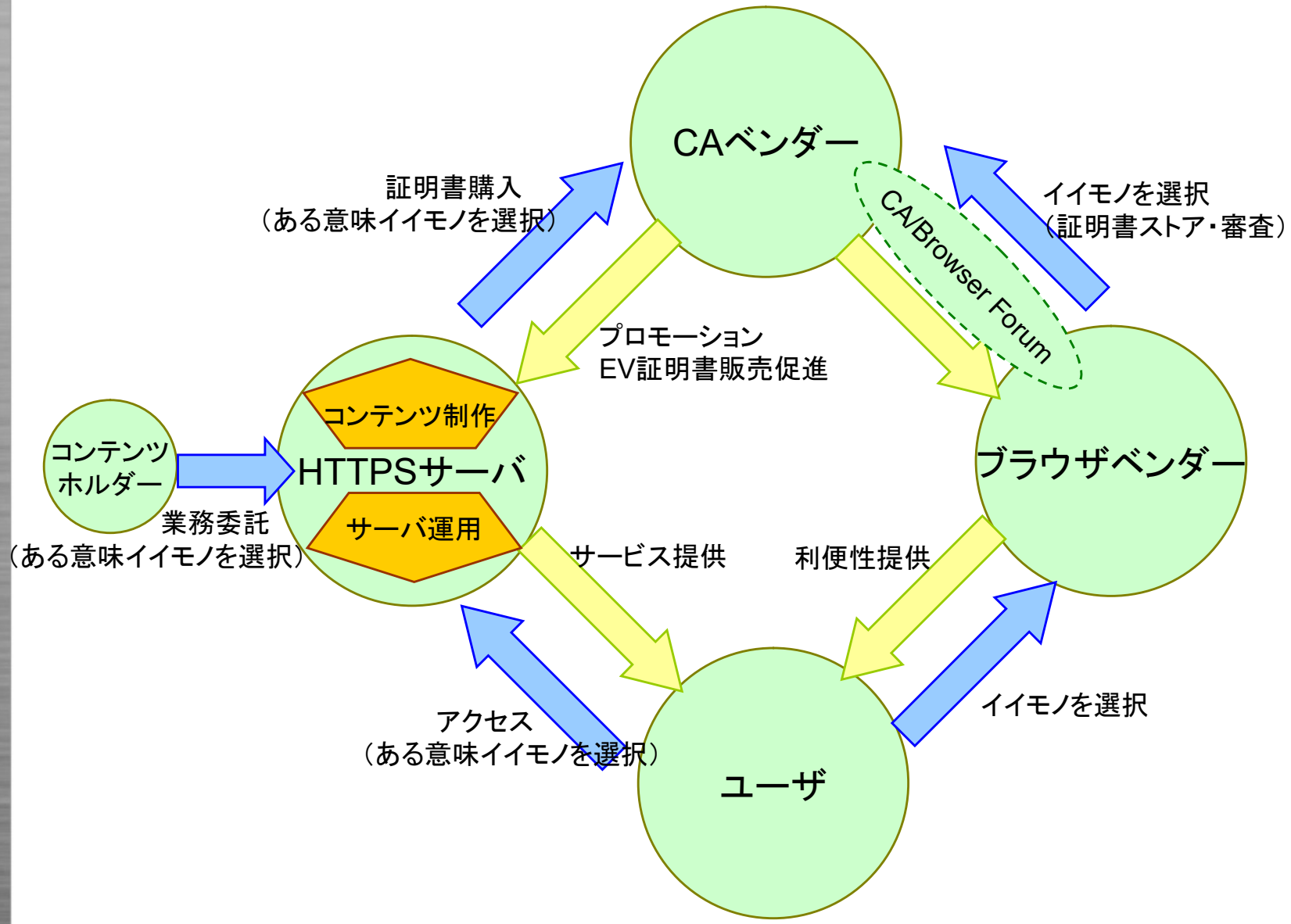
- ある業界：10 / 55

信頼点の変更による影響

信頼点って重要だと知らされた事例

- あんまりなにもいいません

ステークホルダーの関係



閉域での証明書利用の是非

閉域での証明書利用の是非

- 信頼点が突然変わるケース:
  基本的にはCAがポカしたとき

- 利用形態によってはプライベート証明書を許諾することもあるのだろうか？
- もう一つの観点:CTとプライバシ

Certificate Transparencyの課題

- CTの仕組みにより証明書発行すると
  FQDNがバレる
  - JANOG40で紹介した Censys 等で簡単に

  - https://censys.io/certificates?q=[念のため自主規制] [念のため自主規制] parsed.validity.start [念のため自主規制] 2019-07-01 [念のため自主規制] [念のため自主規制] ev

https://www.janog.gr.jp/meeting/janog40/program/censys

Predictions

POSITIVE

- Decentralized PKIの普及
  - https://github.com/WebOfTrustInfo/rwot1-sf/blob/master/draft-documents/Decentralized-Public-Key-Infrastructure-CURRENT.md
- Self-sovereign Identity (Management) から Self-sovereign Key Management へ
  - 暗号資産の鍵管理をユーザ自らの手に

NEGATIVE

- 正常系では動作するTLS1.3 Clientは
  - 相変わらず証明書検証無視
    - 新署名アルゴリズム対応でエラー処理時に起こりそう
  - TLS1.2 ClientHello extensionとの互換性確保が仇となる実装ミス
    - 同じ用語で異なる意味・エンコーディング
- DNS over TLS, DNS over HTTPS も結局信じられない世界が来るかも
  - 結局手元のツール・U/I をどう信じるか
- CT（Certificate Transparency）とプライバシ

TLS1.3 への移行に際して

TLS1.3実装状況

https://github.com/tlswg/tls13-spec/wiki/Implementations

name	language	role(s)	version	features/limitations
fizz	C++	C/S	-28	Based on libsodium, includes secure design abstractions. Zero-copy for advanced pe
NSS	C	C/S	RFC 8446	Almost everything, except post-handshake auth and X448
Mint	Go	C/S	-18	PSK resumption, 0-RTT, HRR
nqsb	OCaml	C/S	-11	PSK/DHE-PSK, no EC*, no client auth, no 0RTT -- live server at tls13test.nqsb.io por ping @hannesm, contains a static PSK/DHE_PSK token: id: 0x0000secret:
ProtoTLS	JavaScript	C/S	-13	EC/DHE/PSK, no HelloRetryRequest
miTLS	F*	C/S	RFC 8446	EC/DHE/PSK/0-RTT, no RSA-PSS, no post-HS-auth, no ESNI
Tris	Go	C/S	RFC 8446	ECDHE/PSK/0-RTT, no HelloRetryRequest
BoringSSL	C	C/S	-23, -28, RFC 8446	P-256, X25519, HelloRetryRequest, resumption, 0-RTT, KeyUpdate
Wireshark	C	other	-18 to -28, RFC 8446	Full decryption and dissection support for drafts 19-21 since 2.4.0 (keylog format). Su since 2.4.3, -23 since 2.4.5, -24 to -28 (+0RTT trial decryption) since 2.6.0. Tracking I
picotls	C	C/S	-18,-21,-23,-26	P-256, X25519, HelloRetryRequest, resumption, 0-RTT
rustls	Rust	C/S	-28 (final on branch)	P-256/P-384/curve25519, HRR, resumption, 0-RTT client
Haskell tls	Haskell	C/S	-28	ECDHE w/ P* and X*, full, HRR, PSK, 0RTT
Leto	C#	S	-18	DHE, X25519, AES, no PSK no 0RTT. Tested against NSS
OpenSSL	C	C/S	RFC 8446	P-256, P-384, P-521, X25519, X448, Ed25519, Ed448, HelloRetryRequest, resumpti stateless server, Post-handshake auth, KeyUpdate, RSA-PSS certs, no FFDHE
wolfSSL	C	C/S	-18/-22/-23/-26/-28	P-256, P-384, X25519, Ed25519, HelloRetryRequest, resumption, PSK, 0-RTT, CCS Post-Handshake Auth, KeyUpdate
GnuTLS	C	C/S	-28	P-256, P-384, X25519, FFDHE, RSA-PSS (keys and certs), HelloRetryRequest, Key
tlslite-ng	Python	C/S	RFC 8446	ECDHE (all), EdDHE (X25519, X448), FFDHE (all), AES-GCM, Chacha20, HelloRetr and certificate signatures, cookie extension, CCS, PSK, resumption, no ECDSA certi
tlsfuzzer	Python	C	RFC 8446	ECDHE (all), EdDHE (x25519, X448), FFDHE (all), AES-GCM, Chacha20, RSA, Hell

実装調査のポイント

- クライアント側（ブラウザ）の実装状況に依存した
  サーバの選択が必要

- ブラウザでは以下のようにOpenSSL, BoringSSL, NSS などの暗号ライブラリが内部で利用されている
  - Microsoft IE/Edge　Windows CNG (CAPI)
  - Chrome　BoringSSL
  - Firefox　NSS
  - Opera　Chromeベースでの実装に移行

TLS1.3ライブラリはRFC8446ではなくInternetDraft（中途）版も存在

- 先に挙げたライブラリ群は対応済のものが多いが，未実装の機能・アルゴリズムもあるので注意

BoringSSL	-23, -28, RFC 8446	P-256, X25519, HelloRetryRequest, resumption, 0-RTT, KeyUpdate
OpenSSL	RFC 8446	P-256, P-384, P-521, X25519, X448, Ed25519, Ed448, HelloRetryRequest, resumption, PSK, 0-RTT, CCS, cookies, stateless server, Post-handshake auth, KeyUpdate, RSA-PSS certs, no FFDHE
NSS	RFC 8446	Almost everything, except post-handshake auth and X448

参考：TLS1.3サーバ構築の一例

- 2017年5月 OpenSSLでの構築方法公開
  - https://www.openssl.org/blog/blog/2017/05/04/tlsv1.3/
  - 設定の注意点など細かい内容まで記載
- 2018月9月11日 OpenSSL1.1.1リリースで正式サポート
  - https://www.openssl.org/blog/blog/2018/09/11/release111/
- 2018年10月Apache 2.4.37 リリース mod_ssl で OpenSSL1.1.1対応
  - https://httpd.apache.org/download.cgi
  - enable-tls1_3 フラグでコンパイルすることで簡単に構築可能

参考：Apache2.4系での拡張ログ対応

https://httpd.apache.org/docs/current/mod/mod_ssl.html#logformats

- クライアントがどのバージョン，CipherSuitesで接続されたかを別のログファイルに追記

Custom Log Formats

When mod_ssl is built into Apache or at least loaded (under DSO situation) additional functions exist for the Custom Log Format {*varname*}x" eXtension format function which can be used to expand any variables provided by any module, especially those pro table.

For backward compatibility there is additionally a special ``%{*name*}c" cryptography format function provided. Information about th

Example

```
CustomLog "logs/ssl_request_log" "%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x \"%
```

These formats even work without setting the StdEnvVars option of the SSLOptions directive.

Environment Variables

This module can be configured to provide several iter default for performance reasons. (See SSLOptions available under different names, too. Look in the Com

Variable Name:	Value Type:
HTTPS	flag
SSL_PROTOCOL	string
SSL_SESSION_ID	string
SSL_SESSION_RESUMED	string
SSL_SECURE_RENEG	string
SSL_CIPHER	string
SSL_CIPHER_EXPORT	string
SSL_CIPHER_USEKEYSIZE	number
SSL_CIPHER_ALGKEYSIZE	number

その他様々な情報を残すことが可能→

まとめに代えて

- 実態に即したガイドライン発行を目指したい.
  そのためには多くの皆様のご意見を頂戴できればと思います.

- 本当に世の中で役立つアクティビティで
  ありたいので戦うときは戦うという姿勢で.

IIJ Internet Initiative Japan

Lead Initiative

日本のインターネットは1992年、IIJとともにはじまりました。以来、IIJグループはネットワーク社会の基盤をつくり、技術力でその発展を支えてきました。インターネットの未来を想い、新たなイノベーションに挑戦し続けていく。それは、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たIIJの、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/