Internet Week 2014
サイト管理者が知っておくべきSSLの秘孔(ツボ)

SSL/TLSで使われる暗号技術のキモチ

菅野 哲（かんの さとる）
ＮＴＴソフトウェア株式会社
２０１４年１１月２０日

この人は誰？

- **名前**
  - 菅野 哲（かんの さとる）
- **所属**
  - NTTソフトウェア 株式会社
    - セキュリティ事業部
    - 人事部　人材開発部門 兼務
    - マーケティングやプロモーションも・・・
- **その他の所属**
  - ISOC Japan Chapter（Program Committee）
  - MIT-KIT Japan Chapter（窓口？）

本講演の目的

- **日頃のサーバ管理でギモンに思っていることを解消するためのきっかけに！**
  - 暗号って種類が多くて面倒ですよね？
  - なぜこの設定なのか？判断する足掛かりがほしい
  - 専門用語が意味わからない・・・

日頃のギモンとして・・・

暗号って
どれもいっしょでしょ？

安全な暗号って？

脆弱性対応って？

危殆化対策？

〇〇ビット
セキュリティ？

みんなのSSL/TLSと脅威

SSL/TLSが利用されるまで様々な工程がある

現実的なもの

サービス／システム

暗号製品

SSL/TLSプロトコル

暗号技術

理想的なもの

実装物のバグ

仕様上の欠陥

危殆化

SSL/TLSにおける暗号技術

- 認証や経路上を暗号化するために，複数の暗号技術を組合せで実現
  - **Ciphersuite**で組合せがわかる！

具体例：

0x00,0xBA	TLS_RSA_WITH_CAMELLIA_128_CBC_SHA256	Y	[RFC5932]
0x00,0xBB	TLS_DH_DSS_WITH_CAMELLIA_128_CBC_SHA256	Y	[RFC5932]
0x00,0xBC	TLS_DH_RSA_WITH_CAMELLIA_128_CBC_SHA256	Y	[RFC5932]
0x00,0xBD	TLS_DHE_DSS_WITH_CAMELLIA_128_CBC_SHA256	Y	[RFC5932]
0x00,0xBE	TLS_DHE_RSA_WITH_CAMELLIA_128_CBC_SHA256	Y	[RFC5932]
0x00,0xC2	TLS_DH_RSA_WITH_CAMELLIA_256_CBC_SHA256	Y	[RFC5932]
0x00,0xC6-FE	Unassigned		
0x00,0xFF	TLS_EMPTY_RENEGOTIATION_INFO_SCSV	Y	[RFC5746]
0x01-BF,*	Unassigned		
0xC0,0x01	TLS_ECDH_ECDSA_WITH_NULL_SHA	Y	[RFC4492]
0xC0,0x02	TLS_ECDH_ECDSA_WITH_RC4_128_SHA	N	[RFC4492][RFC6347]
0xC0,0x03	TLS_ECDH_ECDSA_WITH_3DES_EDE_CBC_SHA	Y	[RFC4492]
0xC0,0x04	TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA	Y	[RFC4492]
0xC0,0x05	TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA	Y	[RFC4492]

鍵交換　暗号利用モード

TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA256

認証　共通鍵　MAC

SSL/TLSで利用してる暗号技術：公開鍵暗号

- **どんな特徴？**
  - 異なる一対の鍵を駆使して暗号化／復号（または，署名／検証）を行う
  - 公開できる鍵がある
  - 処理速度は低速
- **SSL/TLSでは何が使えるの？**
  - 鍵交換
    - RSA，DH，ECDH
  - 認証
    - RSA，ECDSA
- **安全性を考えるときの着眼点**
  - 十分な鍵長かどうか？

SSL/TLSで利用してる暗号技術：共通鍵暗号

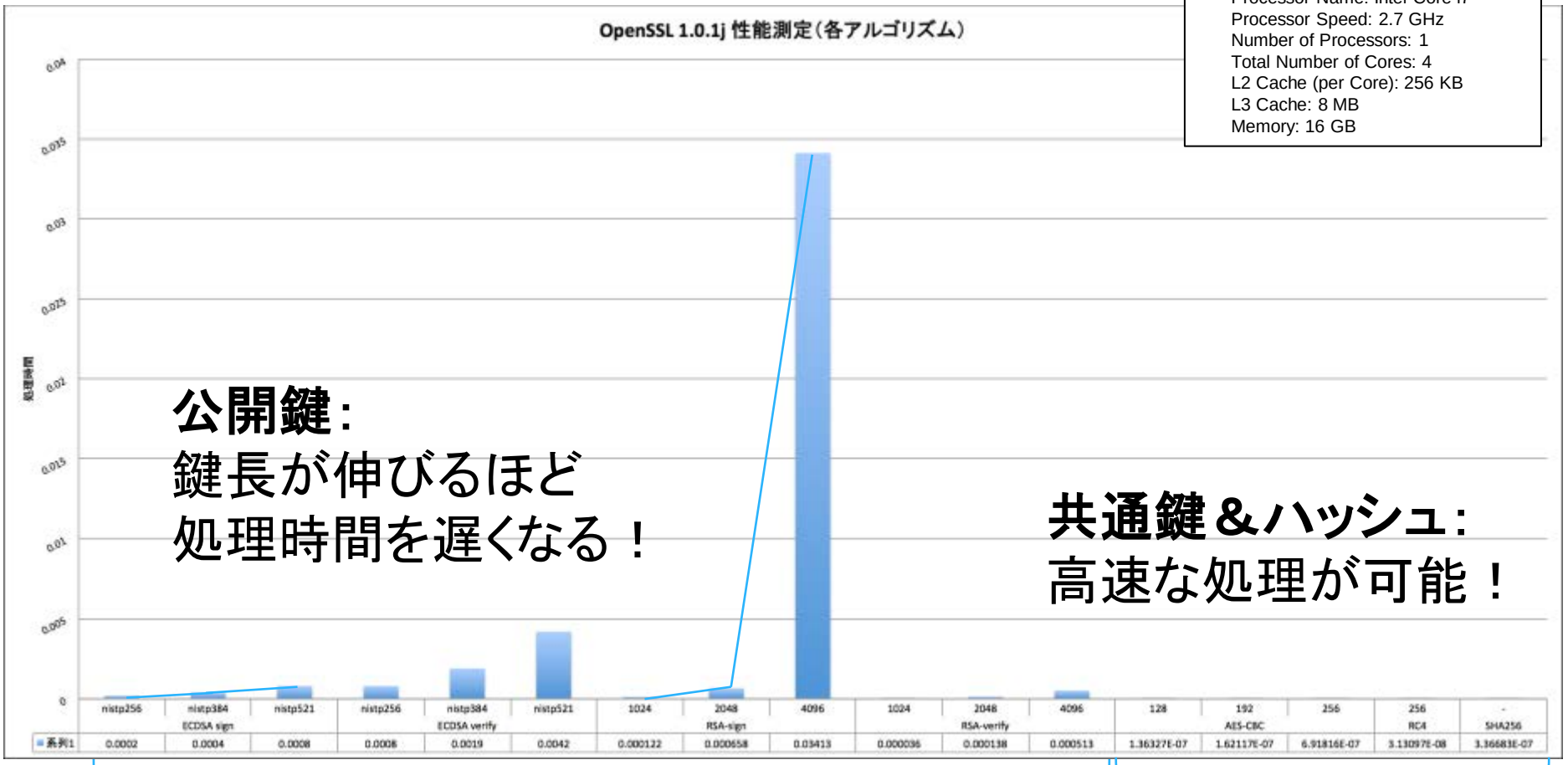
- **どんな特徴？**
  - 共通な鍵を用いて暗号化／復号を行う
  - 処理処理は高速
- **SSL/TLSでは何が使えるの？**
  - 共通鍵暗号
    - DES，3DES，AES，Camelliaなど
  - 暗号利用モード
    - CBC，GCMなど
- **安全性を考えるときの着眼点！**
  - 危殆化したアルゴリズムじゃないか？
  - 十分な鍵長かどうか？

SSL/TLSで利用してる暗号技術：ハッシュ関数

- **どんな特徴？**
  - 任意の長さを固定長のデータに変換できる
  - MACやデジタル署名に利用する
- **SSL/TLSでは何が使えるの？**
  - MD5，SHA-1，SHA-256など
- **安全性を考えるときの着眼点！**
  - 危殆化しているアルゴリズムを利用していないか？
  - どんな用途で使うのか？
    - デジタル署名？　HMAC？

性能で見るSSL/TLSで利用される暗号技術

測定環境:
Model Name: MacBook Pro
Model Identifier: MacBookPro10,1
Processor Name: Intel Core i7
Processor Speed: 2.7 GHz
Number of Processors: 1
Total Number of Cores: 4
L2 Cache (per Core): 256 KB
L3 Cache: 8 MB
Memory: 16 GB



公開鍵:
鍵長が伸びるほど
処理時間を遅くなる！

共通鍵＆ハッシュ:
高速な処理が可能！

公開鍵暗号

共通鍵＆ハッシュ

暗号技術の安全性：鍵長的な側面（1/2）

- 「 RSA-1024はAES-128 はどっちが安全？ 」という質問をよく受ける・・・

ECDH-P256

ECDSA-P521

Camellia-256

SHA-256

RSA-1024

AES-128

DH-1024

異なる鍵長をどう判断するの？

「等価安全性」という概念！

暗号技術の安全性：鍵長的な側面（2/2）

- 等価安全性とは？
  - 異なる種類の暗号技術に対して，同一の評価尺度で安全性を表したもの
  - 例： 80 bit セキュリティ
    - 解読に必要な計算量が2^80乗相当である

・・・と言われても・・・わかりにくし
どうやって判定するの？？

等価安全性に関するお助けサイト

BlueKrypt | Cryptographic Key Length Recommendation

In most cryptographic functions, the key length is an important security parameter. Both academic and private organizations provide recommendations and mathematical formulas to approximate the minimum key size requirement for security. Despite the availability of these publications, choosing an appropriate key size to protect your system from attacks remains a headache as you need to read and understand all these papers.

This web site implements mathematical formulas and summarizes reports from well-known organizations allowing you to quickly evaluate the minimum security requirements for your system. You can also easily compare all these techniques and find the appropriate key length for your desired level of protection. The lengths provided here are designed to resist mathematic attacks; they do not take algorithmic attacks, hardware flaws, etc. into account.

Choose a Method

- Lenstra and Verheul Equations (2000)
- Lenstra Updated Equations (2004)
- ECRYPT II Recommendations (2012)
- NIST Recommendations (2012)
- ANSSI Recommendations (2010)
- Fact Sheet NSA Suite B Cryptography (2013)
- etwork Working Group RFC3766 (2004)
- BSI Recommendations (2014)

Compare all Methods

his kind authorization and comments.
o Law Survey / Digital Signature Law Survey.

Copyright | Release Notes

Date	Minimum of Strength	Symmetric Algorithms	Asymmetric	Discrete Logarithm Key	Discrete Logarithm Group	Elliptic Curve	Hash (A)	Hash (B)
2010 (Legacy)	80	2TDEA*	1024	160	1024	160	SHA-1** SHA-224 SHA-256 SHA-384 SHA-512	SHA-1 SHA-224 SHA-256 SHA-384 SHA-512
2011 - 2030	112	3TDEA	2048	224	2048	224	SHA-224 SHA-256 SHA-384 SHA-512	SHA-1 SHA-224 SHA-256 SHA-384 SHA-512
> 2030	128	AES-128	3072	256	3072	256	SHA-256 SHA-384 SHA-512	SHA-1 SHA-224 SHA-256 SHA-384 SHA-512
>> 2030	192	AES-192	7680	384	7680	384	SHA-384 SHA-512	SHA-224 SHA-256 SHA-384 SHA-512
>>> 2030	256	AES-256	15360	512	15360	512	SHA-512	SHA-256 SHA-384 SHA-512

http://www.keylength.com/

危殆化対策

- 危殆化とは
  - 当初期待していた安全性を担保できない状態になっていること

でも，忙しくてそんな情報を収集してられないよ・・・

危殆化対策を行う際の情報源

- 正直，日々の運用を実施していると暗号アルゴリズムやプロトコルの状況監視している余裕がない．．．

暗号アルゴリズムと**暗号プロトコル**に関する監視／評価している組織があり，信頼のおける情報を発信している

暗号アルゴリズム	暗号プロトコル
CRYPTREC	CELLOS

暗号アルゴリズムに関する有益な情報

- 日本にはCRYPTRECという暗号技術の評価／監視している組織がある

http://www.cryptrec.go.jp/

安全なアルゴリズム

危険なアルゴリズム

暗号プロトコルに関する有益な情報

- 信頼できるプロトコルに関する脆弱性情報などを公開してくれるサイトってないの？

暗号プロトコルの脆弱性の解説を公開！

https://www.cellos-consortium.org/jp/

SSL/TLSにおける禁じ手？

- CRYPTRECやCELLOSで情報監視していれば万全なの？

SSL/TLSの利用の観点からだと不十分・・・

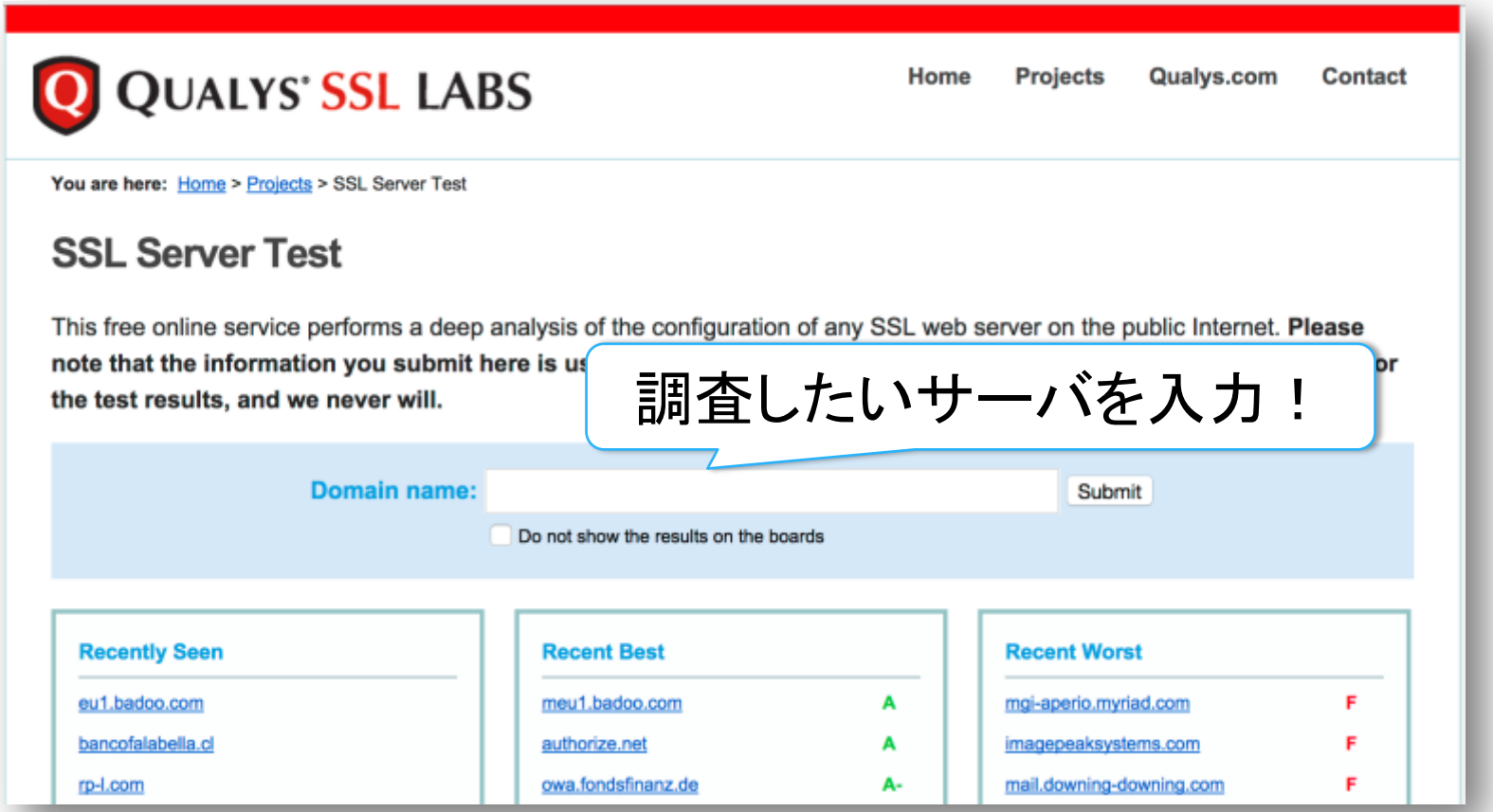
過去の遺産三兄弟

古い環境だと**デフォルト**で**ON**になっている

知っていると便利かもしれない情報

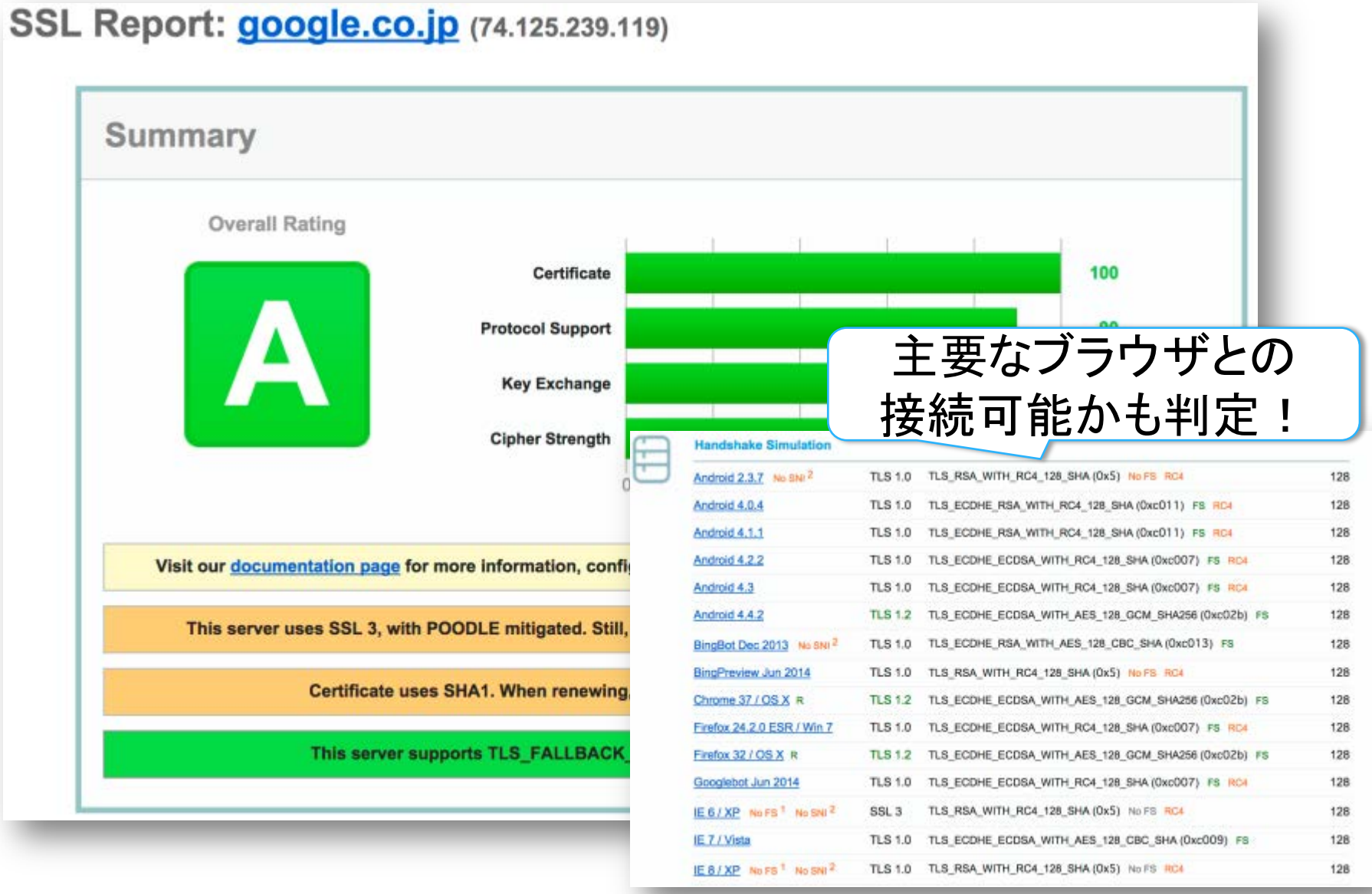
自分のサーバ設定を確かめるには・・・？

- **QUALYS SSL LABS**



https://www.ssllabs.com/ssltest/index.html

手軽に調査できちゃう！



日頃のギモン が解決？！

まとめ

- Ciphersuiteの構成を読み解くことができる！
  - SSL/TLSで利用される暗号技術
- サーバ設定時に耳にする用語を把握した！
- デフォルト設定だと危険なことがある！という事実
- 安全にSSL/TLSを利用するには暗号～システム構築まで広い範囲で注意が必要！
- 情報収集に関する便利なサイト
  - ドンドン活用して負荷を軽減！

https://www.cellos-consortium.org/jp/

CONTACT

- **E-mail**
  - kanno.satoru@po.ntts.co.jp
- **SNS**
  - Twitter (satorukanno)
  - Facebook (satoru.kanno)
  - Linkedin

お気軽にご連絡ください！