楕円曲線暗号におけるPKI

2011年 9月26日

筑波大学　金岡 晃

PKIにおける公開鍵暗号方式

- 公開鍵暗号と言えばRSA暗号
  - 公開鍵暗号の利用されているシーンでは、現在ほぼすべてRSA暗号が使われていると言って良い
  - RSA暗号で使われる鍵のサイズは、現在1024ビットや2048ビットが主流である
- 楕円曲線暗号（ECC：Elliptic Curve Cryptography）
  - 楕円曲線暗号は、楕円曲線利用し、曲線上の点の演算により定義される暗号方式の総称である
  - 楕円曲線上でDiffie-Hellman（DH）鍵共有を行う楕円DH（ECDH）方式や、楕円曲線上でDigital Signature Algorithm（DSA）を実現する楕円DSA（ECDSA）方式などがある。
    - その他：ECMQV、ECIESなど
  - RSA暗号と比較し、鍵サイズが小さいことが特長であり、ポストRSA暗号として注目されている

楕円曲線暗号整備の背景

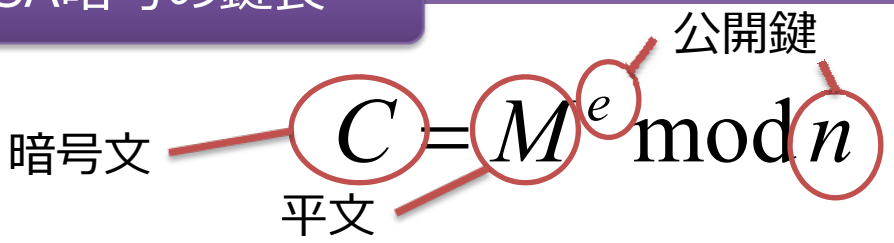
- NSA Suite B
  - 2005年、米国家安全保障局（NSA）は機密情報の保護に利用される暗号アルゴリズムのセットSuite Bを発表した
  - 公開鍵暗号のアルゴリズムにRSA暗号はなく、鍵交換はECDH（256または384ビット素体）、電子署名はECDSA（256ビットまたは384ビット素体）が指定されている
    - 「256ビット」「素体」などの種別については後述
  - その後、Suite Bに合わせた仕様が策定
    - RFC 4869、5430、5008等
  - Suite Bの実装ガイド（ECDH版, ECDSA版）も公開
    - NSA Suite B Implementers' Guide to NIST SP 800-56A
    - NSA Suite B Implementers' Guide to FIPS 186-3 (ECDSA) - February 2010

NSA Suite B Implementer ‘s Guide to FIPS 186-3

- 2010年2月3日発行
- Suite Bに入っているECDSAの実装ガイド
  - NIST FIPS 186-3に定められているECDSAを中心に
- Suite BのECDSA実装に必要となる仕様がそれぞれ抜粋し構成されている
  - ECDSA仕様のうちSuite Bに関するもの
    - P-256とP-384の2つのパラメータ
  - ECDSAアルゴリズムそのもの
    - ANS X9.62
  - 公開鍵の検証
    - NIST SP 800-56A

楕円曲線暗号のパラメータ

RSA暗号の鍵長



nのサイズ＝鍵長

楕円曲線暗号の鍵長

（素数qで決められる素体の場合）

楕円曲線 $\mathrm{E}: \mathrm{y}^2 \equiv x^3 + ax + b \bmod q$

を満たす素体上の点（有理点）の数（位数） $\#E$

$\#E$ のサイズ＝鍵長 → $\#E$ は a, b, q で決まる

RSAと違って、鍵長を決めるのは1つの値（パラメータ）だけではない

→ **パラメータのセットが複数存在**

パラメータの種類：NIST推奨パラメータ群

- 楕円曲線のパラメータの表現
  - 体の構成（素体 Prime Fields,標数2の体 Binary Fields）、サイズ、曲線
- NIST（米国立標準技術研究所）推奨パラメータ
  - FIPS 186-3 Appendix D
  - ちなみにECDSA 。ECDHはない。

Curves over Prime Fields	
P-192	P-384
P-224	P-521
P-256	

Curves over Binary Fields			
K-163	B-233	K-409	K-571
B-163	K-283	B-409	B-571
K-233	B-283		

＊Pは素体、B,Kは標数2の体。KはKoblitz曲線

パラメータの種類：SECG推奨パラメータ群

- SEC2: Recommended Elliptic Curve Domain Parameters
  - SECG（The Standards for Efficent Cryptography）：楕円曲線暗号の標準仕様策定を目指すコンソーシアム
    - www.secg.org/download/aid-784/sec2-v2.pdf

Curves over Prime Fileds
secp192k1
secp192r1
secp224k1
secp224r1
secp256k1
secp256r1
secp384r1
secp521r1

Curves over Binary Fields	
sect163k1	sect283k1
sect163r1	sect283r1
sect163r2	sect409k1
sect233k1	sect409r1
sect233r1	sect571k1
sect239k1	sect571r1

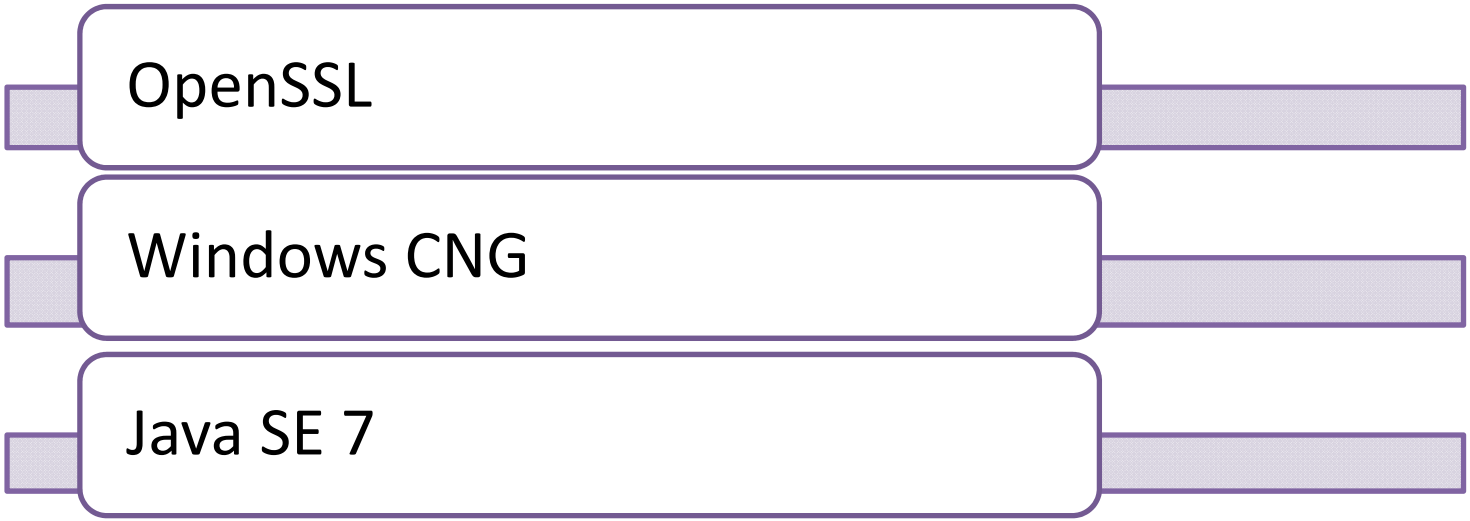
RFC 4492によるマッピング

SECG	ANSI X9.62	NIST
sect163k1		NIST K-163
sect163r1		
sect163r2		NIST B-163
sect193r1		
sect193r2		
sect233k1		NIST K-233
sect233r1		NIST B-233
sect239k1		
sect283k1		NIST K-283
sect283r1		NIST B-283
sect409k1		NIST K-409
sect409r1		NIST B-409
sect571k1		NIST K-571
sect571r1		NIST B-571
secp160k1		
secp160r1		
secp160r2		
secp192k1		
secp192r1	prime192v1	NIST P-192
secp224k1		
secp224r1		NIST P-224
secp256k1		
secp256r1	prime256v1	NIST P-256
secp384r1		NIST P-384
secp521r1		NIST P-521

各プラットフォームの対応状況

対応状況の調査

- 各種暗号ライブラリなどのプラットフォームで実際にECCは利用できるのか
- ECCを使ったPKIが利用可能なのか
  - 以下のプラットフォームで調査



OpenSSL

- OpenSSL 1.0.0よりECCがデフォルトで利用可能
  - 2010年3月29日にリリース
  - ECC自体の鍵生成や証明書発行は0.9.8でも可能だった
  - ECC関連の暗号スイートはALLで呼んでもリストアップされず”ECCdraft”を付ける必要があった
  - 1.0.0からはALLで呼ばれるようになり、ECCdraftはなくなった

利用可能なパラメータ種類：OpenSSL

- OpenSSLは67種類
  - 表示コマンド：openssl ecparam -list_curves

secp112r1	secp384r1	sect131r1	sect283r1	c2tnb191v3	wap-wsg-idm-ecid-wtls1
secp112r2	secp521r1	sect131r2	sect409k1	c2pnb208w1	wap-wsg-idm-ecid-wtls3
secp128r1	prime192v1	sect163k1	sect409r1	c2tnb239v1	wap-wsg-idm-ecid-wtls4
secp128r2	prime192v2	sect163r1	sect571k1	c2tnb239v2	wap-wsg-idm-ecid-wtls5
secp160k1	prime192v3	sect163r2	sect571r1	c2tnb239v3	wap-wsg-idm-ecid-wtls6
secp160r1	prime239v1	sect193r1	c2pnb163v1	c2pnb272w1	wap-wsg-idm-ecid-wtls7
secp160r2	prime239v2	sect193r2	c2pnb163v2	c2pnb304w1	wap-wsg-idm-ecid-wtls8
secp192k1	prime239v3	sect233k1	c2pnb163v3	c2tnb359v1	wap-wsg-idm-ecid-wtls9
secp224k1	prime256v1	sect233r1	c2pnb176v1	c2pnb368w1	wap-wsg-idm-ecid-wtls10
secp224r1	sect113r1	sect239k1	c2tnb191v1	c2tnb431r1	wap-wsg-idm-ecid-wtls11
secp256k1	sect113r2	sect283k1	c2tnb191v2		wap-wsg-idm-ecid-wtls12
					Oakley-EC2N-3
					Oakley-EC2N-4

Windows CNG：

- Vistaから導入されているWindows CNG（Cryptography Next Generation）で対応
- 利用可能なパラメータは3種類
  - NISTパラメータのP-256、P-384、P-521

P-256
P-384
P-521

} NSA Suite Bで規定されているパラメータ

Java SE 7

- Java SE 7より楕円曲線を利用した暗号アルゴリズムに対応
  - 楕円曲線暗号用のプロバイダであるSunECプロバイダが加わった。
  - 対応アルゴリズム：ECIES、 ECDH、ECMQV、ECDSA
- 利用可能パラメータは46種類

secp112r1	secp256r1,NIST P-256,prime256v1	sect163k1,NIST K-163	sect571k1,NIST K-571
secp112r2	secp384r1,NIST P-384	sect163r1	sect571r1,NIST B-571
secp128r1	secp521r1,NIST P-521	sect163r2,NIST B-163	c2tnb191v1
secp128r2	prime192v2	sect193r1	c2tnb191v2
secp160k1	prime192v3	sect193r2	c2tnb191v3
secp160r1	prime239v1	sect233k1,NIST K-233	c2tnb239v1
secp160r2	prime239v2	sect233r1,NIST B-233	c2tnb239v2
secp192k1	prime239v3	sect239k1	c2tnb239v3
secp192r1,NIST P-192,prime192v1	sect113r1	sect283k1,NIST K-283	c2tnb359v1
secp224k1	sect113r2	sect283r1,NIST B-283	c2tnb431r1
secp224r1,NIST P-224	sect131r1	sect409k1,NIST K-409	
secp256k1	sect131r2	sect409r1,NIST B-409	

パラメータ対応表：OpenSSL、Windows CNG、Java SE 7

openssl	Windows CNG	Java SE 7
secp112r1		secp112r1
secp112r2		secp112r2
secp128r1		secp128r1
secp128r2		secp128r2
secp160k1		secp160k1
secp160r1		secp160r1
secp160r2		secp160r2
secp192k1		secp192k1
secp224k1		secp224k1
secp224r1		secp224r1,NIST P-224
secp256k1		secp256k1
secp384r1	P-384	secp384r1,NIST P-384
secp521r1	P-521	secp521r1,NIST P-521
prime192v1		secp192r1,NIST P-192,prime192v1
prime192v2		prime192v2
prime192v3		prime192v3
prime239v1		prime239v1
prime239v2		prime239v2
prime239v3		prime239v3
prime256v1	P-256	secp256r1,NIST P-256,prime256v1
sect113r1		sect113r1
sect113r2		sect113r2
sect131r1		sect131r1
sect131r2		sect131r2
sect163k1		sect163k1,NIST K-163
sect163r1		sect163r1
sect163r2		sect163r2,NIST B-163
sect193r1		sect193r1
sect193r2		sect193r2
sect233k1		sect233k1,NIST K-233
sect233r1		sect233r1,NIST B-233
sect239k1		sect239k1
sect283k1		sect283k1,NIST K-283
sect283r1		sect283r1,NIST B-283

openssl	Windows CNG	Java SE 7
sect409k1		sect409k1,NIST K-409
sect409r1		sect409r1,NIST B-409
sect571k1		sect571k1,NIST K-571
sect571r1		sect571r1,NIST B-571
c2pnb163v1		
c2pnb163v2		
c2pnb163v3		
c2pnb176v1		
c2tnb191v1		c2tnb191v1
c2tnb191v2		c2tnb191v2
c2tnb191v3		c2tnb191v3
c2pnb208w1		
c2tnb239v1		c2tnb239v1
c2tnb239v2		c2tnb239v2
c2tnb239v3		c2tnb239v3
c2pnb272w1		
c2pnb304w1		
c2tnb359v1		c2tnb359v1
c2pnb368w1		
c2tnb431r1		c2tnb431r1
wap-wsg-idm-ecid-wtls1		
wap-wsg-idm-ecid-wtls3		
wap-wsg-idm-ecid-wtls4		
wap-wsg-idm-ecid-wtls5		
wap-wsg-idm-ecid-wtls6		
wap-wsg-idm-ecid-wtls7		
wap-wsg-idm-ecid-wtls8		
wap-wsg-idm-ecid-wtls9		
wap-wsg-idm-ecid-wtls10		
wap-wsg-idm-ecid-wtls11		
wap-wsg-idm-ecid-wtls12		
Oakley-EC2N-3		
Oakley-EC2N-4		

適用事例

- 電子パスポート
  - ICAOの仕様にECDSAが入っている
  - NXP社のSmartMXチップが入っている国が多い（2006年10月現在で50カ国中36カ国）
    - SmartMXチップに搭載されているFameXEコプロセッサがECC対応
- AACS（Advanced Access Content System）
  - 高画質放送コンテンツの記録と保護
  - Blu-Ray Disc、HD-DVDで採用
  - ドライブやソフトウェア認証にECDSA、また鍵交換にECDHが採用されている。
- BlackBerry
  - コンテンツ保護、エンタープライズアクティベーション、S/MIME Support PackageにECCを利用
- Infineon ORIGA SLE 95050ファミリ
  - バッテリ認証や機器認証、温度モニタのハードウェアチップ
  - ECCに対応
- ZigBee Smart Energy
  - ZigBee：無線通信の規格。ワイヤレスセンサネットワーク向け。
  - Smart Energy：家庭内などで利用し「スマートハウス」を実現するために使われるプロファイル。認証や鍵合意（Key Agreement）、署名にECCが採用されている。
- DTCP（Digital Transmission Content Protection）
  - IEEE 1394ネットワーク上でコンテンツを伝送するときの保護規格
  - 鍵交換にECDH、機器認証にECDSAを採用

OPENSSLとWINDOWSでのECC証明書の取り扱い

実装での楕円曲線暗号証明書の取り扱い

- 鍵生成から証明書発行、さらに証明書の閲覧やSSLでの利用などを調査
- 調査項目と対象プラットフォーム
  - 鍵生成と証明書発行
    - OpenSSL
    - Windows Server
  - 証明書の利用
    - ブラウザ
    - OpenSSLのs_serverコマンド

OpenSSL：鍵生成

- 利用したOpenSSLは1.0.0e
- コマンド例
  - RSA
    - openssl req　-x509 -nodes -days 365　-newkey rsa:4096 -sha256 -keyout rsaroot.key -out rsaroot.pem
  - ECC
    - openssl ecparam -out eckey.ecparam -name secp384r1
    - openssl req -new -x509 -nodes -days 3650 -newkey ec:eckey.ecparam -sha256 -keyout ecroot.key -out ecroot.pem
- 結果
  - 67種類のうち、2種類のみ失敗
    - Oakley-EC2N-3
    - Oakley-EC2N-4
- 作成した鍵
  - 全て公開しています
  - http://www.cipher.risk.tsukuba.ac.jp/~kanaoka/pkiday2011/keys/

OpenSSL：CSR作成

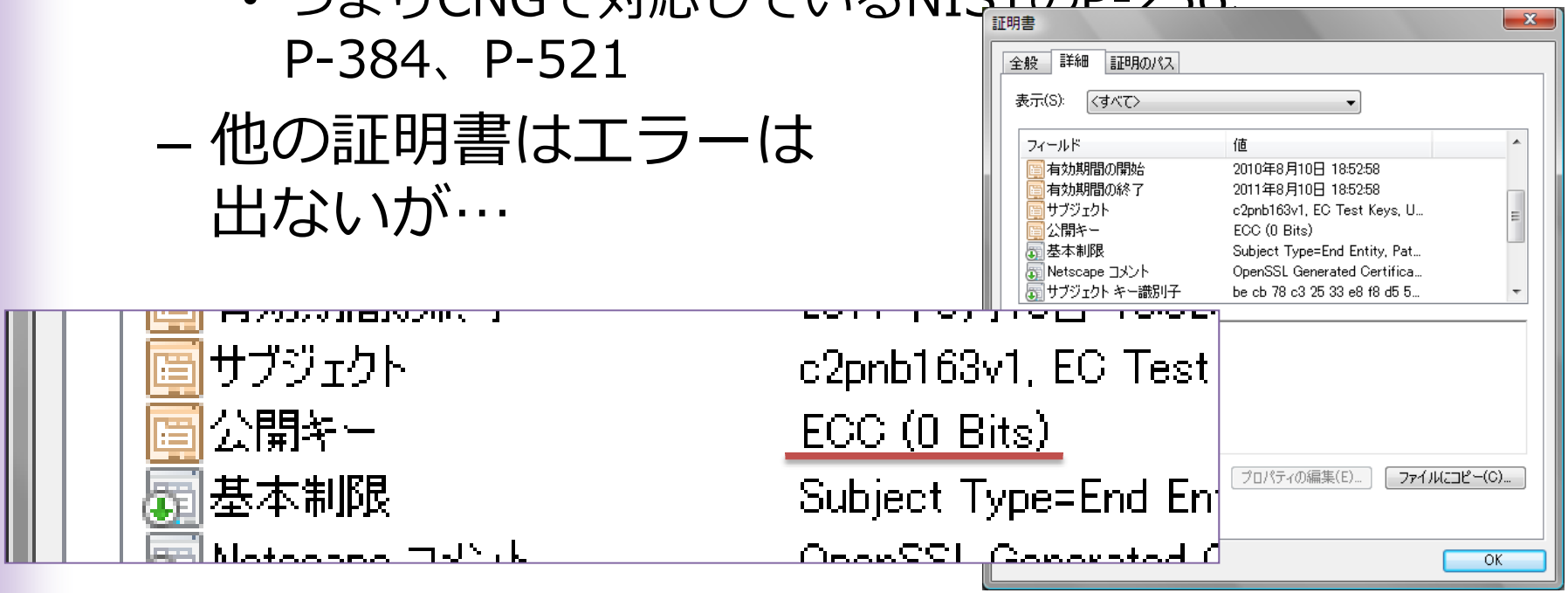
- コマンド例
  - openssl ecparam -out eckey.ecparam -name secp256k1
  - openssl req -nodes -new -newkey ec:eckey.ecparam -keyout ecMCA_byrsa.key -out ecMCA_byrsa.csr
- 結果
  - 67種類のうち、2種類のみ失敗
    - Oakley-EC2N-3
    - Oakley-EC2N-4
- 作成したCSR
  - 全て公開しています
  - http://www.cipher.risk.tsukuba.ac.jp/~kanaoka/pkiday2011/csrs/

OpenSSL：証明書発行

- OpenSSLで作成した65種類のCSRに対して実行
- ルートCAと中間CAを用意し、中間CAから証明書発行
  - ルートCA、中間CAともにprime256v1の証明書
- 結果
  - 65種類とも成功
- 作成した証明書
  - 全て公開しています
  - http://www.cipher.risk.tsukuba.ac.jp/~kanaoka/pkiday2011/certs1/

OpenSSL発行証明書をWindowsで見る

- 65種類の証明書をWindows（Vista）で閲覧
  - 3種のパラメータで問題なく見える
    - prime256v1, secp384r1, secp521r1
    - つまりCNGで対応しているNISTのP-256, P-384、P-521
  - 他の証明書はエラーは出ないが…



OpenSSL：RootCAと中間CA

- CAと中間CAの暗号アルゴリズムを変えて、パターンを複数作る
  - CA証明書：ECC or RSA
  - 中間CA証明書：ECC or RSA
  - クライアント証明書：ECC or RSA
- CA/中間CAのECC証明書はprime256v1を利用
- Windowsで証明書ファイルを見てみる
  - いずれも問題なく見える
- 作成した証明書
  - 全て公開しています
  - http://www.cipher.risk.tsukuba.ac.jp/~kanaoka/pkiday2011/certs2/
  - 命名ルール
    - *root.cer はルート証明書
    - *MCA*.cer は中間CA証明書
    - XXX_byYZ.cerがクライアント証明書
      - XXX：ECCパラメータ
      - Y： r or e、中間CAの鍵の暗号アルゴリズム
      - Z： r or e、ルートCAの鍵の暗号アルゴリズム
      - 例：ec_byre.cer → ルートCAがECC鍵、中間CAがRSA鍵、クライアントがECC鍵

Windows Serverでの鍵生成・CSR生成・証明書発行

- 鍵とCSRを作る
  - Windows Server上のツールを利用
  - 作成できず。
    - プロバイダにECCが現れない
- 自己署名証明書は作成可能
- 自己署名証明書を利用して証明書発行
  - OpenSSLで作成したCSR群（65種）
  - 限られたものだけ発行できた
    - prime256v1, secp384r1, secp521r1
    - つまりCNGで対応しているNISTのP-256、P-384、P-521

クライアント側の動作：ブラウザ （１）

- 実際に動いているWebサーバを利用
  - https://comodoecccertificationauthority-ev.comodoca.com/
  - 利用パラメータ
    - secp384r1 (P-384)
- 結果
  - IE8 （8.0.6001.19048）
    - O.K.
  - Chrome 14.0.85-186 beta-m
    - O.K.
  - Firefox 6.0.2
    - O.K.
  - Opera 11.51（Build 1087）
    - N.G. クライアントのスイートに入っていない。
  - Safari 5.1（7534.50）
    - O.K.

クライアント側：ブラウザ（２）

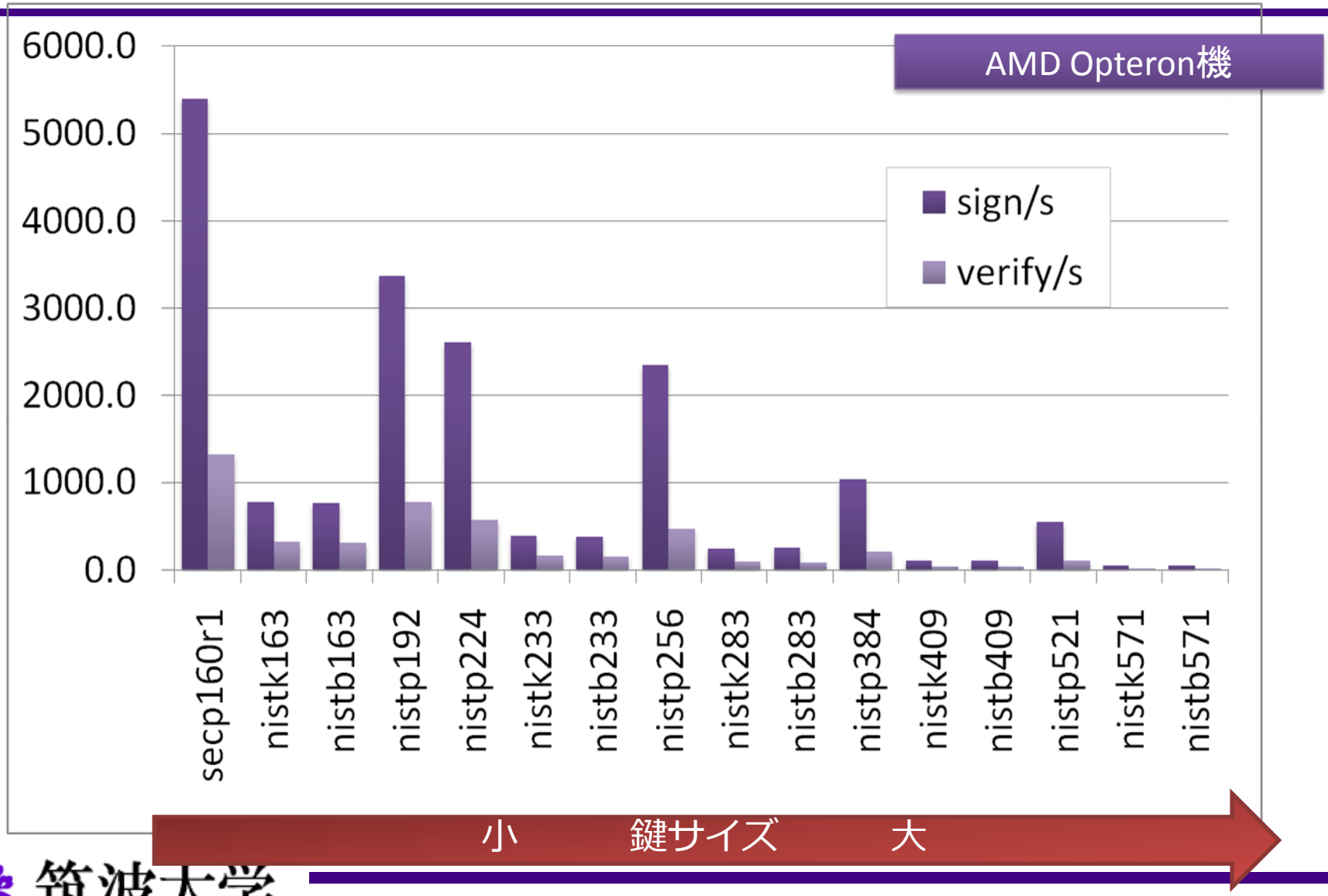
- サーバにOpenSSLのs_serverを利用
- 利用するパラメータはOpenSSLで利用可能かつ証明書発行が可能だった65種類
- 結果
  - Operaは全てのパラメータで接続できず（スイートにECC関連が入ってないため）
  - IE、Firefox、Chrome、Safariは以下の曲線を使った鍵（と証明書）で閲覧可能
    - prime256v1 （P-256）
    - secp384r1 （P-384）
    - secp521r1 （P-521）

ECCのパフォーマンス

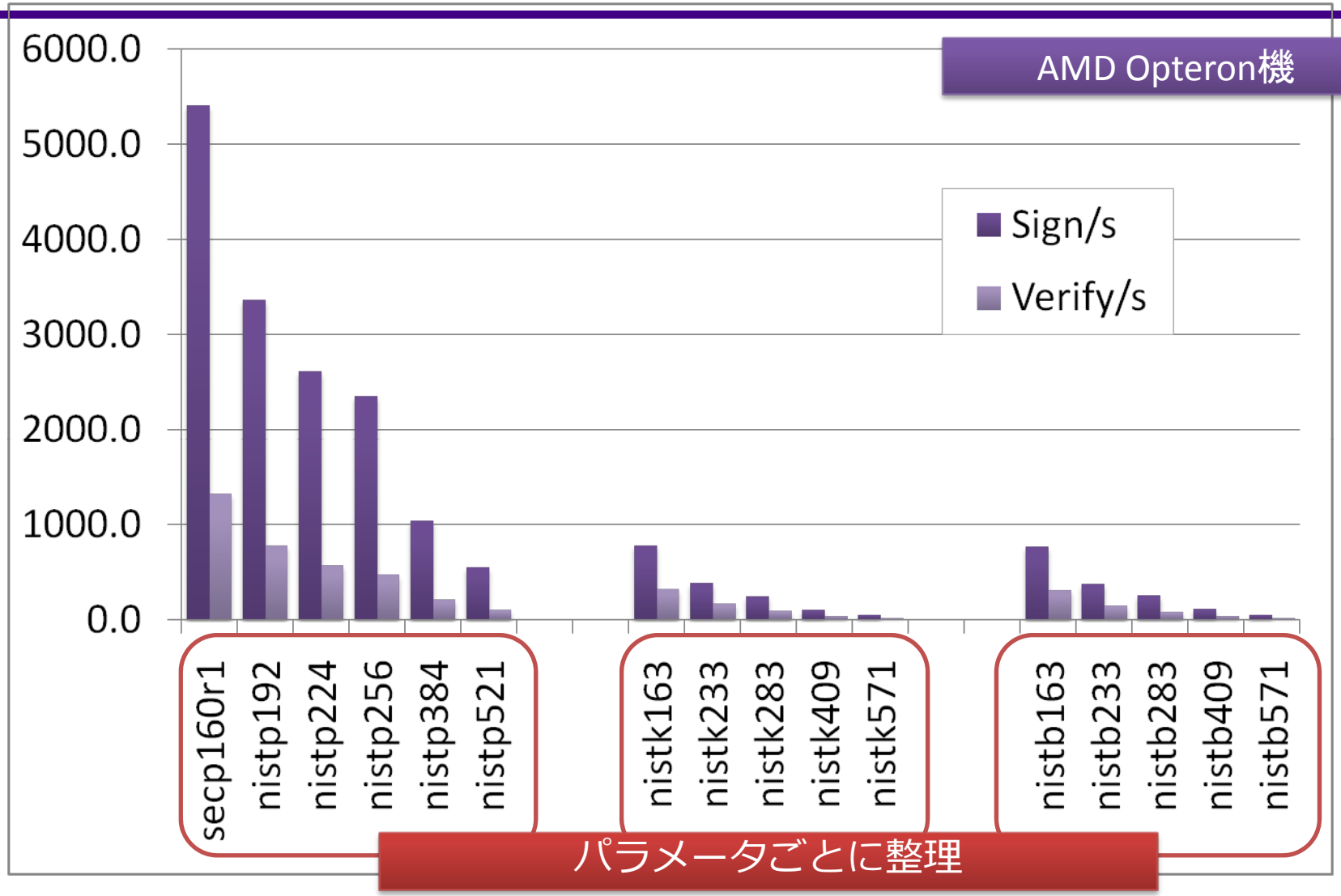
パフォーマンス測定

- ECC自体の速度とSSLでECCを利用している時の接続速度を測定
- OpenSSLのspeedコマンド（暗号自体の速度を計測）とs_time（SSLの接続時間を計測）を利用
- 測定に2台のサーバを利用
  - １）CPU:Intel Core i7 920, RAM:8GB, OS:Linux (CentOS 5.6)
  - ２）CPU:AMD Opteron 1216, RAM:2GB, OS:Linux（CentOS 5.6)
- speedコマンド利用
  - RSA, ECDH, ECDSAの速度を比較
- s_timeコマンド利用
  - 1コネクションあたりの時間を測定
  - ECC証明書を利用するスイート群、RSA証明書を用いて一部ECCを用いるスイート群、RSA証明書を利用しECCを利用しないスイート群を分けて調査

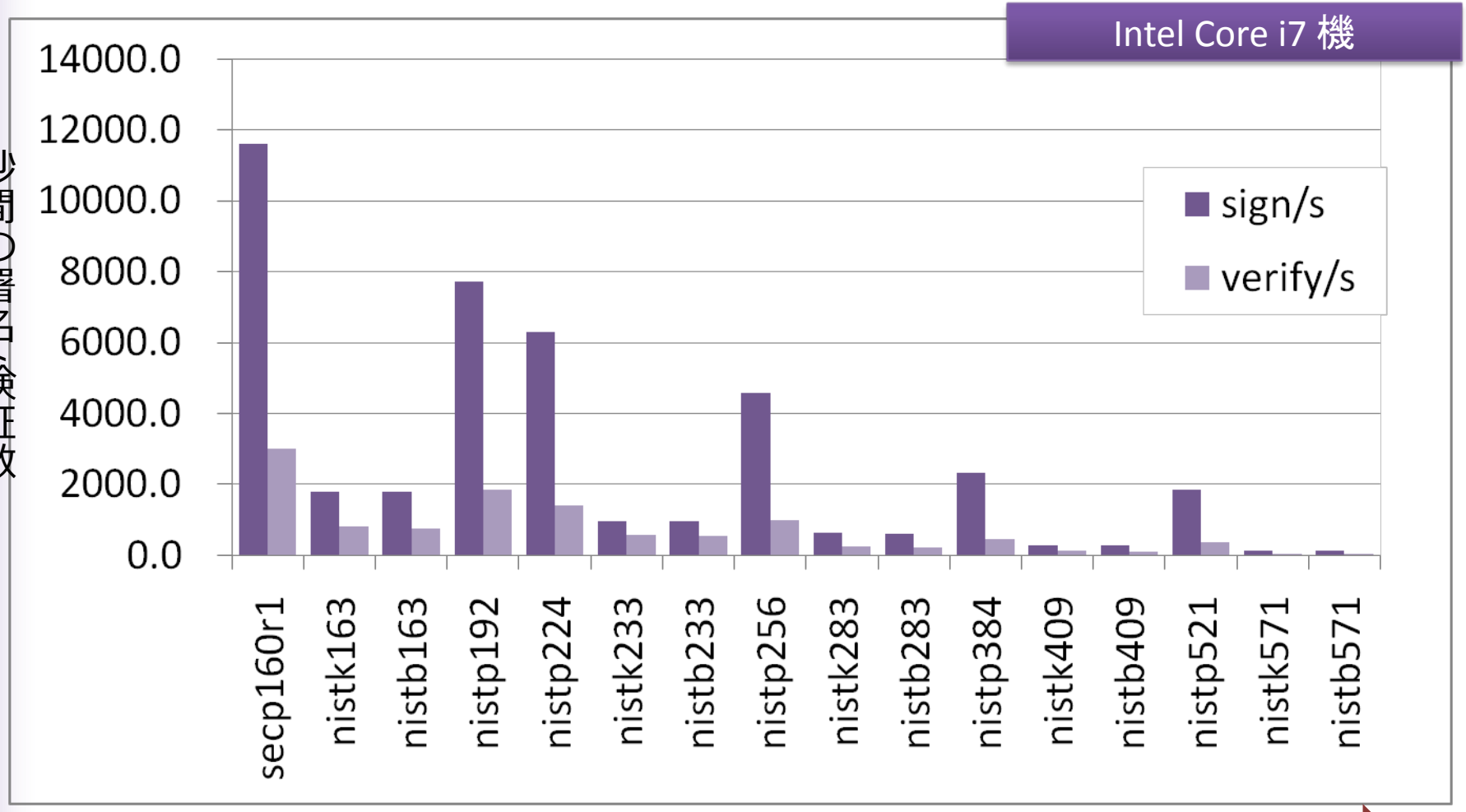
OpenSSL speedコマンドでの比較：ECDSA



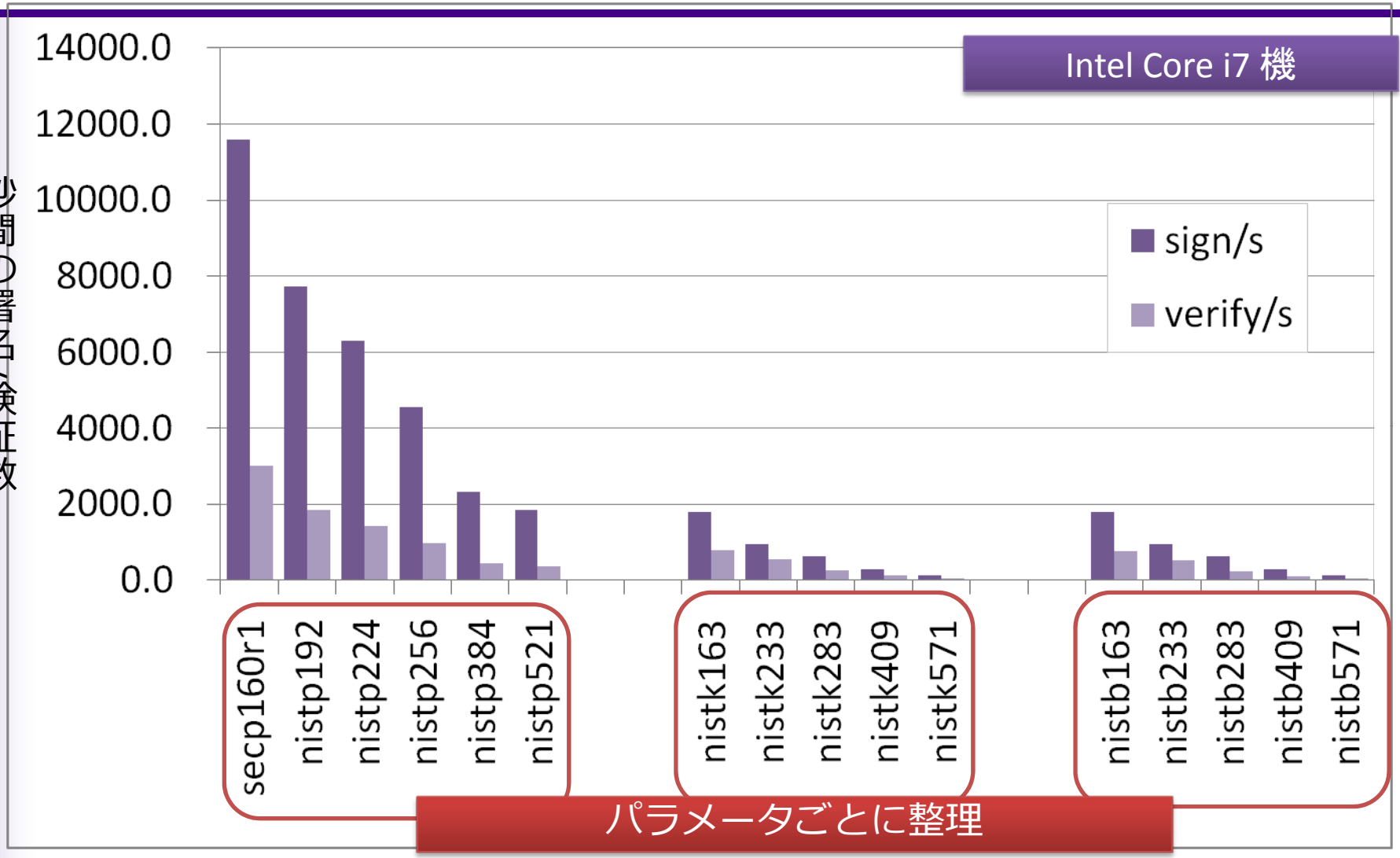
OpenSSL speedコマンドでの比較：ECDSA



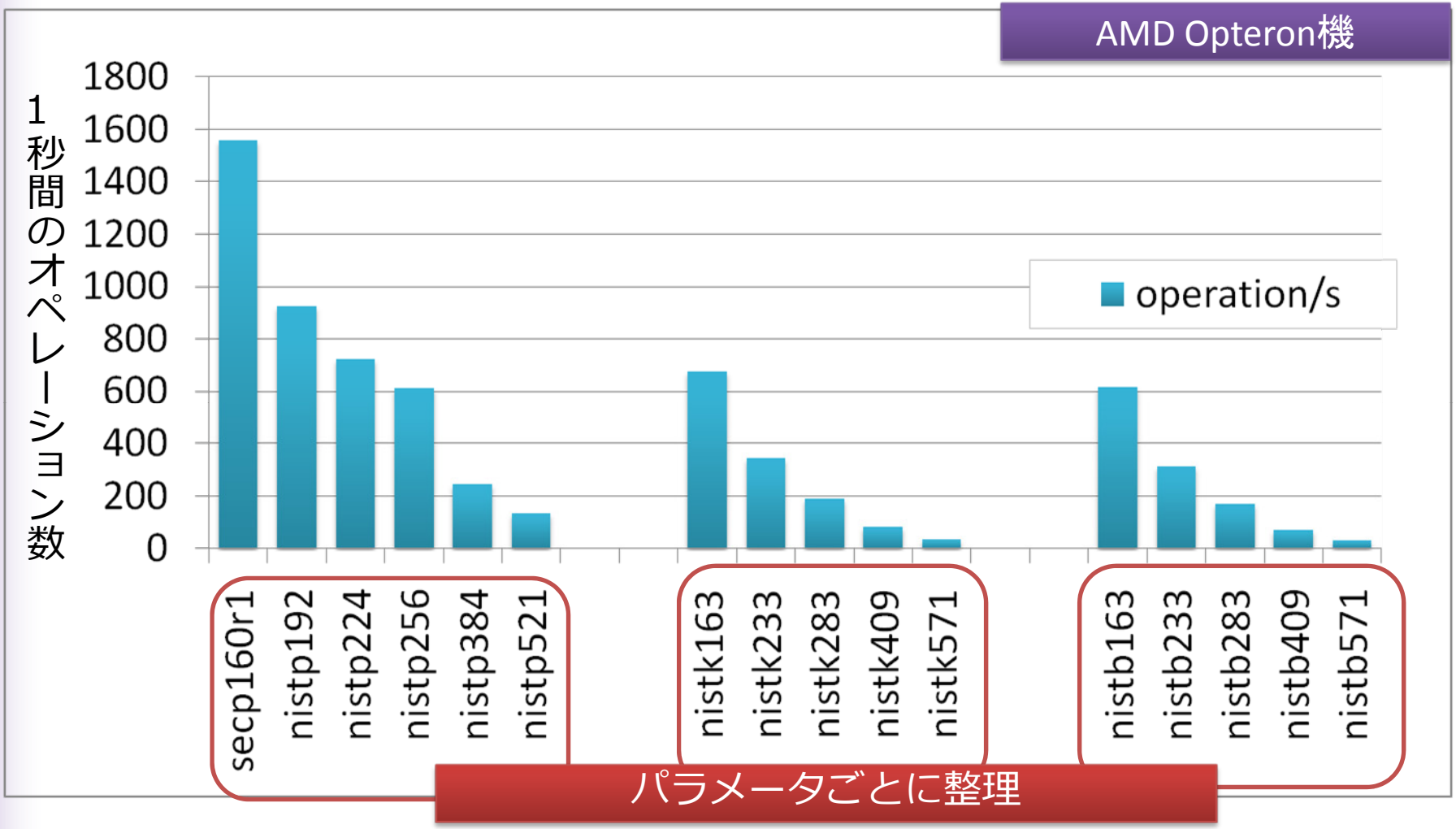
OpenSSL speedコマンドでの比較：ECDSA



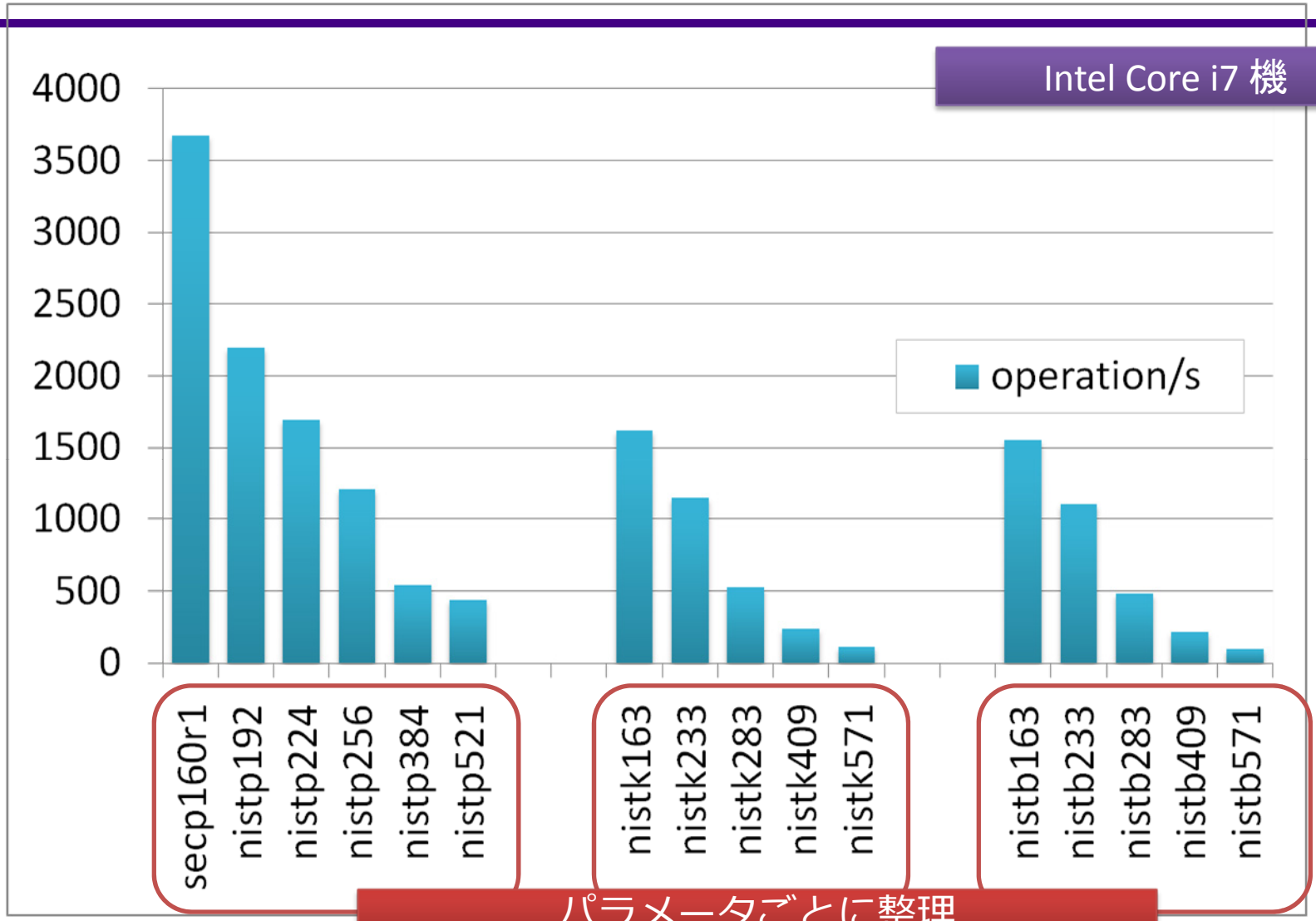
OpenSSL speedコマンドでの比較：ECDSA



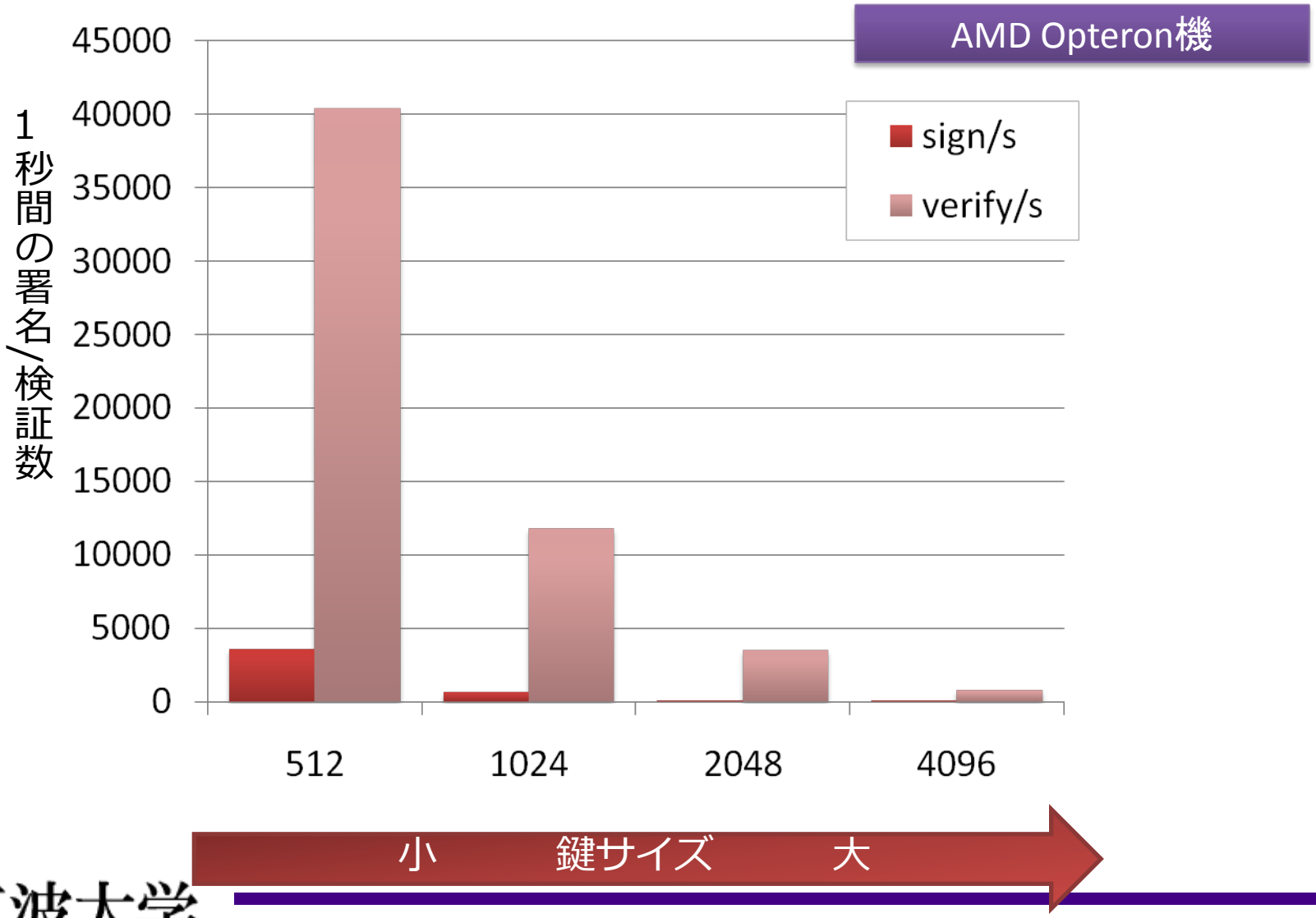
OpenSSL speedコマンドでの比較：ECDH



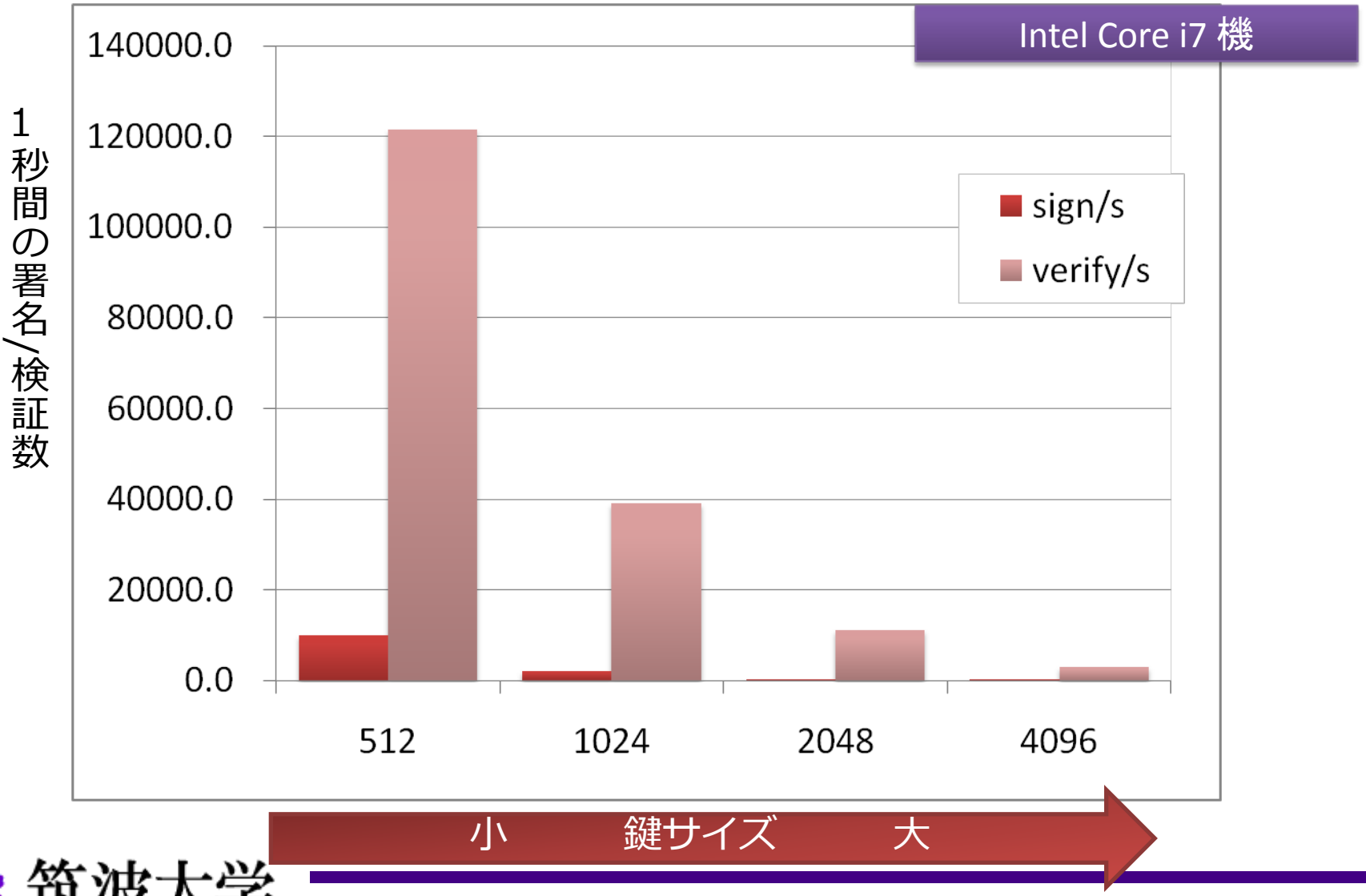
OpenSSL speedコマンドでの比較：ECDH



OpenSSL speedコマンドでの比較：RSA



OpenSSL speedコマンドでの比較：RSA

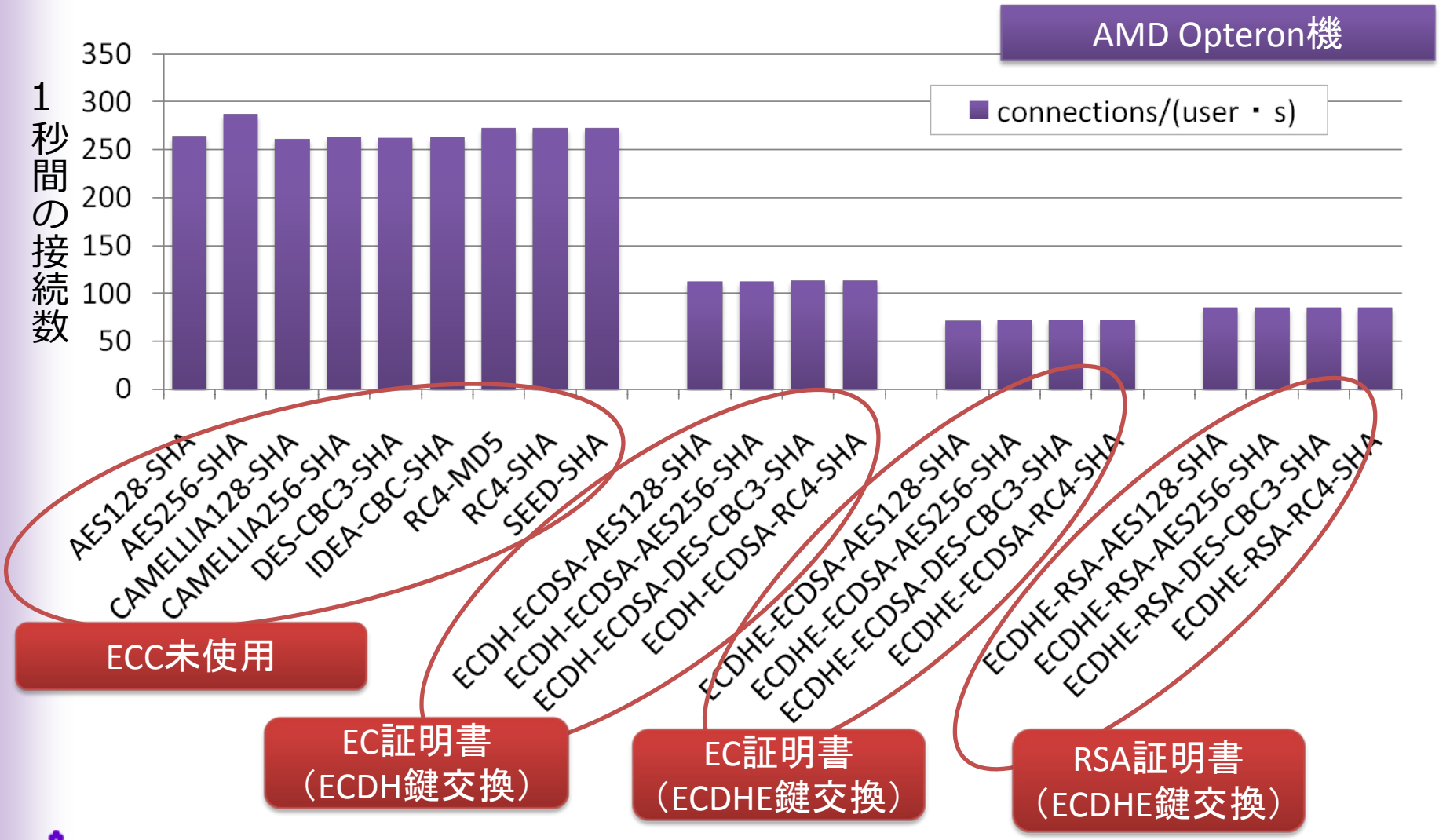


speed比較：ECDSA V.S. RSA

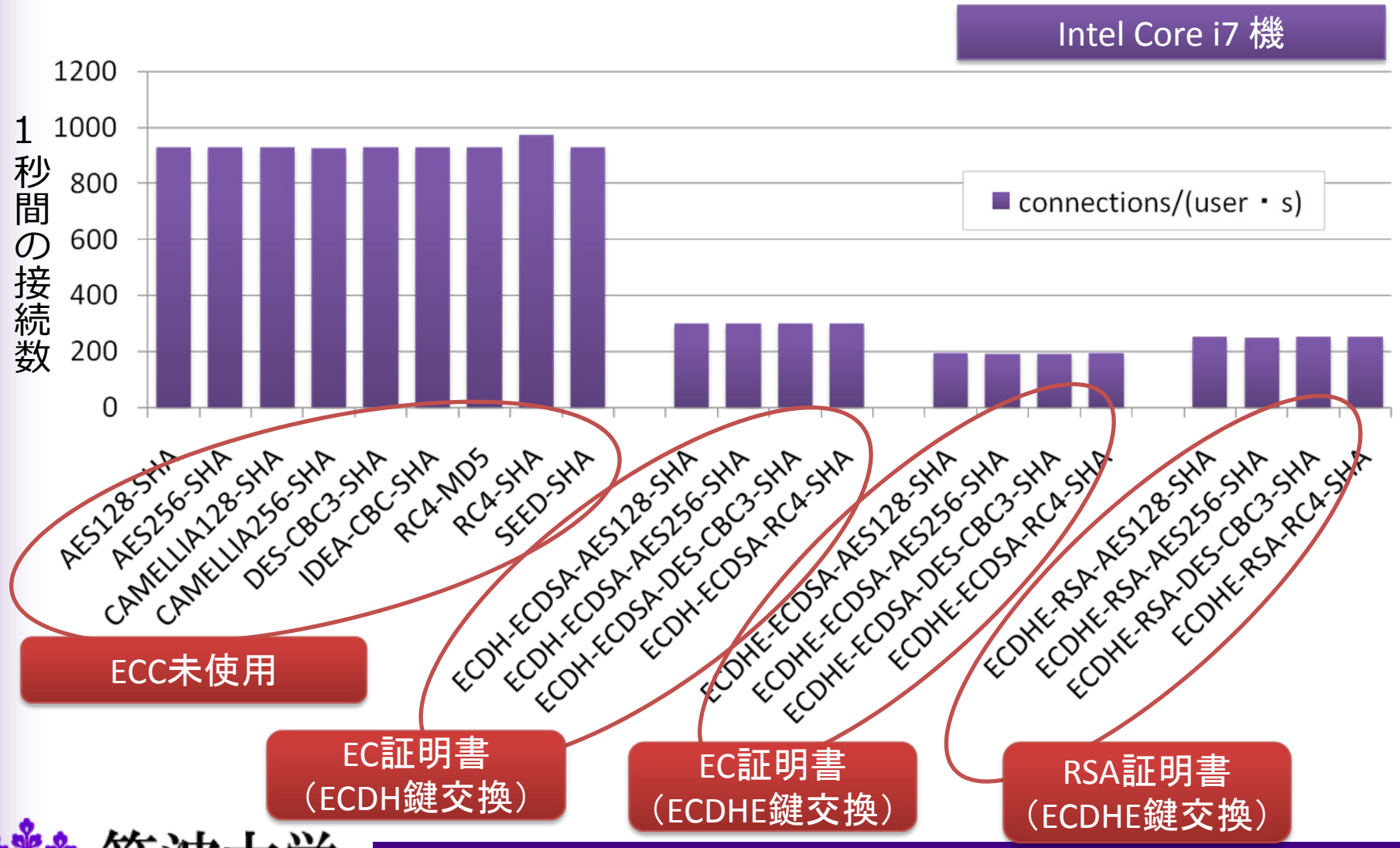
安全性 (bit)	ECDSA Param	Sign/s	Verify/s	RSA Size	Sign/s	Verify/s
				512	9926.4	121570.4
80	**secp160r1**	11592.3	3003.2	**1024**	2024.3	39128.0
112	**nistp224**	6294.4	1410.6	**2048**	323.1	11119.1
128	**nistp256**	4570.6	985.2	**3072**		
				4096	45.8	2924.2
192	**nistp384**	2327.0	455.3	**7680**		

同一の安全性の場合
1. 署名はECDSAのほうが高速
2. 検証はRSAのほうが高速

SSLコネクション速度



SSLコネクション速度



まとめ

- PKIで楕円曲線暗号を使うプラットフォームは揃ってきている
  - OpenSSL、Windows CNG、Java SE 7
- 鍵サイズ、署名長がRSAと比較して短いという利点
- 速度はRSAと比較して一長一短
  - 署名は楕円曲線暗号（ECDSA）が高速
  - 検証はRSAが高速
- パラメータにより速度が異なる
  - 素体を使うパラメータが高速
  - 演算手法に依存？

連絡先：
筑波大学 システム情報工学研究科 金岡 晃
E-mail: kanaoka@risk.tsukuba.ac.jp