☆「暗号技術　実践活用ガイド」正誤表

2020/10/28 版

略語～目次

頁	位置	誤	正	訳注
11	13	選択暗号文攻撃者		CCAは「選択暗号文攻撃」を意味することが多いですが、原著では"chosen-ciphertext attackers"（attackではない）となっているので、「選択暗号文攻撃者」と訳しています。
11	下から5	決定論的乱数生成器	決定論的ランダムビット生成器	
12	3	完全準同型暗号化	完全準同型暗号	
12	5	フォーマット保持暗号化	フォーマット保持暗号	
12	下から7	多変量二次方程式	多変数多項式	
13	11	四分円	四分ラウンド	
13	12	量子乱数発生器	量子乱数生成器	
13	19	換字-転置ネットワーク	換字-置換ネットワーク	
13	下から8	調整値付き暗号	調整可能暗号	
13	下から3	有線同等機密	無線暗号プロトコル	"Wireless Encrypted Protocol"ではなく、"Wired Equivalent Privacy"が正しい
14	16	転置	置換	
14	17	演算モード	暗号利用モード	
14	23	暗号化の安全性	暗号の安全性	
14	28	非対称暗号化	非対称暗号	
14	30-34	～暗号化	～暗号	
15	下から6	ビットの安全性を測る	ビットで安全性を測る	
16	17	換字-転置ネットワーク	換字-置換ネットワーク	
17	下から2	記憶証明への欺瞞プロトコル	記憶証明プロトコルへの攻撃	
20	3	現実世界のPRNG	RSAトラップドア付き置換	
20	13	高速累乗アルゴリズム：平方乗算	高速累乗アルゴリズム：繰り返し自乗法	
20	24	決定論的ディフィー・ヘルマン問題	決定的ディフィー・ヘルマン問題	
21	19	悪いランダムさのあるECDSA	悪いランダム性のあるECDSA	
22	24	コードを基にした暗号	符号を基にした暗号	
22	下から8	多変量暗号	多変数暗号	

第1章

頁	位置	誤	正	訳注
23	4	データを理解不能なものへと	データを理解不能なものへと	半角スペースが不要
24	1～4	この章では、もっとも単純な種類の暗号化である対称暗号に焦点を当てます。対称暗号化では、復号用の鍵と暗号用の鍵は同じです。（暗号化と復号とで異なる鍵を使う非対称暗号化や公開鍵暗号化とは異なります）。最も弱い形式の対称暗号化と、～	この章では、もっとも単純な種類の暗号である対称暗号に焦点を当てます。対称暗号では、復号用の鍵と暗号化用の鍵は同じです（暗号化と復号とで異なる鍵を使う非対称暗号や公開鍵暗号とは異なります）。最も弱い形式の対称暗号と、～	
24	下から5～6	ビットよりも文字に対して	ビットではなく文字に対して	
27	11	暗号家	暗号学者	※用語の修正は「用語の統一」欄にまとめました。
27	15	転置	置換	※用語の修正は「用語の統一」欄にまとめました。
27	15	演算モード	暗号利用モード	※用語の修正は「用語の統一」欄にまとめました。
27	下から4～5	ーー言い換えれば、	ー言い換えれば、	
29	12	暗号の演算モード	暗号利用モード	
34	1	暗号化の安全性	暗号の安全性	
35	下から8	どんなクエリを行える	どんなクエリを実行できる	
36	7	暗号文単独攻撃者	単独暗号文攻撃者	
36	14～15	暗号化クエリを行い、	暗号化クエリを実行し、	
36	21	復号クエリを行えます。	復号クエリを実行できます。	
36	27	復号クエリの実行を行えるが、	復号クエリを実行できるが、	
37	15	その整合性を変化させることはなく、	その完全性を変化させることはなく、	※用語の修正は「用語の統一」欄にまとめました。
38	17	選ばれた平文攻撃者に対する	選択平文攻撃者に対する	
38	17～18	選ばれた暗号文攻撃者に対する	選択暗号文攻撃者に対する	
38	19	強秘匿性と無作為暗号化：IND-CPA	強秘匿性とランダム化暗号：IND-CPA	
38	下から4	無作為暗号化を使うことです。	ランダム化暗号を使うことです。	
38	下から10	強秘匿性	※訳注参照	強秘匿性(semantic security)：平文を隠すだけでなく、部分的な情報も解読が困難であること。SSと略され、INDの定義とは異なりますが、安全性は等価です。
39	2～3	37ページの「安全性のゴール」で紹介した～	38ページで紹介した～	
39	5	Ci=E(K, P*i*)	C*i*=E(K, P*i*)	添え字の「i」は斜体。
39	5	iが1か2か	*i*が1か2か	「i」は斜体。
39	7	暗号化がランダム化されていないなら、	暗号化プロセスがランダム化されていないなら、	
39	8	Ciが	C*i*が	添え字の「i」は斜体。
39	11	無作為暗号化では、	ランダム化暗号では、	
39	14	強秘匿性暗号化の達成	強秘匿性暗号の達成	
39	15	決定論的乱数生成器（DRBG）	決定論的ランダムビット生成器（DRBG）	
39	22	乱数生成器を利用して	ランダムビット生成器を利用して	
39	下から4	任意のPに	任意の*P*に	「P」は斜体
41	11	非対称暗号化	非対称暗号	※用語の修正は「用語の統一」欄にまとめました。
41	12	対称暗号化	対称暗号	※用語の修正は「用語の統一」欄にまとめました。
41	25(下から3)	暗号化の2つの主なタイプで、	暗号の2つの主なタイプで、	
42	8	認証付き暗号化	認証付き暗号	※用語の修正は「用語の統一」欄にまとめました。
42	図1-4		位置をNOTEの上に移動	
43	9	フォーマット保持暗号化	フォーマット保持暗号	※用語の修正は「用語の統一」欄にまとめました。
43	10	基本的な暗号は	基本的な暗号化は	
43	20	完全準同型暗号化	完全準同型暗号	※用語の修正は「用語の統一」欄にまとめました。
43	23	に置き換えられるような暗号化です。	に置き換えられるような暗号です。	
44	5	検索可能暗号化	検索可能暗号	※用語の修正は「用語の統一」欄にまとめました。
44	14	調整可能暗号化	調整可能暗号	※用語の修正は「用語の統一」欄にまとめました。

44	15	標準的な暗号化に似ていますが、	標準的な暗号に似ていますが、	
44	16	調整と呼ばれる追加のパラメーターが	調整値と呼ばれる追加のパラメーターが	
44	17	調整は別の関係者が	調整値は別の関係者が	
44	18〜19	TEの主な利用はディスク暗号化です。	TEの主な利用はディスクの暗号化です。	
44	20	低水準な暗号化の一種です。	低水準な暗号の一種です。	
44	下から4	ディスク暗号化では、	ディスクの暗号化では、	
45	下から4	最近のモバイル暗号化の	最近のモバイル暗号の	
46	4	暗号化を使う多くの通信プロトコルは、	暗号を使う多くの通信プロトコルは、	
46	下から4	暗号家は通常	暗号学者は通常	
47	2,4,7,11,12,13	暗号化	暗号	
47	9	CHES会議の記録が	CHES会議の発表論文集が	CHES:Conference on Cryptographic Hardware and Embedded Systems（暗号化ハードウェアと組み込みシステムに関する会議）https://ches.iacr.org/

第2章

頁	位置	誤	正	訳注
54	11	量子乱数発生器（QRNG）は	量子乱数生成器（QRNG）は	
54	19	疑似乱数ビット	疑似ランダムビット	
55	最終行	エントロピープールを変更します。	エントロピープールを更新します。	
56	9	システムに侵入しエントロピープールの値を	システムに侵入してエントロピープールの値を	
56	11	PRNGは攻撃者に知られておらず推測も困難な	PRNGは攻撃者に知られておらず、推測も困難な	
63	下から7	DoS状態の生じる可能性があります。	DoS状態が生じる可能性があります。	
67	3〜4	（"Analysis of Intel's Ivy Bridge Digital Random Number Generator"『IntelのIvy Bridgeデジタル乱数発生器の解析』と題した報告書を参照）	（『Analysis of Intel's Ivy Bridge Digital Random Number Generator（IntelのIvy Bridgeデジタル乱数生成器の解析）』と題した報告書を参照）	
69	6	通常係数値が違っていればすべてのpおよびqは異なって	本来であれば相異なる法（n）の因数であるpやqはすべて異なって	

第3章

頁	位置	誤	正	訳注
75	10	Cを満たすものが	*C*を満たすものが	「C」は斜体
75	下から13	暗号方式？は(t, ε)安全であると言います。	暗号方式は(t, ε)安全であると言います。	
75	下から6	下限であると言えます。	下界であると言えます。	
76	2	効率の良い攻撃があるということは暗号に	効率の良い攻撃があるということは、暗号に	
76	16	安全となっていることがnビット鍵を	安全となっていることが、nビット鍵を	
76	下から4	この例で測っているのは暗号の評価値であり、	この例で測っているのは暗号を評価する回数であり、	
76	下から3	テクノロジーに依存してません。	テクノロジーに依存していません。	
77	7	ビットの安全性を測る	ビットで安全性を測る	
77	17	最高の安全性はnビット安全性であることを	最高の安全性はnビット安全であることを	
77	19	あくまでも安全性レベルの上限、すなわち	あくまでも安全性レベルの上界、すなわち	
77	21	小さくなる原因は以下の2つのどちらかです。	小さくなる原因は、以下の2つのどちらかです。	
79	下から4	例えば2G携帯電話暗号に対する	例えば2Gモバイル暗号に対する	
81	下から5	256 ビットを超える安全性は、金融機器を除けば実質的に不要です。	256ビットを超える安全性は、（マーケティングに使う機器としてでなければ）実質的には不要です。	
82	14	複雑性理論の領域に	計算複雑性理論の領域に	
83	20	単一転置アルゴリズムさえあれば、	一つの置換アルゴリズムがあれば、	※用語の修正は「用語の統一」欄にまとめました。
84	18	証明を無視して鍵を復元することにほとんど支障はないでしょう。	証明を無視できます。鍵を復元することにほとんど意味はありません。	
88	13	実際にこのセカンドキーはよく、	実用上はこのセカンドキーはしばしば、	
89	18	暗号学的安全性がに失敗が生じるケースは	暗号学的安全性に問題が生じるケースは	
89	下から2〜3	その後の新たな証明によって、OAEP は選択暗号文攻撃者に対し「ほぼ安全」であるにすぎないことが示されました。	※訳注参照	実際に起きたこととは少し違っています。[FOPS02]によりPOW-TDP+OAEPでIND-CCA安全になることはROMで示されており、暗号学者は既に合意済みです。また、同論文でRSAがOW→POWも示されています。この2つの合わせ技でRSA-OAEPはRSAがOWという仮定の下、ROMでIND-CCA安全です。[FOPS02] https://link.springer.com/article/10.1007/s00145-002-0204-y

第4章

頁	位置	誤	正	訳注
91	下から2	組み合わせたタイプの暗号、すなわち一連のデータ	組み合わせたタイプの暗号であり、一連のデータブロック	
92	4	攻撃技術である、	攻撃技術である、	半角スペースが不要
95	下から4	『高度なスライド攻撃』	『Advanced Slide Attacks（高度なスライド攻撃）』	
97	14	換字-転置ラウンドFを使って、	換字-置換ラウンド関数Fを使って、	※用語の修正は「用語の統一」欄にまとめました。
97	17	LとRを統合し、64ビットの	LとRをつなげて64ビットの	
97	下から3	LとRを入れ替える代わりに、ラウンドが	LとRを入れ替える代わりに、各ラウンドで	
98	6	PRFではF(X)=F(Y)となる値XとYが存在します。	※訳注参照	置換⊆関数なので、そのようなXとYが存在しない可能性があります。
98	16	高レベルの安全性（112ビット安全性）を	高レベルの安全性（112ビット安全）を	
110	9	～新しいランダムな初期値Vを選択します	～新しいランダムな初期値Vを選択します。	「。」を追加。
114	8〜9	ビットを取り込んで吐き出すだけでそれ自体はブロックの概念を妨げない、ストリーム暗号	ビット列を取り込んでビット列を吐き出し、かつブロックという概念を損なわない、ストリーム暗号	
120	10〜11	他にも14のアルゴリズムが競争していました。CAST-256、CRYPTON、DEAL、DFC、E2、FROG、HPC、LOKI97、Magenta、MARS、RC6、SAFER+、Serpent、Twofish。	他にも14のアルゴリズム（CAST-256、CRYPTON、DEAL、DFC、E2、FROG、HPC、LOKI97、Magenta、MARS、RC6、SAFER+、Serpent、Twofish）が競争していました。	

第5章

頁	位置	誤	正	訳注
121	4	チャンク	ブロック	※用語の修正は「用語の統一」欄にまとめました。
131	10	キューブ攻撃は、	**キューブ攻撃**は、	強調
136	15	For LFSR1の初期ステートの全219の値について	For LFSR1の初期ステートの全2^19の値について	
136	16	For LFSR2の初期ステートの全222の値について	For LFSR2の初期ステートの全2^22の値について	

136	17	For 最初の11クロック中のLFSR3のクロックビットの全211の値について	For 最初の11クロック中のLFSR3のクロックビットの全2^11の値について	
137	15	鍵と値の（鍵：値）組み合わせを	鍵と値の組み合わせを	

第6章

頁	位置	誤	正	訳注
158	12,13	原像困難性	原像計算困難性	※用語の修正は「用語の統一」欄にまとめました。
158	14	原像の困難性は、ランダムな	原像計算困難性とは、ランダムな	
158	17～18	ハッシュ関数は単方向関数と	ハッシュ関数は一方向関数と	
159	9	原像のコスト	原像計算のコスト	
160	14	衝突困難性	衝突計算困難性	※用語の修正は「用語の統一」欄にまとめました。
162	12	省メモリー衝突探索 :Rhoメソッド	省メモリー衝突探索 :ρ（「ロー」と読みます）法	※用語の修正は「用語の統一」欄にまとめました。
163	1	Rhoハッシュ関数の構造。	ρハッシュ関数の構造。	
182	4	記憶証明への欺瞞プロトコル	記憶証明プロトコルへの攻撃	

第7章

頁	位置	誤	正	訳注
186	下から12	整合性	完全性	
187	5～6	組み合わせをでっち上げたものは	組み合わせを不正な手段で作り出したものは	
188	下から3～4	こうした安全の概念の学問的な言い回しは、ランダム関数との識別不可能性です	こうした安全性の概念は、学問的にはランダム関数との識別不可能性と呼ばれます	
189	2～3	PRFは根本的にMACよりも強力であり、その主な理由はMACの方が安全性要件が弱いということです。	PRFは根本的にMACよりも強力であるとされます。その主な理由は、MACの方が安全性の要件が弱いということです。	
189	7	であることは、その値を推測できないことをも意味します。	であるならば、その値を推測することはできません。	
189	下から4	しかし心配は無用です。実際のアプリケーションにそのようなMAC構造が見られることはないでしょう。	しかし実際のアプリケーションにそのようなMAC構造が見られることはないので、心配は無用です。	
190	4	方法として明白に思われるものの1つは、	作る方法として単純に考えつくのは、	
190	18	Hash(K||M1||M2)を計算できません。	Hash(K||M1||M2)を計算できます。	
190	21	MACやPRFと同じくらいに低くなります。	MACやPRFと同程度まで低くなります。	
192	10	Hash((K⊕opad) Hash((K⊕ipad)M))	Hash((K⊕opad) || Hash((K⊕ipad) || M))	
192	図7-1	Compress	圧縮	
194	図7-2	衝突	圧縮	
194	図7-2	Collision	衝突	
195	22	T1とK2を導出します	K1とK2を導出します	
196	3,8	メッセージチャンク	メッセージブロック	※用語の修正は「用語の統一」欄にまとめました。
197	最終行	安全性の目標を達成するのに	安全性のゴールを達成するのに	
199	6	幸い、1メッセージ限りの安全性から複数のメッセージの安全性へと移行する方法	幸い、1メッセージ限りではなく複数のメッセージについて安全性を実現する方法	
202	3～4	加算やワードローテートと一緒にXORの束を使います。	加算やワードローテートと共にXORを使います。	
203	4	脆弱になる可能性があります。	脆弱性を抱える可能性があります。	
204	下から5	実装者は固定時間の実装を行う	実装者は定数時間の実装を行う	
204	下から5	バッファを固定時間で比較するものです。	バッファを定数時間で比較するものです。	
205	リスト7-3	2つのバッファを固定時間で比べる	2つのバッファを定数時間で比較する	
206	3	追跡論文	後継の論文	

第8章

頁	位置	誤	正	訳注
208	下から6	互いに独立しているので	互いに独立していて	
209	21	パケットが送られるごとに増分される	パケットが送られるごとにインクリメントされる	
209	下から5	暗号文を作り出します。	暗号文を生成します。	
209	最終行	MAC(生成)に従って	MAC(K2, P)に従って	
210	7～8	TLS1.3では、化は認証付き暗号に	TLS1.3では認証付き暗号に	
210	9	暗号化後MAC		
211	下から10～11	選ばれた暗号文クエリ（暗号文を作って対応する平文を問い合わせる攻撃）の実行することをを防ぐことができます。	選択暗号文クエリ（暗号文を作って対応する平文を問い合わせる攻撃）の実行を防ぐことができます。	
213	下から10	AEADの認証部分を取り除けば	安全なAEADの認証部分を取り除けば	
214	下から3	直接関係しない、	に直接関係しない	
215	下から6～7	安全なネットワークプロトコルIPSec、SSH、TLS1.2のためのインターネット技術特別調査委員会（ETF）のもあります。	インターネット技術特別調査委員会（ETF）の安全なネットワークプロトコル（IPSec、SSH、TLS1.2）の一部でもあります。	
216	6	2、3等と増分されえる	2、3等とインクリメントされる	
218	下から7	すべてがゼロの文字列がノンスとして	ゼロだけの文字列がノンスとして	
218	下から3	暗号化または復号できるという部分です。	暗号化または復号できるという点です。	
219	下から2	増えていきます	インクリメントします	

第9章

頁	位置	誤	正	訳注
231	2	すなわち、現代暗号学の土台です。	すなわち難問は、現代暗号学の土台です。	
234	下から8	複雑性の上限を示します。	複雑性の上界を示します。	
234	下から7	アルゴリズムが固定時間で実行される	アルゴリズムが定数時間で実行される	
241	8	総当たりが対象暗号の鍵を	総当たりが対称暗号の鍵を	
252	17	符号化問題は、	符号問題は、	
252	19	多変量問題は	多変数問題は	

第10章

頁	位置	誤	正	訳注
253	2	最初の公開鍵暗号化方式として	最初の公開鍵暗号方式として	
253	5～7	従来の対称鍵暗号化方式ではメッセージの暗号化と復号に同じ秘密鍵を使うのに対し、公開鍵暗号化（非対称暗号化とも呼ばれる）は2つの鍵を使います。	従来の対称鍵暗号方式ではメッセージの暗号化と復号に同じ秘密鍵を使うのに対し、公開鍵暗号（非対称暗号とも呼ばれる）では2種類の鍵を使います。	
254	7	トラップドア（落とし戸）付きと	トラップドア（落とし戸）付き置換と	

256	1	現実世界のPRNG	RSAトラップドア付き置換	
258	15～16	x^e mod n もしくはnを法とするe乗根からxを計算できる場合にも、	x^e mod nからxを計算できる場合、もしくはnを法とするe乗根を計算できる場合にも、	
259	10.4-1	通常、RSAは対象暗号方式と組み合わせて使われます。	通常、RSAは対称鍵暗号方式と組み合わせて使われます。	
262	下から4	さらに厄介なのが幻惑攻撃です。	さらに厄介なのが目隠し攻撃です。	
263	1	標的を説得して	標的を誘導して	
267	2	高速累乗アルゴリズム：平方乗算	高速累乗アルゴリズム：繰り返し自乗法	※用語の修正は「用語の統一」欄にまとめました。
268	3	平方乗算法は指数ビットを	繰り返し自乗法は指数ビットを	
268	下から5	こうした平方乗算累乗アルゴリズムは、	こうした繰り返し自乗アルゴリズムは、	
276	9	CHESワークショップの記録を	CHESワークショップの発表論文集を	

第11章

頁	位置	誤	正	訳注
278	下から6	～他も同様に評価に値します。	～他の研究者も同様に評価に値します。	
279	6	すべての算術演算は、係数pで行われます。	すべての算術演算は、pを法とします。	
279	下から9～10	しかしながら、ディフィー・ヘルマンの単純さは欺けます。1つには、これは任意の素数pや基数gだけでは動作しません。	しかしながら、ディフィー・ヘルマンの単純さはしばしば人を欺きます。一つには、任意の素数pまたは任意の基数gでうまく動くというわけではありません。	
279	下から7～8	秘密を期待しても、なんらかの値のgは共有される秘密のg^abを可能な値の小さな部分集合に制限します。	秘密に期待したとしても、gの値によっては共有された秘密g^abのとりうる値が小さな部分集合に含まれてしまいます。	
281	下から4	決定論的ディフィー・ヘルマン問題	決定的ディフィー・ヘルマン問題	※用語の修正は「用語の統一」欄にまとめました。
282	下から1～2	基本的により簡単でそれゆえにより低い安全性しか保証しなかったりします。	まったく簡単であり低い安全性しか提供しなかったりします。	
283	2	DDH	CDH	
292	7～8	長期鍵を共有された秘密の計算に統一することです。	共有された秘密を計算する際に長期鍵も組み込むことです。	
294	9	対象鍵は	対称鍵は	
295	13	OpenSSLは小群を	OpenSSLは小さな部分群を	
295	下から5	による論文『素数位数の部分群を使った離散対数方式による鍵復元攻撃』に遡ります。	による論文『A Key Recovery Attack on Discrete Logbased Schemes Using a Prime Order Subgroup（素数位数の部分群を使った離散対数方式による鍵復元攻撃）』に遡ります。	
295	下から4	群の係数として素数pを受け入れる際に、群Zp*が部分群を持たないことと、	群の法として素数pを受け入れる際に、群Zp*が小さな部分群を持たないこと（ただし位数pの部分群は存在します）と、	
296	8	（未知の鍵共有攻撃や群代理攻撃のような）	（未知の鍵共有攻撃や群表現攻撃のような）	「群表現」としましたが、数学的な意味とは異なります。
296	9	『HMQV：高性能で安全なディフィー・ヘルマンプロトコル』	『HMQV：A High-Performance Secure Diffie-Hellman Protocol（HMQV：高性能で安全なディフィー・ヘルマンプロトコル）』	
296	11	『暗黙認証付きディフィー・ヘルマンプロトコルの新しい仲間』	『A New Family of Implicitly Authenticated Diffie–Hellman Protocols（暗黙認証付きディフィー・ヘルマンプロトコルの新しい仲間）』	
296	下から4	通常、加算と置き換えられた乗算や、加算と置き換えられた指数を見つけます。その理由は、そうしたプロトコルが通常は整数群には定義されていませんが、第12章で論じるように楕円曲線には定義されているからです。	一般的に言えば、乗算は加算に、累乗は乗算に置き換えられています。その理由は、そうしたプロトコルが通常は整数群上では定義されていませんが、第2章で論じるように楕円曲線上に定義されているからです。	

第12章

頁	位置	誤	正	訳注
305	19～20	実際、楕円曲線を元にした暗号システムは、素数p（言い換えると、有限体Zpのs数）を法とする値のxとy座標で動作します。	実際、ほとんどの楕円曲線ベースの暗号方式は、素数pを法とする値（言い換えると有限体Zpの要素）をxy座標としたものを扱います。	
306	10	点に影響して、乗算の代わりに累乗を使います	点を扱い、累乗ではなく乗算を使います	
306	11	すべての楕円曲線暗号にはECDLP 問題が組み込まれていて DLP のように難しい	全ての楕円曲線暗号はECDLPに基づいています。ECDLPはDLPのように難しい	
307	10	楕円曲線は少なくとも256ビットの数に拡張します。	楕円曲線は最低でも256ビットの数の上で定義されます。	
308	13	ECDSAの署名検証	ECDSAの署名生成	
308	16～17	座標(x, y)の点kGを計算します。	点kGを計算してその座標を(x, y)とします。	
309	17～18	（電子メールアドレスのような個人的な識別子から取り出した暗号化鍵を使う暗号の）本人情報を基にした暗号のような、	IDベース暗号（電子メールアドレス等の個人的な識別子から取り出した暗号化鍵を使う暗号）のような、	
311	7	ハイブリッドな非対称鍵暗号アルゴリズムである、	ハイブリッドな非対称暗号アルゴリズムである、	
311	9	秘密から対象鍵を	秘密から対称鍵を	
311	15	ランダムな数dを選んで、基点Gが定数の場合にQ = dGを計算します。	ランダムな数dを選び、基点Gを共通パラメーターとしてQ = dGを計算します。	
311	19	Sから対象鍵Kを取り出します。	Sから対称鍵Kを取り出します。	
312	4	楕円曲線には数種類ありますが、	楕円曲線にはさまざまな種類がありますが、	
312	10～11	群の次数は小さな数の積であってはなりません、そうでないとECDLPを解くのがとても簡単になります。	群の位数は小さな数の積であってはなりません。その場合、ECDLPを解くのが非常に簡単になってしまいます。	
312	16～17	（曲線が2倍算に特定の式を必要としない場合、統一された加算法を認めていると言います）	（曲線が二倍算に特定の式を必要としない場合、その曲線は統一された加算法を許すと言います）	
312	18	曲線の作成者がaとbの原点を説明しないと、	曲線の作成者がaとbの由来を説明しないと、	
312	最終行	他10個のNIST曲線は、二項多項式で動作し、	他10種のNIST曲線は二進多項式で動作し、	
313	1	二項多項式は	二進多項式は	
313	8	係数bの素性を知っており、	係数bの由来を知っており、	
313	10	P-256の係数bは以下の	P-256の係数bは	
313	13	線の素性がなんらかの脆弱性を隠しているとは思っていません。	線の由来に脆弱性が隠されているとは考えていません。	
313	下から7	ほぼ2^255通りの256ビットの素数の剰余数で動作します。	2^255にかなり近い256ビットの素数の剰余数で動作します。	
314	9	素性不明の定数を使っています。	由来不明の定数を使っています。	
315	11	悪いランダムさのあるECDSA	悪いランダム性のあるECDSA	
315	下から2～3	ドイツのベルリンでの第27回カオス理論通信会議でfail0verflowチームに示された、2010年にゲーム機PlayStation3で攻撃が起きたように、	2010年にドイツのベルリンでの第27回CCCでfail0verflowチームが示したPlayStation3への攻撃のように、	CCC:Chaos Communication Congress https://events.ccc.de/
316	2	入力点の有効化に失敗すると、	入力点の検証に失敗すると、	

316	4〜6	これの残念な結果は、実際は異なる曲線上の点を異なる係数bと加算するかもしれないので、2点の加算時に、正しい曲線で動作していることが絶対にわからない可能性があることです。	二つの点の加算を行うときに、正しい曲線上で演算しているかどうか確信が持てないという残念な結果になります。なぜなら、実際には異なる係数bを使った別の曲線上の点を加算しているかもしれないからです。	
316	7	無効な曲線攻撃	不正曲線攻撃	
316	13	彼女は、kP = Oとなるような、比較的小さいkという低い次数の点を選びます。	彼女は位数が小さく、kP = Oとなる小さいkが存在するような点を選びます。	
316	18〜20	結果として、Pは点のより大きな群の中の小さな小群に属するように選ばれたので、結果のdBPもその小さな部分群に属していて、攻撃者がPの次数を知っていれば効率的に共有された秘密dBPを特定するのを可能にします。	結果として、点の大きな群の中の小さな部分群に属するようなPが選ばれたので、結果のdBPもその小さな部分群に属していて、攻撃者はPの位数を知っていれば共有された秘密dBPを効率的に特定することができます。	
316	21〜23	これを防ぐ1つの方法は、点PとQが正しい曲線に属することをその座標が曲線の等式を確実に満足することを確認することです。そうすることは、安全な曲線上で動作できるようにするだけであることを確かにし、この攻撃を防ぎます。	これを防ぐ方法の1つは、点の座標が曲線の等式を満足することを確かめ、点PとQが正しい曲線に属するかどうかを検証することです。そうすることで安全な曲線上での動作を保証し、このような攻撃を回避できます。	
316	24	無効な曲線攻撃のようなものは	こうした不正曲線攻撃は	
316	最終行	実践的な無効曲線攻撃	実践的な不正曲線攻撃	

第13章

頁	位置	誤	正	訳注
320	11	2013年に、エンジニアはオーバーホールされたTLSの新しい暗号の脆弱性を直すのに疲れて、TLS1.3に取り組み始めました。	2013年、TLSの新しい脆弱性を直すのに疲れたエンジニアたちは、TLSをオーバーホールしてTLS1.3に取り組み始めました。	
321	2	相互利用性があること、	相互運用性があること、	
321	9	相互利用性があることが必要でした。	相互利用性が必要でした。	
322	18	本来、TLS1.3は成熟したTLSです。	本質的には、TLS1.3は成熟したTLSであると言えます。	
322	23〜24	これは人々がよく忘れるシンプルなプロトコルであり、TLSの一部です。	これはTLSの一部であることをしばしば忘れてしまうほどにシンプルなプロトコルです。	
323	15	もしくは実体に属する	もしくは主体に属する	
324	3〜4	s: で始まる行は主題名の説明で、i: で始まる行は署名の発行者の説明です。	s: で始まる行は署名発行先の説明で、i: で始まる行は署名発行者の説明です。	
324	14〜15	〜証明書1②は証明書0に署名をした実体に属していて、証明書2③は証明書1に署名をした実体に属しています。	〜証明書1②は証明書0に署名をした主体に、証明書2③は証明書1に署名をした主体に属しています。	
324	20〜22	信頼に足る実体にのみ証明書を発行せねばならず、攻撃者が当事者になりすまして（例えば、正規のgoogle.comサーバーになりすますために）証明書を発行するのを防ぐために、	また信頼に足る主体にのみ証明書を発行するものでなくてはなりません。そして攻撃者が当事者になりすまして（例えば、正規のgoogle.comサーバーになりすますために）証明書を発行するのを防げるよう、	
325	23	base64で暗号化されたデータブロックとして提供されています。	base64で符号化されたデータのブロックとして提供されます。	
328	最終行	取り出された対象鍵を使って計算されます。	取り出された対称鍵を使って計算されます。	
331	13.3-13	これら3つの特徴、ダウングレード保護、一往復ハンドシェイク、セッション再開、を手短に説明します。	そうした機能のうち、ダウングレード保護、一往復ハンドシェイク、セッション再開の3つについて手短に説明していきます。	
331	17〜19	TLS1.3のダウングレード保護の特徴は、攻撃者がクライアントとサーバーに1.3よりも脆弱なバージョンのTLSを使わせる、ダウングレード攻撃に対する防御として設計されていることです。	TLS1.3のダウングレード保護機能は、攻撃者がクライアントやサーバーに1.3よりも脆弱なバージョンのTLSを使わせる、ダウングレード攻撃に対する防御として設計されています。	
333	図13-2(キャプション)	O-RTTデータ	0-RTTデータ	アルファベットの「オー」ではなく、数字のゼロ
333	下から14〜19	TLS1.3ハンドシェイクの間、サーバーは証明書システムを使ってクライアントを認証します。しかしながら、クライアントは認証せずに、クライアントはハンドシェイクを行った後、TLSレコードのユーザー名とパスワードで（Gmailのような）サーバーを元にしたアプリケーションで認証できます。クライアントが既にリモートサービスとセッションを確立していると、TLS接続を通してのみ送れる安全なクッキーを用いて、認証できます。	TLS 1.3のハンドシェイクプロトコルでは、サーバーは証明書システムを使ってクライアントに自身を認証させますが、クライアントは認証されません。クライアントはハンドシェイク実行後にユーザー名とパスワードをTLSのレコードプロトコルで送ることで、（Gmailのような）サーバーベースのアプリケーションに認証されます。クライアントが既にリモートサービスとセッションを確立している場合は、TLSコネクションを通じてのみ送信できるセキュアクッキーを送ることで認証されます。	
333	下から8〜最終行	ある場合においては、クライアントはサーバーがクライアントを認証するために使うものに似た証明書を元にしたメカニズムを使って、サーバーを認証することができます。クライアントはサーバーにクライアント証明書を送って、クライアントを認証する前に順番にこの証明書を検証します。しかしながら、クライアント証明書はクライアントとサーバー（つまり証明書発行者）の両方で物事を複雑にするので、クライアント証明書はほとんど使われません。クライアントは、証明書をシステムに統合し、秘密鍵を保護するために複雑な操作を実行する必要がありますが、発行者は認証されたクライアントのみが証明書を受け取ることを確認する必要があります。	場合によっては、クライアントが証明書ベースの方法（サーバーがクライアントに認証を受けるときと同様の方法）でサーバーに認証を受けることもできます。クライアントはクライアント証明書をサーバーに送り、サーバーはクライアントを認可する前にその証明書を検証します。しかしながら双方で物事が複雑になることから、クライアント証明書が使われることはめったにありません。クライアント側は証明書をシステムに統合し自分の秘密鍵を保護するために複雑な操作を行う必要があり、サーバー側も認可されたクライアントだけが証明書を受け取っていることを確認しなければならないなど、さまざまな要件があるのです。	

第14章

頁	位置	誤	正	訳注
346〜347	下から2〜最上行	これは古典的なコンピューターでO(2n)のような指数関数的な時間がかかるタスクを、量子コンピューターで多項式の複雑さで行われる場合に起き、つまりkが定数のときにO(nk)となります。	古典的コンピュータでO(2^n)のような指数関数的時間を要するタスクを、量子コンピュータでは多項式時間—すなわちある定数kについてO(n^k)—で計算できるとき、指数関数的加速が起きていると言えます。	
347	5〜8	この計算上の問題において、関数()は値mがある場合にf(x) = f(y)とy = x ⊕ mを満足する2つの値xとyがある場合を除いて、f()の出力がランダムに見えるような、nビットの文字列をnビットの文字列に変換します。	この計算問題では、関数()は通常nビット文字列をランダムに見えるnビット文字列に変換します。ただし、任意のx, y（x≠y）についてf(x) = f(y)ならばy = x ⊕ mとなるような値mが存在します。	
348	2〜3	『量子コンピューターにおける素因数分解と離散対数のための多項式時間アルゴリズム』	『Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer（量子コンピューターにおける素因数分解と離散対数のための多項式時間アルゴリズム）』	
349	下から4〜5	g^(ω + xω') mod p = 1を得て、これはx = –ω / ω'から導くと、ω + xω' mod q = 0と等価です。	g^(ω + xω') mod p = 1となり、これはω + xω' mod q = 0と等価です。このことから、x = -ω/ω' mod qを得ます。	
349	下から3	nがpビット長である場合に、	pがnビット長である場合に、	

353	2〜5	耐量子暗号の領域は、量子コンピューターで破ることのできない公開鍵アルゴリズムの設計に関するものでです。れは既製の量子コンピューターが4,096ビットのRSA係数をすぐに破れてしまう将来、RSAや楕円曲線を基にしたアルゴリズムは量子安全なものに置き換えられるということです。	耐量子暗号の領域は、量子コンピューターでも破れない公開鍵アルゴリズムの設計に関するものです。つまり量子コンピューターに対しても安全で、既製の量子コンピュータが4,096ビットのRSAを短時間で破ってしまうような未来において、RSAや楕円曲線ベースのものに取って代わっているであろうアルゴリズムです。	
353	6〜11	そのようなアルゴリズムは、素因数分解や離散対数問題の難しさを圧倒する、ショアのアルゴリズムで効率的に解けることが知られている難しい問題に依存するべきではありません。ブロック暗号やハッシュ関数のような対称なアルゴリズムは、量子コンピューターに直面して理論的安全性の半分だけを失いますが、RSAのようにひどく破られてしまうことはないでしょう。これらは、耐量子方式の基礎となるかもしれません。	そのようなアルゴリズムは、因数分解や離散対数の問題の難しさを解消するショアのアルゴリズムによって効率的に解けることが知られている難問に依存していてはなりません。ブロック暗号やハッシュ関数などの対称鍵アルゴリズムは、量子コンピューターによって理論的安全性の半分を失うことになりますが、RSAのようにひどく破られることはないでしょう。したがって、これらは耐量子方式の基礎となりえます。	
353	12	以下のセクションで、コードを基にしたもの、格子を基にしたもの、多変量なもの、〜	以下のセクションで、符号を基にしたもの、格子を基にしたもの、多変数なもの、〜	
353	16	コード	符号	※用語の修正は「用語の統一」欄にまとめました。
353	22〜23	しかし、ウェブページの平均サイズが2メガバイト前後の場合、実際に問題ではないでしょうか？	しかしウェブページの平均サイズが2メガバイト前後になっている時代に、これは問題になるでしょうか？	
354	5〜10	線形符号の場合、暗号化するワードはn ビットのベクトル v として表され、暗号化は符号ワードw = vGを計算するための、vとm × n行列Gとの乗算からなります（この例ではmはnより大きく、これは符号ワードが元のワードより長いことを意味しています）。値Gは、与えられた数tに対して、wのうちのtビットの誤りを受取側が正しいvにデコードできるようにできます。言い換えると、tは訂正される可能性のある誤りの最大数です。	線形符号の場合、符号化するワードはnビットのベクトルvとして表され、符号化はvにm × n行列Gを乗算して、符号ワードw = vGを計算することから成ります（この例ではmはnよりも大きいため、符号ワードは元のワードよりも長くなります）。値Gが与えられた数tについて構成されているとき、受信者はwにtビットの誤りがあっても正しいvを復号できます。言い換えると、tは訂正可能な誤りの最大数です。	vがn次元であれば、Gはn×m。
354	11〜16	線形符号を使ってデータを暗号化するために、McEliece暗号システムは3つの行列の秘密の組み合わせとしてGを構成して、w = vGに1ビットの固定値であるランダムな値eを加算して計算することで暗号化します。ここで、Gは公開鍵で、秘密鍵はG = ABCとなるような行列A、B、Cで構成されています。A、B、Cを知っていることで確実にメッセージをデコードしてwを読み出せます（オンラインでデコード手順の説明を見つけられます）。	線形符号を用いてデータを暗号化するにあたり、McEliece暗号方式はまず3つの行列の秘密の組み合わせから成るGを構成し、w = vGにランダムな誤りe（1ビットの数が固定されている値）を加算することで暗号化します。Gは公開鍵で、行列A、B、Cは秘密鍵で、G = ABCとなっています。A、B、Cを知っていればメッセージを復号し、wを復元できます（復元手順の説明はオンラインで見つけてください）。	
355	図14-5キャプション	s はが星形の点にもっとも近いベクトル	s が星型の点にもっとも近いベクトル	
355	8	短整数解（SIS）で知られています。	短整数解（SIS）問題として知られています。	
355	16〜20	SISとLWEはやや等価で、基本ベクトルを結合することで、格子上の最近ベクトル問題(CVP)、もしくは与えられた点に最も近い、格子内のベクトルを見つける問題の例と言い換えられます。図14-5の点ベクトルsは、基本ベクトルvとwを結合することで、星状の点に最も近いベクトルを見つける方法を示しています。	SISとLWEはある程度等価で、どちらも格子上の最近ベクトル問題（CVP）のインスタンスとして捉えられます。これは基底ベクトルを足し合わせて、与えられた点に最も近い格子上の点を探索する問題です。図14-5の点線ベクトルsは、基本ベクトルvとwを結合することで、星型の点にもっとも近いベクトルを見つける方法を示しています。	
356	1	多変量暗号	多変数暗号	※用語の修正は「用語の統一」欄にまとめました。
356	4	未知数x1、x2、x3、x4のある方程式の、以下のシステムを考えてみましょう。	未知数x1、x2、x3、x4を含む以下の連立方程式を考えてみましょう。	
356	9〜10	これらの方程式は、x4（もしくは一次元の項）のような1つの未知数か、あるいはx2x3（二次元の項）のような2つの未知数の積の項の合計で構成されています。	これらの方程式はx4のような1つの未知数（すなわち一次の項）と、x2x3のような2つの未知数の積（二次の項）の合計で構成されています。	
356	11	システムを解くには、	連立方程式を解くには、	
356	12〜13	方程式はすべて実数か整数だけ、あるいは有限個の数です。しかしながら暗号においては、方程式は通常、素数を法とする数か二値（0か1）です。	方程式の範囲にはすべての実数、整数、有限個の数のセットなどがあります。しかしながら暗号学においては、方程式の範囲は通常なんらかの素数を法とする数、あるいは二進数（0と1）となります。	
356	14〜17	ここでの問題は、方程式のランダムな二次元システムが与えられた、NP困難な解を見つけることです。したがって、多変量二次（MQ）方程式として知られるこの難しい問題は量子コンピューターはNP困難な問題を効率的に解けないので、耐量子システムの潜在的な基盤になります。	ここでの問題は、NP困難で、ランダムな多変数二次連立方程式の解を求めることです。量子コンピューターはNP困難な問題を効率的に解けないとされているため、多変数多項式（MQ）問題として知られるこの難問は、耐量子暗号の基礎となりえます。	
356〜357	下から3〜最上行	そのような暗号システムで、L1、N、L2はL1とL2が可逆でNが方程式の二次システムである場合に、線形変換となるように選ばれます（つまり、項は加算されるだけで乗算されない方程式を持ちます）。これは可逆でもある3つの二次システムの組み合わせを作りますが、その逆はL1、N、L2の逆元を知らずに特定するのは困難です。	そのような暗号方式では、L1、N、L2は以下のように選ばれます。L1とL2は可逆な線形変換です（つまり項を乗算せず、加算するだけの方程式からなります）。Nもまた可逆な二次連立方程式です。この構成により、三つを組み合わせた関数Pも可逆ですが、L1、N、L2の逆関数を知らずにPの逆関数を特定することは困難となります。	
357	2	署名の検査は、	署名の計算では、	
357	2〜3	L1、N、L2の逆元を計算します。	L1、N、L2を逆計算します。	
357	6	攻撃者がPの逆元を計算したり、PからL1、N、L2を特定しようとすると、	攻撃者がPの逆関数を計算したり、PからL1、N、L2を特定できるなら、	
357	9〜10	一つ以上の多変量方式が安全性が破られてきました。	安全であると考えられていた複数の多変数方式が破られてきました。	
359	9〜11	Ring-LWEは、NP困難なRing-LWE問題の最も難しい例を解くのと同じぐらい原理的に破られにくい暗号システムを構築するために活用できるので、暗号家にとって魅力的です。	ring-LWE問題の最も難しいものはNP困難かもしれないとされており、ring-LWEを活用すればその問題と同程度に破られにくい暗号システムを構築できるので、暗号学者にとって魅力的です。	
359	12〜14	これは基本的な格子の次元のように、大量のパラメーターがある場合にだけ真実であることを意味しています。しかしながら、実際には、とても少ないパラメーターが使われています。	例えば基になる格子問題の次元のようなパラメーターが大きい場合にのみ、真実であることを意味しています。しかしながら実際には、とても小さなパラメーターが使われることもあります。	
359	下から7	すばらしい調査を読んでください	すばらしいサーベイを読んでください	
359	下から2	結論として、耐量子暗号は耐量子署名よりも、危機的であるということです。	要点として、耐量子暗号は耐量子署名よりも危機に直面しています。	
360	下から6	実際には、耐量子方式はコードであってアルゴリズムではなくコードです。	実際には、耐量子方式はアルゴリズムではなくコードです。	
361	13〜14	『Quantum Computing Since Democritus)』（デモクリトス以来の量子計算）』	『Quantum Computing Since Democritus（デモクリトス以来の量子計算）』	

索引

誤	正
Rhoメソッド	ρ法
暗号文単独攻撃者	単独暗号文攻撃者
演算モード	暗号利用モード
落とし戸付き	トラップドア（落とし戸）付き置換
換字-転置ネットワーク	換字-置換ネットワーク
完全準同型暗号化	完全準同型暗号
決定論的ディフィー・ヘルマン問題	決定的ディフィー・ヘルマン問題
決定論的乱数生成器	決定論的ランダムビット生成器
原像困難性	原像計算困難性
検索可能暗号化	検索可能暗号
幻惑攻撃	目隠し攻撃
公開鍵暗号化	公開鍵暗号
第一原像困難性	第一原像計算困難性
対称暗号化	対称暗号
第二原像困難性	第二原像計算困難性
多変量暗号	多変数暗号
多変量二次方程式	多変数多項式問題
調整可能暗号化	調整可能暗号
転置	置換
非対称暗号化	非対称暗号
ビットの安全性	ビット安全性
平文（へいぶん）	平文（ひらぶん）
無効な曲線攻撃	不正曲線攻撃
無作為暗号化	ランダム化暗号
量子乱数発生器	量子乱数生成器

その他

位置	誤	正
裏表紙、カバー折り返し表紙側	認証された暗号	認証付き暗号