

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12

13 地上デジタルテレビジョン放送 IP 再送信  
14 Marlin IPTV-ES 運用仕様

15  
16 Document Version: 1.5  
17 Final  
18  
19 Date: 31 March, 2015

20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41

42 Copyright © 2008-2015 ALL RIGHTS RESERVED  
43 ソニー株式会社  
44 パナソニック株式会社

45  
46  
47 本仕様の内容は予告無しに変更されることがあります。  
48

## 49 Contents

50		
51	1. はじめに.....	5
52	1.1. 本書の規定範囲 .....	5
53	1.2. 引用文書 .....	6
54	1.3. 用語の定義 .....	6
55	1.4. 略語 .....	7
56	1.5. バイトオーダー .....	7
57	1.6. ビットオーダー .....	7
58	2. SAC に関する規定 .....	8
59	2.1. メッセージパラメータ .....	8
60	2.2. SAC タイムアウト .....	8
61	2.3. 1つの TCP Connection を利用可能な SAC セッション .....	8
62	2.4. Response & Commit message.....	8
63	3. Service Protocol および ECM に関する規定.....	9
64	3.1. Get Permission Protocol .....	9
65	3.1.1. メッセージパラメータの設定 .....	9
66	3.1.1.1. Get Permission Request parameters .....	9
67	3.1.1.2. Get Permission Reply parameters .....	10
68	3.1.2. メッセージパラメータの検証 .....	10
69	3.1.2.1. Get Permission Request parameters .....	10
70	3.1.2.2. Get Permission Reply parameters .....	11
71	3.1.3. UsageRuleReference の設定.....	12
72	3.1.4. 受信機における「WorkKey・SubscriptionTierBits・ExtractInfo」の 73 取得に関する処理 .....	13
74	3.1.5. DRM サーバにおける「WorkKey・SubscriptionTierBits・ExtractInfo」の 75 送信に関する処理 .....	14
76	3.2. Get Trusted Time Protocol .....	14
77	3.3. Packed Message Protocol .....	15
78	3.3.1. メッセージパラメータの設定 .....	15
79	3.3.1.1. Packed Message Request parameters .....	15
80	3.3.1.2. Packed Message Reply parameters .....	16
81	3.3.2. メッセージパラメータの検証 .....	17
82	3.3.2.1. Packed Message Request parameters .....	17
83	3.3.2.2. Packed Message Reply parameters .....	17
84	3.4. ECM に関する規定.....	17
85	3.4.1. CA_descriptor および ECM の送出.....	17
86	3.4.1.1. CA_descriptor の送出.....	17
87	3.4.1.2. ECM の送出.....	17
88	4. ネットワーク通信プロトコル (HTTP) に関する規定.....	18
89	4.1. HTTP による SAC のメッセージの伝送 .....	18
90	4.2. HTTP ヘッダ .....	18
91	4.2.1. サイズ .....	18
92	4.2.2. メソッド.....	18
93	4.2.3. リクエストヘッダ .....	18
94	4.2.4. レスポンスヘッダ .....	19
95	4.3. メッセージ処理中の受信機から HTTP のリクエストを受信した時の 96 DRM サーバの処理.....	19
97	A. Appendix (Informative) .....	20
98	A.1 SAC 処理の例 .....	20
99	A.1.1 状態遷移 .....	20
100	A.1.2 メッセージ処理 .....	22
101	(1). Challenge message 送信時の受信機処理 .....	23
102	(2). Challenge message 受信時の DRM サーバ処理.....	24

103	(3).	Response & Challenge message 送信時の DRM サーバ処理 .....	24
104	(4).	Response & Challenge message 受信時の受信機処理 .....	24
105	(5).	Response & Request message 送信時の受信機処理 .....	24
106	(6).	Response & Request message 受信時の DRM サーバ処理 .....	25
107	(7).	Reply message 送信時の DRM サーバ処理 .....	25
108	(8).	Reply message 受信時の受信機処理 .....	25
109	(9).	Request message 送信時の受信機処理 .....	25
110	(10).	Request message 受信時の DRM サーバ処理 .....	26
111	(11).	Encrypted command message 送信時の受信機処理 .....	26
112	(12).	Encrypted command message 受信時の DRM サーバ処理 .....	26
113	(13).	Command が「ACK」の Encrypted command message 送信時の	
114		DRM サーバ処理 .....	27
115	(14).	Encrypted command message 受信時の受信機処理 .....	27
116	(15).	Plain command message 受信時の受信機処理 .....	27
117	(16).	Plain command message 送信時の DRM サーバ処理 .....	27
118	(17).	Command が「ERROR」の Encrypted command message 送信時の	
119		DRM サーバ処理 .....	27
120	A.2	SAC と Service Protocol を用いたシーケンス .....	28
121	A.3	コンテンツの利用シーケンス .....	29
122	A.3.1	「WorkKey・SubscriptionTierBits・ExtractInfo」の取得処理 .....	30
123	A.3.1.1	ティアビット .....	30
124	A.3.1.2	WorkKeyManagementID .....	30
125	A.3.1.3	WorkKey (odd/even) .....	30
126	A.3.1.4	受信機における「WorkKey・SubscriptionTierBits・ExtractInfo」の管理	31
127	A.3.1.5	RenderingObligation による EXTRACT・RECORD・EXPORT .....	31
128	A.3.1.6	Packed Message Protocol による	
129		「WorkKey・SubscriptionTierBits・ExtractInfo」の取得 .....	31
130	A.3.2	コンテンツ受信時の ECM 処理 .....	32
131	A.3.3	「WorkKey・SubscriptionTierBits・ExtractInfo」の更新処理 .....	32
132	A.3.3.1	更新の有無と更新開始日時オフセット .....	32
133	A.3.3.2	「WorkKey・SubscriptionTierBits・ExtractInfo」の更新 .....	32
134	A.4	WorkKeyID および UsageRuleReference の運用例 .....	33
135	A.4.1	ティアビット・WorkKeyID・UsageRuleReference の関係 .....	33
136	A.4.1.1	WorkKeyID とティアビットとの関係 .....	33
137	A.4.1.2	UsageRuleReference と WorkKeyID との関係 .....	34
138	A.4.1.3	ティアビットと WorkKeyID・UsageRuleReference の値との関係の例..	34
139	A.4.2	WorkKey を更新する運用における WorkKeyID と UsageRuleReference との	
140		関係の例 .....	34
141	A.5	「WorkKey・SubscriptionTierBits・ExtractInfo」の更新運用の例 .....	35
142	A.6	メッセージの例 .....	46
143	A.6.1	HTTP のメッセージの例 .....	46
144	A.6.2	SAC のメッセージの例 .....	47
145	A.6.2.1	Challenge message .....	47
146	A.6.2.2	Response & Challenge message .....	47
147	A.6.2.3	Response & Request message .....	48
148	A.6.2.4	Request message .....	48
149	A.6.2.5	Reply message .....	49
150	A.6.2.6	Plain command message .....	49
151	A.6.2.7	Encrypted command message .....	50
152	A.6.3	Service Protocol のメッセージ例 .....	50
153	A.6.3.1	Get Permission Protocol .....	50
154	A.6.3.1.1.	DeviceInformation .....	50
155	A.6.3.1.2.	Get Permission Request message .....	51
156	A.6.3.1.3.	Get Permission Reply message .....	52
157	A.6.3.2	Get Trusted Time Protocol .....	52

158	A.6.3.2.1	Get TrustedTime Request.....	52
159	A.6.3.2.2	Get TrustedTime Reply.....	53
160	A.6.3.3	Packed Message Protocol .....	53
161	A.6.3.3.1	Packed Message Request message .....	53
162	A.6.3.3.2	Packed Message Reply message.....	53
163			

## 164 1. はじめに

### 165 1.1. 本書の規定範囲

166 本書では、地上デジタルテレビジョン放送 IP 再送信において、コンテンツの暗号を  
167 復号するための鍵を、以下で取得するコンテンツ（以下、本書では“コンテンツ”  
168 と記す）の利用に関し、[MIPTV]および“地上デジタルテレビジョン放送 IP 再送信  
169 Marlin IPTV-ES Specific Compliance Rules” [RTDBCR]に対する詳細規定項目と、  
170 [MIPTV]に対する追加規定項目を規定する。

- 171
- 172 ● “Marlin Trust Management Document for IPTV-ES Supplement 1: RTDB/J  
173 Support” [MTMDSUP] 2 章で規定する RTDB Client Key 及び RTDB Service Key  
174 を用いた “Marlin IPTV End-point Service Specification” [MIPTV], 4.1 項で規定  
175 される SAC において、[MIPTV], 4.2.1.2 項で規定される ActionID が  
176 「EXTRACT with Indirect Key Delivery (02h)」の Get Permission Request  
177 ● [MIPTV], 6.1.2 項で規定される ECM  
178

179 本書は、Marlin IPTV-ES Device、Marlin IPTV-ES Server（以下、本書ではそれぞれ  
180 “受信機”、“DRM サーバ”と記す）、および、ECM を送出するサービス事業者  
181 に適用する。

182 本書の規定に準拠する受信機および DRM サーバが、[MIPTV]と本書以外の任意の運  
183 用仕様と組み合わせにも準拠する場合、本書での規定と相反する規定項目がある場  
184 合には、本書の規定を優先するものとする。

185

186 以下に、本書の規定項目を示す。

- 187
- 188 ● [MIPTV]および[RTDBCR]に対する詳細規定項目
- 189 ➤ SAC に関する規定 ([MIPTV], 4.1 節 Secure Authenticated Channel (SAC)  
190 Protocol)
- 191 ☆ メッセージパラメータ
- 192 ☆ SAC タイムアウト
- 193 ☆ 1 つの TCP Connection を利用可能な SAC セッション
- 194 ☆ Response & Commit message
- 195
- 196 ➤ Service Protocol に関する規定 ([MIPTV], 4.2 節 Marlin IPTV-ES Service  
197 Protocol over SAC に関する規定)
- 198 ☆ メッセージパラメータの設定
- 199 ☆ メッセージパラメータの検証
- 200 ☆ UsageRuleReference の設定
- 201 ☆ 受信機における WorkKey ・ WorkKeyID ・ PrivateData ・  
202 SubscriptionTierBits ・ ExtractInfo の取得に関する処理
- 203 ☆ DRM サーバにおける WorkKey ・ WorkKeyID ・ PrivateData ・  
204 SubscriptionTierBits ・ ExtractInfo の送信に関する処理
- 205
- 206 ➤ ECM に関する規定 ([MIPTV], 6.1.2 項 ECM format)
- 207 ☆ CA\_descriptor および ECM の送出
- 208
- 209 ● [MIPTV]に対する追加規定項目
- 210 ➤ ネットワーク通信プロトコル (HTTP) に関する規定

- 211            ✧ HTTP による SAC のメッセージの伝送  
212            ✧ HTTP ヘッダ  
213

214            **1.2. 引用文書**

[RTDBCR]	“地上デジタルテレビジョン放送 IP 再送信 Marlin IPTV-ES Specific Compliance Rules” , Version 1.5
[MIPTV]	“Marlin IPTV End-point Service Specification” , Version 1.0.2
[MP2S]	ISO/IEC 13818-1 “Information technology – Generic coding of moving pictures and associated audio information: Systems” Second edition 2000-12-01
[MTMDSUP]	“Marlin Trust Management Document for IPTV-ES Version 2.0 Supplement 1: RTDB/J Support” , Version 1.0
[RFC2109]	HTTP State Management Mechanism
[RFC2616]	Hypertext Transfer Protocol – HTTP/1.1

215

216            **1.3. 用語の定義**

217            本書で用いる用語を以下に定義する。

218

用語	定義
CA_descriptor	[MP2S], 2.6.16 項で規定される CA_descriptor。
SAC 確立	受信機と DRM サーバとの間で相互認証とセッション鍵の共有を行うこと。
SAC 終了	SAC で用いたメッセージパラメータとセッション鍵を利用できないようにし、SAC で用いた TCP Connection を切断すること。
更新開始日時オフセット	WorkKey・WorkKeyID・PrivateData・SubscriptionTierBits・ExtractInfo を更新する運用における、更新の開始日時（更新開始日時）の NotAfter からのオフセット時間（単位は分）。更新開始日時オフセットは、PrivateData ([MIPTV], 4.2.1.5 項で規定される StatusExtension の PrivateData) で指定する。
コンテンツ	地上デジタルテレビジョン放送 IP 再送信において、暗号を復号するための鍵を[MIPTV], 4.2.1.2 項で規定される ActionID が「EXTRACT with Indirect Key Delivery (02h)」の Get Permission Request と[MIPTV], 6.1.2 項で規定される ECM で取得するコンテンツ。

219

220            本書で用いる用語と[MIPTV]の用語との対応を以下に示す。

221

本書	[MIPTV]
受信機	Marlin IPTV-ES Device
DRM サーバ	Marlin IPTV-ES Server
チャンネル	Channel
ティアビット	Tier Bits

222

223 **1.4. 略語**

224 本書で用いる略語を以下に示す。

225

略語	正式名称
LSB	Least Significant Bit
MSB	Most Significant Bit

226

227 **1.5. バイトオーダー**

228 本書で規定するプロトコルの多バイト数値のバイトオーダーは、“Big Endian”で  
229 ある。

230

231 **1.6. ビットオーダー**

232 本書で規定するプロトコルのビットオーダーは、“MSB First”である。

233

## 234 **2. SAC に関する規定**

235 本章では IPTV-ES SAC の運用を規定する。

236

### 237 **2.1. メッセージパラメータ**

238 [MIPTV], 4.1.3 項で規定されるプロトコルのメッセージパラメータの運用を以下に示  
239 す。

240

#### 241 ● SenderID

242 ☆ 受信機は、DRM サーバの SenderID としていかなる値を受信しても、  
243 「NULL value (00h)」を受信したものとして処理する。

#### 244 ● SequenceNumber

245 ☆ 受信機は Request message 送信時に SequenceNumber が  $(2^{24}-$   
246  $3)$  以上になる場合には、[MIPTV], 4.1.4.10.1 項に従い、SAC を終了す  
247 る。

#### 248 ● TransactionFlag

249 ☆ DRM サーバと受信機は、Encrypted command message の  
250 TransactionFlag を検証しない。

#### 251 ● Status

252 ☆ DRM サーバは、Plain command message の Status として「Certificate  
253 issuer mismatch (8005h)」を運用しない。

#### 254 ● SinkCertificate

255 ☆ 受信機が送信する SinkCertificate は certificate chain を含む PKIPath と  
256 する。

#### 257 ● SourceCertificate

258 ☆ DRM サーバが送信する SourceCertificate は certificate chain を含む  
259 PKIPath とする。

260

### 261 **2.2. SAC タイムアウト**

262 DRM サーバは、メッセージ送信後に 10 秒間はタイムアウトせずにメッセージ受信  
263 待ちを行う。

264 DRM サーバはタイムアウト後に[MIPTV], 4.1.4.10.2 項に従い SAC を終了する。

265

### 266 **2.3. 1 つの TCP Connection を利用可能な SAC セッション**

267 1 つの TCP Connection を利用可能な SAC セッションは 1 つとする。従って、受信  
268 機と DRM サーバは、SAC を終了した時、速やかに TCP Connection を切断する。

269

### 270 **2.4. Response & Commit message**

271 Response & Commit message は運用しない。

272



## 273 **3. Service Protocol および ECM に関する規定**

274 本章では、Marlin IPTV-ES Service Protocol および ECM の運用を規定する。  
275 なお、メッセージパラメータに設定する UsageRuleReference、メッセージ送信先  
276 の DRM サーバの URI を受信機が取得する方法については、本書では規定しない。  
277

### 278 **3.1. Get Permission Protocol**

279 [MIPTV], 4.2 項で規定される Get Permission Protocol は、WorkKey・WorkKeyID・  
280 PrivateData・SubscriptionTierBits・ExtractInfo の取得に用いる。以降、Get  
281 Permission Protocol で取得する一組の WorkKey・WorkKeyID・PrivateData・  
282 SubscriptionTierBits・ExtractInfo を示す場合、かぎっこを付与して「WorkKey・  
283 SubscriptionTierBits・ExtractInfo」などと表記する。  
284 本節では、以下の項目を規定する。

- 285
- 286 ● メッセージパラメータの設定
- 287 ● メッセージパラメータの検証
- 288 ● UsageRuleReference の設定
- 289 ● 受信機における「WorkKey・SubscriptionTierBits・ExtractInfo」の取得に関する  
290 処理
- 291 ● DRM サーバにおける「WorkKey・SubscriptionTierBits・ExtractInfo」の送信に  
292 関する処理
- 293

294 なお、DRM サーバは、同時期に一对となる 2 つの WorkKey (odd) と WorkKey  
295 (even) とを常に発行する運用をおこなうこととする。よって、受信機は、  
296 「WorkKey・SubscriptionTierBits・ExtractInfo」を取得する場合、一对となる 2 つ  
297 の WorkKey (odd) と WorkKey (even) とを取得することとする。  
298 以降、2 つの Get Permission Reply で同時期に取得する同一 ServiceProviderID・同  
299 一 WorkKeyManagementID の WorkKey (odd) と WorkKey (even) の組、または、  
300 Packed Message Reply で同時に取得する同一 ServiceProviderID・同一  
301 WorkKeyManagementID の WorkKey (odd) と WorkKey (even) の組を示す場合、  
302 “一对の” という表記を加え、“一对の WorkKey”、“一对の WorkKey (odd) と  
303 WorkKey (even) ”などと表記する。  
304

#### 305 **3.1.1. メッセージパラメータの設定**

306 受信機および DRM サーバは、以下の規定に従い、メッセージパラメータを設定す  
307 る。  
308

##### 309 **3.1.1.1. Get Permission Request parameters**

310 受信機は、[MIPTV], 4.2.1.2 項および以下の規定に従い、Get Permission Request の  
311 メッセージパラメータを設定する。

- 312
- 313 ● UsageRuleReference
- 314 ▶ 事前に取得する UsageRuleReference を設定する。UsageRuleReference の  
315 規定については、3.1.3 項を参照のこと。  
316

### 317 3.1.1.2. Get Permission Reply parameters

318 DRM サーバは、[MIPTV], 4.2.1.3 項、4.2.1.4 項および以下の規定に従い、Get  
319 Permission Reply のメッセージパラメータを設定する。

320

- 321 ● Status

- 322 ➤ 3.1.2.1 項を参照のこと。

- 323 ● WorkKeyID

- 324 ➤ 同一 ServiceProviderID かつ同一 WorkKeyManagementID の WorkKey に関  
325 し、同時期に発行する一対の WorkKey (odd) と WorkKey (even) の  
326 WorkKeyVersion には連続した値を設定する。WorkKeyVersion が 255  
327 (FFh) と 0 (00h) とである場合は、連続した値とみなす  
328 なお、WorkKeyID の運用例については、A.4 節を参照のこと。

- 329 ● PrivateData

- 330 ➤ 同一 ServiceProviderID かつ同一 WorkKeyManagementID の WorkKey に関  
331 し、同時期に発行する一対の WorkKey (odd) と WorkKey (even) の  
332 PrivateData には、送信する「WorkKey・SubscriptionTierBits・  
333 ExtractInfo」の次の更新有無および次の更新における NotAfter から更新  
334 開始日時までのオフセット時間（更新開始日時オフセット、単位は分）と  
335 して、同一の値を設定する。

- 336 なお、更新開始日時の詳細については、3.1.4 項の式(3.1)を参照のこと。

- 337 ☆ 次回の更新をおこなう場合、更新開始日時オフセットの値に 0001h (1  
338 分) ~ FFFFh (65535 分) を設定する。ただし、NotAfter の値が  
339 0000FFFFh (更新開始日時オフセットの最大値) ~ FFFFFFFEh である  
340 場合のみ設定可である。

- 341 ☆ 次回の更新をおこなわない場合、更新開始日時オフセットの値に 0000h  
342 を設定する。NotAfter の値が FFFFFFFFh である場合は、更新開始日時  
343 オフセットの値に 0000h を設定する。

- 344 ➤ 同一 ServiceProviderID かつ同一 WorkKeyManagementID の WorkKey に関  
345 し、同時期に発行する一対の WorkKey (odd) と WorkKey (even) の  
346 SubscriptionTierBits には同一の値を設定する。

- 347 ● NotBefore/NotAfter

- 348 ➤ 同一 ServiceProviderID かつ同一 WorkKeyManagementID の WorkKey に関  
349 し、同時期に発行する一対の WorkKey (odd) と WorkKey (even) の  
350 NotBefore には、同一の値を設定する。同様に、同時期に発行する一対の  
351 WorkKey (odd) と WorkKey (even) の NotAfter には、同一の値を設定す  
352 る。

353

### 354 3.1.2. メッセージパラメータの検証

355 DRM サーバおよび受信機は、以下の規定に従い、メッセージ受信時にメッセージパ  
356 ラメータを検証する。

357

#### 358 3.1.2.1. Get Permission Request parameters

359 DRM サーバは、[MIPTV], 4.2.4.1 項および以下の規定に従い、Get Permission  
360 Request のメッセージパラメータを検証する。

361

- 362 ● ActionID

- 363 ➤ ActionID が、以下に示す値の場合には検証失敗としない。

364            ✧ EXTRACT with Indirect Key Delivery (02h)  
365

### 366        3.1.2.2.        Get Permission Reply parameters

367        受信機は、[MIPTV], 4.2.4.2 項、4.2.4.4 項および以下の規定に従い、Get Permission  
368        Reply のメッセージパラメータを検証する。

369        受信機は、以下の検証の成功後に、一对の「WorkKey・SubscriptionTierBits・  
370        ExtractInfo」を使用することとし、なお、受信機は検証に失敗した場合は、一对の  
371        WorkKey として取得した 2 つの「WorkKey・SubscriptionTierBits・ExtractInfo」の  
372        両方を使用しない。

373        また、受信機は保持する一对の「WorkKey・SubscriptionTierBits・ExtractInfo」と同  
374        一 ServiceProviderID・同一 WorkKeyManagementID の一对の「WorkKey・  
375        SubscriptionTierBits・ExtractInfo」を取得した場合、以下の検証の成功後に、  
376        [MIPTV], 6.1.3 項の規定にしたがい一对の「WorkKey・SubscriptionTierBits・  
377        ExtractInfo」を更新する。

378

#### 379        ● WorkKeyID

380        ➤ 一对の WorkKey (odd)・WorkKey (even) として取得した 2 つの  
381        WorkKey が、同一 ServiceProviderID かつ同一 WorkKeyManagementID の  
382        一对の WorkKey (odd) と WorkKey (even) であることを検証する。  
383        したがって、取得した 2 つの WorkKeyID の値が以下の条件のいずれかに該  
384        当する場合、検証失敗とする。

385        ✧ ServiceProviderID・ReservedByte (WorkKeyID の上位 3 バイト目)・  
386        WorkKeyManagementID のいずれかの値が異なる場合

387        ✧ WorkKeyVersion の値が連続した値でない場合。ただし、2 つの  
388        WorkKeyVersion の値が 255 (FFh) と 0 (00h) である場合は、連続し  
389        た値とみなすこと。

390        ➤ PrivateData (更新開始日時オフセット) の値が 0001h~FFFFh (更新され  
391        る) である「WorkKey・SubscriptionTierBits・ExtractInfo」を更新する場合  
392        において、更新前後の WorkKey (odd)・WorkKey (even) が、同一  
393        ServiceProviderID かつ同一 WorkKeyManagementID の WorkKey であるこ  
394        とを検証する。

395        したがって、更新前後の WorkKey (odd) または WorkKey (even) の  
396        WorkKeyID の値が以下の条件に該当する場合、検証失敗とする。

397        ✧ ServiceProviderID・ReservedByte・WorkKeyManagementID のいずれ  
398        かの値が異なる場合

#### 399        ● PrivateData

400        ➤ 一对の WorkKey (odd)・WorkKey (even) として取得した 2 つの  
401        WorkKey の PrivateData (更新開始日時オフセット) の値が異なる場合、検  
402        証失敗とする。

#### 403        ● SubscriptionTierBits

404        ➤ 一对の WorkKey (odd)・WorkKey (even) として取得した 2 つの  
405        WorkKey の SubscriptionTierBits の値が異なる場合、検証失敗とする。

#### 406        ● NotBefore/NotAfter

407        ➤ 一对の WorkKey (odd)・WorkKey (even) として取得した 2 つの  
408        WorkKey の NotBefore の値が異なる場合、検証失敗とする。

409        同様に、2 つの WorkKey の NotAfter の値が異なる場合、検証失敗とする。

410        ➤ PrivateData (更新開始日時オフセット) の値が 0001h~FFFFh (更新され  
411        る) である WorkKey に関して、NotAfter (単位は分) の値が 0000FFFFh

412 (更新開始日時オフセットの最大値) よりも小さい場合、または、  
413 FFFFFFFFh (期限なし) である場合、検証失敗とする。  
414 更新開始日時オフセットの詳細については、3.1.4 項の式(3.1)を参照のこと。  
415

### 416 3.1.3. UsageRuleReference の設定

417 UsageRuleReference の設定は、表 3-1 に示す通りとする。  
418 UsageRuleReference は、上位 6 バイトの値のみを規定し、下位 10 バイトの値は規  
419 定しない。すなわち、UsageRuleReference の下位 10 バイトの値は、サービス事業  
420 者の運用により任意の値を設定して良い。  
421 UsageRuleReference の上位 6 バイトは、“ServiceProviderID”、  
422 “ReservedByte”、“WorkKeyManagementID”、および“odd/evenID”から構成  
423 される。UsageRuleReference の運用例については、A.4 節を参照のこと。  
424 なお、表 3-1 のバイトインデックスの値は、UsageRuleReference の最上位バイトか  
425 らの相対値である。  
426 また、以降では、「WorkKey (odd) ・ SubscriptionTierBits ・ ExtractInfo」に対応す  
427 る UsageRuleReference を“UsageRuleReference (odd)”、「WorkKey  
428 (even) ・ SubscriptionTierBits ・ ExtractInfo」に対応する UsageRuleReference を  
429 “UsageRuleReference (even)”と表記する。  
430

表 3-1 UsageRuleReference の設定

バイト インデ ックス	パラメータ	パラメータの説明	パラメータ値の規定
0-1	ServiceProviderID	・ [MIPTV], 4.2.1.5.1 項 で規定される ServiceProviderID である。	・ UsageRuleReference に対 応する WorkKey の ServiceProviderID と同一の値 を設定する。
2	ReservedByte	・ [MIPTV], 4.2.1.5.1 項 で規定される ReservedByte である。	・ UsageRuleReference に対 応する WorkKey の ReservedByte と同一の値 (00h) を設定する。
3-4	WorkKeyManagementID	・ [MIPTV], 4.2.1.5.1 項 で規定される WorkKeyManagement ID である。	・ UsageRuleReference に対 応する WorkKey の WorkKeyManagementID と同 一の値を設定する。
5	odd/evenID	・ [MIPTV], 4.2.1.5.1 項 で規定される WorkKeyVersion の LSB の値である。	・ LSB に UsageRuleReference に対 応する WorkKey の WorkKeyVersion の LSB と同 一の値を設定する。 ・ 上位 1 ビット目から上位 7 ビット目までは 0b を設定す る。

431

### 432 3.1.4. 受信機における「WorkKey・SubscriptionTierBits・ 433 ExtractInfo」の取得に関する処理

434 受信機は、「WorkKey・SubscriptionTierBits・ExtractInfo」の取得に関して、  
435 [MIPTV], 4.2.4.4 項および以下の規定に従い処理をおこなう。

436

437 ● 受信機は、「WorkKey・SubscriptionTierBits・ExtractInfo」の取得時には、事前  
438 に取得する UsageRuleReference (odd) ・ UsageRuleReference (even) を用  
439 いて、同一 ServiceProviderID かつ同一 WorkKeyManagementID の一対の  
440 「WorkKey (odd) ・ SubscriptionTierBits ・ ExtractInfo」と「WorkKey  
441 (even) ・ SubscriptionTierBits ・ ExtractInfo」とを取得する。

442 ● 受信機は、取得した一対の WorkKey (odd) ・ WorkKey (even) の更新開始日  
443 時オフセット (StatusExtension の PrivateData) の値が 0001h~FFFFh (更新  
444 される) である「WorkKey・SubscriptionTierBits・ExtractInfo」の更新の制御を  
445 おこなう場合は、以下に従う。

446 ▶ 受信機は、WorkKey (odd) または WorkKey (even) の NotAfter および更  
447 新開始日時オフセットの値を用いて、式(3.1)に従い更新開始日時 (単位は  
448 分) を算出する。更新開始日時オフセットの値は、NotAfter (単位は分) か  
449 ら更新開始日時までのオフセット時間 (単位は分) を示す。

450

$$(更新開始日時) = (NotAfter) - (更新開始日時オフセット) \cdots (3.1)$$

451

452

- 453       ▶ 受信機は、式(3.1)により算出した更新開始日時以降、DRM サーバから一対  
454       の更新された「WorkKey・SubscriptionTierBits・ExtractInfo」を取得するこ  
455       とができる。更新開始日時以降、受信機は速やかに更新することが望まし  
456       い。
- 457 ● 受信機は、取得した一対の WorkKey (odd) ・ WorkKey (even) の更新開始日  
458       時オフセットの値が 0000h (更新されない) である場合は、当該「WorkKey・  
459       SubscriptionTierBits・ExtractInfo」の更新の制御はおこなわない。  
460

### 461       **3.1.5.       DRM サーバにおける「WorkKey・** 462       **SubscriptionTierBits・ExtractInfo」の送信に関する処理**

463       DRM サーバは、「WorkKey・SubscriptionTierBits・ExtractInfo」の送信に関して、  
464       以下の規定に従い処理をおこなう。

- 465
- 466 ● DRM サーバは、同一 ServiceProviderID かつ同一 WorkKeyManagementID の  
467       WorkKey に関し、同時期に一対の WorkKey を発行する運用をおこなうこと。す  
468       なわち、DRM サーバは、同一 ServiceProviderID かつ同一  
469       WorkKeyManagementID の WorkKey に関して、受信機から「WorkKey  
470       (odd) ・ SubscriptionTierBits ・ ExtractInfo」または「WorkKey (even) ・  
471       SubscriptionTierBits ・ ExtractInfo」の要求を受信した場合、対応する  
472       「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」を送信する。
- 473       ▶ このとき DRM サーバは、WorkKey (odd) または WorkKey (even) のい  
474       ずれか一方は、WorkKey の送信時点で ECM を暗号化する WorkKey を送信  
475       する。
- 476 ● サービス事業者が、ServiceProviderID および WorkKeyManagementID で特定さ  
477       れる「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」を更新する運用をおこなう  
478       場合、DRM サーバは、当該「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」の  
479       受信機ごとの更新開始日時以降に当該受信機からの要求を受信した場合、少な  
480       くとも NotAfter の値を更新した「WorkKey ・ SubscriptionTierBits ・  
481       ExtractInfo」を送信する。
- 482       ▶ DRM サーバは、3.1.4 項の式(3.1)に従い、受信機ごとの更新開始日時を算  
483       出する。式(3.1)の NotAfter は、当該「WorkKey ・ SubscriptionTierBits ・  
484       ExtractInfo」に関して、当該受信機に対して最後に送信した WorkKey の  
485       NotAfter の値を用いる。  
486

### 487       **3.2. Get Trusted Time Protocol**

488       [MIPTV], 4.2.2 項で規定される Get Trusted Time Protocol は、Datetime の取得に用  
489       いる。

490       受信機は、[MIPTV], 4.2.2.2 項の規定に従い、Get Trusted Time Request のメッセー  
491       ジパラメータを設定する。また、受信機は、[MIPTV], 4.2.4.10 項の規定に従い、Get  
492       Trusted Time Reply のメッセージパラメータを検証する。

493       DRM サーバは、[MIPTV], 4.2.4.9 項の規定に従い、Get Trusted Time Request のメ  
494       ッセージパラメータを検証する。また、DRM サーバは、[MIPTV], 4.2.2.3 項の規定  
495       に従い、Get Trusted Time Reply のメッセージパラメータを設定する。  
496

### 497 3.3. Packed Message Protocol

498 [MIPTV], 4.2.3 項で規定される Packed Message Protocol は、以下のパラメータを同  
499 時に取得する場合に用いる。

500

- 501 ● 1 または複数の “一対の「WorkKey (odd) ・ SubscriptionTierBits ・  
502 ExtractInfo」と「WorkKey (even) ・ SubscriptionTierBits ・ ExtractInfo」 ”
- 503 ● 1 または複数の “一対の「WorkKey (odd) ・ SubscriptionTierBits ・  
504 ExtractInfo」と「WorkKey (even) ・ SubscriptionTierBits ・ ExtractInfo」 ”  
505 および Datetime

506

507 本節では、以下の項目を規定する。

508

- 509 ● メッセージパラメータの設定
- 510 ● メッセージパラメータの検証

511

#### 512 3.3.1. メッセージパラメータの設定

513 受信機およびDRMサーバは、以下の規定に従い、メッセージパラメータを設定する。

514

##### 515 3.3.1.1. Packed Message Request parameters

516 受信機は、[MIPTV], 4.2.3.2 項および以下の規定に従い、Packed Message Request  
517 のメッセージパラメータを設定する。

518

- 519 ● RequestMessageBoxList
  - 520 ➤ RequestMessageBoxList には、表 3-2 に示す順番で RequestMessage を格  
521 納する。
  - 522 ☆ RequestMessage の個数は  $(2 \times N)$  個または  $(2 \times N + 1)$  個とする。  
523 ここで、N は 16 以下の自然数である。  
524  $(2 \times M - 1)$  番目と  $(2 \times M)$  番目の RequestMessage には、同一  
525 ServiceProviderID ・ 同一 WorkKeyManagementID の「WorkKey ・  
526 SubscriptionTierBits ・ ExtractInfo」を取得するための Get Permission  
527 Request (UsageRuleReference の ServiceProviderID ・  
528 WorkKeyManagementID が同一の値) を格納する。ここで、M は N 以  
529 下の自然数である。  
530 なお、以降、N または M を用いた RequestMessage の個数および順番  
531 の表記については「×」を省略し、 $(2 \times N)$  を  $(2N)$  などと記す。
  - 532 ☆  $(2M - 1)$  番目の RequestMessage には「WorkKey (odd) ・  
533 SubscriptionTierBits ・ ExtractInfo」を、 $(2M)$  番目の  
534 RequestMessage には「WorkKey (even) ・ SubscriptionTierBits ・  
535 ExtractInfo」を取得するための Get Permission Request を格納する。
  - 536 ☆ RequestMessage の個数が  $(2N + 1)$  個の場合、 $(2N + 1)$  番目の  
537 RequestMessage には Get Trusted Time Request を格納する。

538

表 3-2 RequestMessageBoxList に格納可能な  
RequestMessage の組み合わせ

Request Message	1 番目の Request	2 番目の Request	...	$(2M - 1)$ 番目 *5 の	$(2M)$ 番目 *5 の Request	...	$(2N + 1)$ 番目 *5 の
-----------------	---------------	---------------	-----	--------------------	------------------------	-----	--------------------

の個数	Message	Message		Request Message *6	Message *6		Request Message
2N *5	Get Permission Request *1  ActionID : EXTRACT with Indirect Key Delivery (02h)  UsageRuleReference : UsageRuleReference (odd) *3	Get Permission Request *1  ActionID : EXTRACT with Indirect Key Delivery (02h)  UsageRuleReference : UsageRuleReference (even) *4	...	Get Permission Request *1  ActionID : EXTRACT with Indirect Key Delivery (02h)  UsageRuleReference : UsageRuleReference (odd) *3	Get Permission Request *1  ActionID : EXTRACT with Indirect Key Delivery (02h)  UsageRuleReference : UsageRuleReference (even) *4	...	
2N+1 *5	Get Permission Request *1  ActionID : EXTRACT with Indirect Key Delivery (02h)  UsageRuleReference : UsageRuleReference (odd) *3	Get Permission Request *1  ActionID : EXTRACT with Indirect Key Delivery (02h)  UsageRuleReference : UsageRuleReference (even) *4	...	Get Permission Request *1  ActionID : EXTRACT with Indirect Key Delivery (02h)  UsageRuleReference : UsageRuleReference (odd) *3	Get Permission Request *1  ActionID : EXTRACT with Indirect Key Delivery (02h)  UsageRuleReference : UsageRuleReference (even) *4	...	Get Trusted Time Request *2

539

540 \*1 : メッセージパラメータの設定については、3.1.1 項を参照のこと。

541 \*2 : メッセージパラメータの設定については、3.2 項を参照のこと。

542 \*3 : 「WorkKey (odd) ・ SubscriptionTierBits ・ ExtractInfo」に対応する

543 UsageRuleReference (odd/evenID の値が 01h) を示す。詳細は 3.1.3 項を参  
544 照のこと。

545 \*4 : 「WorkKey (even) ・ SubscriptionTierBits ・ ExtractInfo」に対応する

546 UsageRuleReference (odd/evenID の値が 00h) を示す。詳細は 3.1.3 項を参  
547 照のこと。

548 \*5 :  $N \leq 16$ 、 $M \leq N$  ( $N$ 、 $M$  はともに自然数) とする。

549 \*6 :  $(2M-1)$  番目と  $(2M)$  番目の RequestMessage の UsageRuleReference は、  
550 同一 ServiceProviderID ・ 同一 WorkKeyManagementID とする。  
551

### 552 3.3.1.2. Packed Message Reply parameters

553 DRM サーバは、[MIPTV], 4.2.3.3 項の規定に従い、Packed Message Reply のメッセ  
554 ージパラメータを設定する。  
555



### 556 3.3.2. メッセージパラメータの検証

557 DRM サーバおよび受信機は、以下の規定に従い、メッセージ受信時にメッセージパ  
558 ラメータを検証する。  
559

#### 560 3.3.2.1. Packed Message Request parameters

561 DRM サーバは、[MIPTV], 4.2.4.11 項および以下の規定に従い、Packed Message  
562 Request のメッセージパラメータを検証する。  
563

- 564 ● RequestMessageBoxList
  - 565 ▶ RequestMessageBoxList に ActionID が「EXTRACT with Indirect Key  
566 Delivery (02h)」の Get Permission Request の RequestMessage が 1 以上  
567 格納されている場合、かつ、RequestMessage の組み合わせが表 3-2 の組  
568 合せ以外の場合には検証失敗とし、Packed Message Reply parameter の  
569 Status を「Message format error (8009h)」とする。  
570

#### 571 3.3.2.2. Packed Message Reply parameters

572 受信機は、[MIPTV], 4.2.4.12 項および以下の規定に従い、Packed Message Reply  
573 のメッセージパラメータを検証する。  
574

- 575 ● ReplyMessageBoxList
  - 576 ▶ Status が「Success (0000h)」の場合には、3.1.2.2 項にしたがい、  
577 ReplyMessageBoxList に格納された ReplyMessage を検証する。  
578 ReplyMessageBoxList に格納されたいずれかの ReplyMessage の検証に失  
579 敗した場合には、Packed Message Reply 全体を検証失敗とする。  
580

### 581 3.4. ECM に関する規定

582 本節では、[MIPTV], 6.1.2 項で規定される ECM format に関する運用を規定する。  
583

- 584 ● CA\_descriptor および ECM の送出  
585

#### 586 3.4.1. CA\_descriptor および ECM の送出

##### 587 3.4.1.1. CA\_descriptor の送出

- 588 ● CA\_descriptor の descriptor\_tag の値は、09h とする。
- 589 ● CA\_descriptor の CA\_system\_ID の値は 0x000Dh とする。  
590

##### 591 3.4.1.2. ECM の送出

- 592 ● ECM の更新
  - 593 ▶ ECM の更新間隔は、6 秒以上とする。
- 594 ● ECM の再送
  - 595 ▶ ECM の再送間隔は、最小 100ms ・最大 1000ms とする。推奨値は 100ms  
596 とする。  
597

## 598 4. ネットワーク通信プロトコル (HTTP) に関する規定

599 本章では、SAC のメッセージの伝送に用いるネットワーク通信プロトコル  
600 (HTTP) の運用を規定する。

601 ネットワーク通信プロトコルは、[RFC2616]で規定される HTTP/1.1 および  
602 [RFC2109]で規定される Cookie に準拠し、以下に示す運用とする。Cookie は SAC  
603 のセッションを識別するために用いる。なお、Cookie と Set-Cookie における  
604 attribute のうち、設定および解釈を必須とするのは NAME のみとする。  
605

### 606 4.1. HTTP による SAC のメッセージの伝送

607 HTTP メッセージには SAC のメッセージを 1 個のみ格納して送信する。  
608

### 609 4.2. HTTP ヘッダ

610 本節では、必須となる HTTP ヘッダについて規定する。  
611 以下に示す HTTP ヘッダ以外は実装依存であり、受信しても解釈しなくてよい。  
612

#### 613 4.2.1. サイズ

614 受信機と DRM サーバは、以下に示す規定に従う。  
615 なお、本項におけるヘッダとは、HTTP ヘッダを含む HTTP メッセージにおける  
616 start-line から空行までを示す。  
617

- 618 ● ヘッダ 1 行のサイズ
  - 619 ▶ 受信機と DRM サーバは、ヘッダ 1 行 (CR+LF を含む) のサイズの上限は  
620 256byte とし、それを超える場合は、複数行に分割する。
  - 621 ▶ 受信機と DRM サーバは、1 行が 256byte を超えるヘッダを含むメッセージ  
622 は受信できなくてもよい。
- 623 ● ヘッダ全体のサイズ
  - 624 ▶ 受信機と DRM サーバは、ヘッダ全体のサイズが 4096byte を超えるメッセ  
625 ージを受信できなくてもよい。但し、プロキシにより 1Kbyte 程度のヘッダ  
626 が付加されても受信できるように、5Kbyte 程度のヘッダは受信できるよう  
627 受信機の実装は考慮されるべきである。
  - 628 ▶ DRM サーバは、送信するレスポンスヘッダ全体のサイズが 4096byte を超  
629 えないようにすべきである。但し、受信したリクエストヘッダに表 4-1 に記  
630 載されている以外のヘッダ項目が設定されている場合は、送信するレスポ  
631 ンスヘッダ全体のサイズが 4096byte を超えてもよい。  
632

#### 633 4.2.2. メソッド

634 受信機は、メソッドとして POST のみ運用する。  
635

#### 636 4.2.3. リクエストヘッダ

637 DRM サーバは、リクエストヘッダとして表 4-1 に記載されたもののみ解釈を必須と  
638 する。受信機は、表 4-1 の運用に従う。

表 4-1 解釈を必須とする HTTP リクエストヘッダ

ヘッダ		運用
Request	Cookie	受信機は、SAC を終了する場合に Cookie を削除する。 受信機は、Challenge message 以外を送信する場合は Set-Cookie で指定された Cookie をつけて、Challenge message 送信時には Cookie をつけない。
	Host	
General	Connection	Close のみ運用し、受信機のリクエストヘッダの設定処理において、HTTP のリクエストを送信する際に HTTP のレスポンス受信後に TCP connection を切断することが確定している場合には必ず用いる。
Entity	Content-Length	
	Content-Type	Application/octet-stream のみ運用する。

639

#### 640 4.2.4. レスポンスヘッダ

641 受信機は、レスポンスヘッダとして表 4-2 に記載されたもののみ解釈を必須とする。  
642 DRM サーバは表 4-2 の運用に従う。

643 なお、DRM サーバは HTTP のリクエストが正常である場合はステータスコードとして  
644 「200 OK」のみを送信する。受信機は、ステータスコードとして「200 OK」以  
645 外はエラーとする。なお、HTTP がエラーの場合でも、SAC は終了しなくてよい。

646

表 4-2 解釈を必須とする HTTP レスポンスヘッダ

ヘッダ		運用
Response	Set-Cookie	Set-Cookie の値は SAC のセッションを識別可能な情報を格納する。 Cookie の最大数は 1 個とする。 Set-Cookie は複数行に分割しない。
General	Cache-Control	No-cache のみ運用する。
	Connection	Close のみ運用し、TCP connection 切断時には必ず用いる。
Entity	Content-Length	
	Content-Type	Application/octet-stream のみ運用する。

647

#### 648 4.3. メッセージ処理中の受信機から HTTP のリクエストを受信し 649 た時の DRM サーバの処理

650 受信したメッセージ処理中に、メッセージを送信した受信機から HTTP のリクエ  
651 ストを受信した場合、DRM サーバは [MIPTV], 4.1 節に規定する SAC の処理は行わず  
652 に、ステータスコードが「200 OK」以外の HTTP レスポンスを送信する。

653

## 654 **A. Appendix (Informative)**

### 655 **A.1 SAC 処理の例**

#### 656 **A.1.1 状態遷移**

657 SAC 処理に関わる受信機の状態遷移を表 A-1 に、DRM サーバの状態遷移を表 A-2  
658 に示す。

659 SAC を行う 1 対の受信機と DRM サーバは、状態遷移表に従って状態を遷移する。  
660 本書の規定外となる状態とイベントの組み合わせは、表中 “-” で示す。なお、本  
661 書に規定されていないが、表 A-1 に受信機の SAC タイムアウトを記述した。

662  
663  
664  
665  
666  
667  
668  
669  
670  
671  
672  
673  
674  
675  
676  
677  
678  
679  
680  
681  
682  
683  
684  
685  
686  
687  
688  
689  
690  
691  
692  
693  
694  
695  
696  
697  
698  
699  
700  
701  
702



		受信機の状態					
		①SAC 開始前	②Challenge message 送信後のメッセージ受信待ち	③Response & Request message 送信後のメッセージ受信待ち		④Request message 送信後のメッセージ受信待ち	⑤Encrypted command message 送信後のメッセージ受信待ち
			送信する Request がある	送信する Request がない	送信する Request がある	送信する Request がない	
場合を除く)							
SAC タイムアウト	—	①に移る	①に移る	①に移る	①に移る	①に移る	①に移る

703

表 A-2 DRM サーバの状態遷移表

		DRM サーバの状態		
		⑦SAC 開始前のメッセージ受信待ち	⑧Response & Challenge message 送信後のメッセージ受信待ち	⑨Reply message 送信後のメッセージ受信待ち
メッセージ受信イベント	Challenge message 受信	A.1.2 項(2)の処理を実行する ● メッセージ検証に成功した場合は⑧に移る ● メッセージ検証に失敗した場合は⑨に移る	新たな SAC として 0 項(2)の処理を実行する。Response & Challenge message を送信した SAC は終了する。 ● メッセージ検証に成功した場合は⑧に移る ● メッセージ検証に失敗した場合は⑨に移る	新たな SAC として 0 項(2)の処理を実行する。Reply message を送信した SAC は終了する。 ● メッセージ検証に成功した場合は⑧に移る ● メッセージ検証に失敗した場合は⑨に移る
	Response & Request message 受信	⑧に移る	A.1.2 項(6)の処理を実行する ● メッセージ検証に成功した場合は⑨に移る ● メッセージ検証に失敗した場合は⑦に移る	⑨に移る
	Request message 受信	⑨に移る	⑨に移る	A.1.2 項(10)の処理を実行する ● メッセージ検証に成功した場合は⑨に移る ● メッセージ検証に失敗した場合は⑦に移る
	Encrypted command message 受信	⑨に移る	⑨に移る	A.1.2 項(12)の処理を実行して、⑦に移る
	[MIPTV], 4.1.4.1 項の Message header 検証に失敗 (Payload Type が上記のメッセージの場合を除く)	⑨に移る	⑨に移る	⑨に移る
	SAC タイムアウト	—	⑨に移る	⑨に移る

704

705 A.1.2 メッセージ処理

706 本項では、メッセージ送受信時に行うメッセージ処理を示す。

707 以降で説明するメッセージ処理の基本シーケンスについて、図 A-1 と図 A-2 に示す。  
708 図中の()つきの番号は、各メッセージの送受信処理の種類を示す。以下、各処理につ  
709 いて説明する。

710  
711  
712  
713  
714  
715  
716  
717  
718  
719  
720  
721  
722  
723  
724  
725  
726  
727

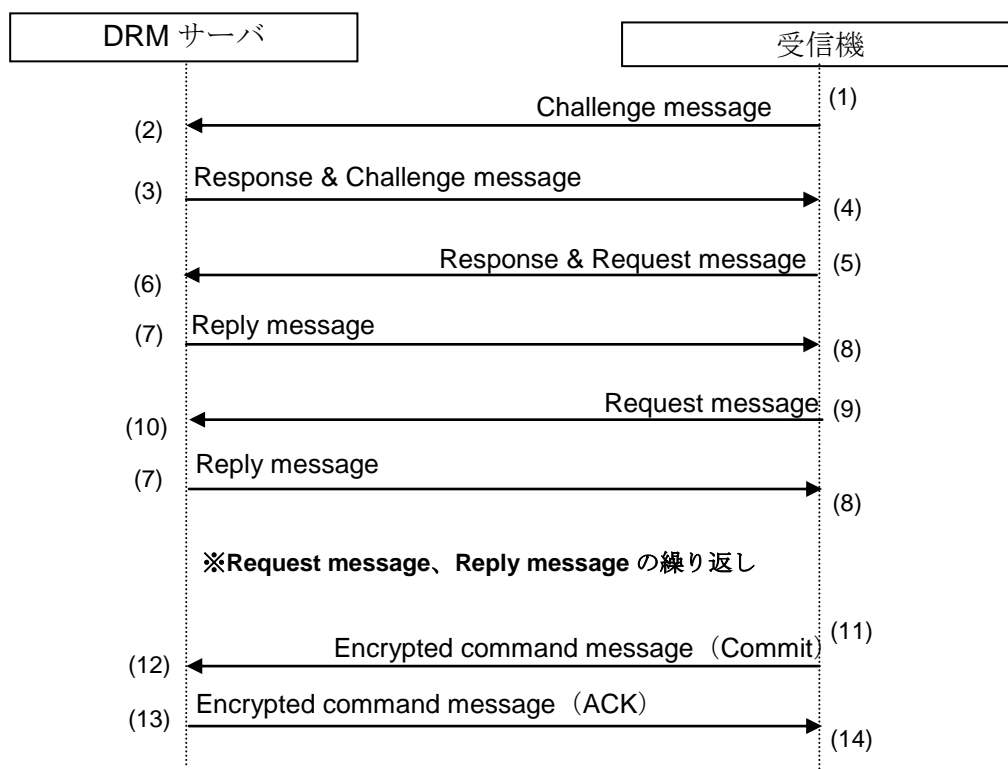


図 A-1 基本シーケンス (複数の Request を連続送信する場合)

728  
729  
730  
731  
732  
733  
734  
735  
736  
737  
738  
739  
740

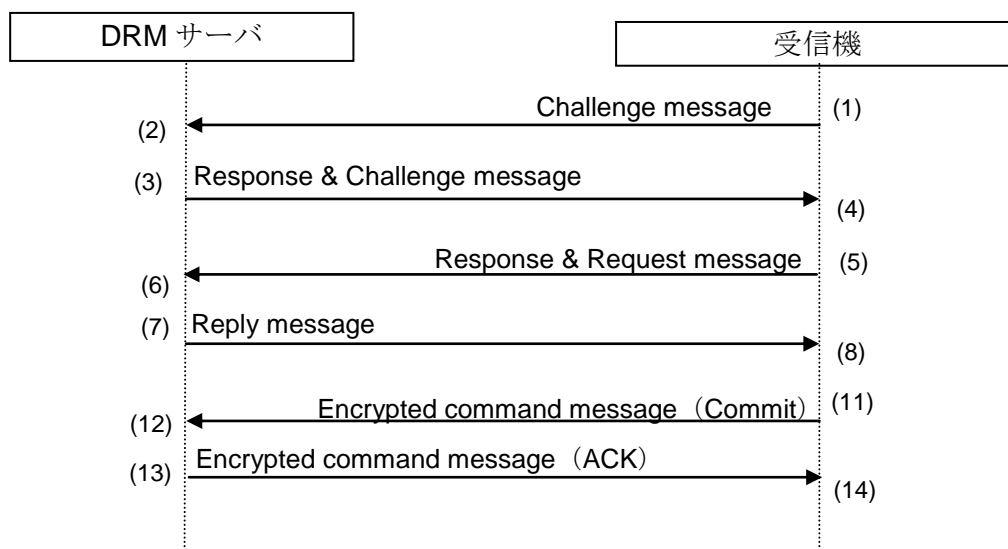


図 A-2 基本シーケンス (一つの Request のみ送信する場合)

741

742 (1). Challenge message 送信時の受信機処理

- 743 ➤ Challenge message を作成する。
- 744 ➤ DRM サーバに Challenge message を送信する。

745       ➤ Challenge message 送信後のメッセージ受信待ちの状態に移る。  
746

747 **(2). Challenge message 受信時の DRM サーバ処理**

748       ➤ [MIPTV], 4.1.4.2 項の規定に従い、Challenge message の検証を行う。  
749       ✧ 検証が成功した場合、(3)Response & Challenge message 送信時の  
750       DRM サーバ処理を実行する。  
751       ✧ 検証が失敗した場合、(16)Plain command message 送信時の DRM サー  
752       バ処理を実行する。  
753

754 **(3). Response & Challenge message 送信時の DRM サーバ処理**

755       ➤ Response & Challenge message を作成する。Response & Challenge  
756       message のメッセージパラメータに関する処理を以下に示す。  
757       ✧ Signature : SinkRandomNumber と SourceEC-DHPhase1Value に対し  
758       て生成する。  
759       ➤ 受信機にResponse & Challenge messageを送信する。  
760       ➤ Response & Challenge message送信後のメッセージ受信待ちの状態に移る。  
761

762 **(4). Response & Challenge message 受信時の受信機処理**

763       ➤ [MIPTV], 4.1.4.3項の規定に従い、Response & Challenge messageの検証を  
764       行う。  
765       ✧ 検証が成功した場合、(5)Response & Request message 送信時の受信  
766       機処理を実行する。  
767       ✧ 検証が失敗した場合、SAC を終了して、SAC 開始前の状態に移る。  
768

769 **(5). Response & Request message 送信時の受信機処理**

770       ➤ セッション鍵を生成する。  
771       ➤ Response & Request message を作成する。SequenceNumber、  
772       TransactionFlag、Request、MessageDigest は生成したセッション鍵で暗  
773       号化する。Response & Request message のメッセージパラメータに関す  
774       る処理を以下に示す。  
775       ✧ Signature : SourceRandomNumber と SinkEC-DHPhase1Value に対し  
776       て生成する。  
777       ✧ SequenceNumber : 1 を Response & Request message に設定し、保  
778       持する。  
779       ✧ TransactionFlag : 「even (00h) 」を Response & Request message  
780       に設定し、保持する。  
781       ✧ Request : Service Protocol のメッセージを設定する。  
782       ✧ MessageDigest : 暗号化前の MessageDigest を除く Response &  
783       Request message のパラメータから生成する。  
784       ➤ Response & Request message 作成後に保持している SequenceNumber を  
785       1 増加する。  
786       ➤ Response & Request messageをDRMサーバに送信する。  
787       ➤ Response & Request message送信後のメッセージ受信待ちの状態に移る。  
788



789 **(6). Response & Request message 受信時の DRM サーバ処理**

- 790     > [MIPTV], 4.1.4.4 項の規定に従い、Response & Request message の検証を  
791     行い、セッション鍵を生成する。  
792     ✧ 検証が成功した場合、以下の処理を行い、(7)Reply message 送信時の  
793     DRM サーバ処理を実行する。  
794         ● SequenceNumber として、[MIPTV], 4.1.4.4 項で SequenceNumber  
795         の確認に用いた値よりも 1 大きい値である 2 を保持する。  
796         ● Response & Request message の TransactionFlag を保持する。  
797         ● Response & Request message から Service Protocol のメッセージ  
798         を抽出する。  
799     ✧ 検証が失敗した場合、(16)Plain command message 送信時の DRM サー  
800     バ処理を実行する。  
801

802 **(7). Reply message 送信時の DRM サーバ処理**

- 803     > Reply message を作成する。SequenceNumber、  
804     TransactionFlagRecordFlag、Reply、MessageDigest はセッション鍵で暗  
805     号化する。Reply message のメッセージパラメータに関する処理を以下に  
806     示す。  
807     ✧ SequenceNumber : 保持している SequenceNumber を用いる。  
808     ✧ TransactionFlagRecordFlag : 00h を設定する。  
809     ✧ Reply : Service Protocol のメッセージを設定する。  
810     ✧ MessageDigest : 暗号化前の MessageDigest を除く Reply message の  
811     パラメータから生成する。  
812     > Reply message 作成後に保持している SequenceNumber を 1 増加する。  
813     > Reply message を受信機に送信する。  
814     > Reply message 送信後のメッセージ受信待ちの状態に移る。  
815

816 **(8). Reply message 受信時の受信機処理**

- 817     > [MIPTV], 4.1.4.6 項の規定に従い、Reply message の検証を行う。  
818     ✧ 検証が成功した場合、以下の処理を行い、送信する Request がある場  
819     合は(9)の Request message 送信時の受信機処理を実行し、送信する  
820     Request がない場合は(11)Encrypted command message 送信時の受信  
821     機処理を実行する。  
822         ● 保持している SequenceNumber を 1 増加する。  
823         ● TransactionFlag の反転と保持を行う。現在、保持している  
824         TransactionFlag が even (00h) の場合は「odd (01h)」を、odd  
825         (01h) の場合は「even (00h)」を保持する。  
826         ● Reply message から Service Protocol のメッセージを抽出する。  
827     ✧ 検証が失敗した場合、SAC を終了して、SAC 開始前の状態に移る。  
828

829 **(9). Request message 送信時の受信機処理**

- 830     > Request message を作成する。SequenceNumber、TransactionFlag、  
831     Request、MessageDigest はセッション鍵で暗号化する。Request message  
832     のメッセージパラメータに関する処理を以下に示す。  
833     ✧ SequenceNumber : 保持している SequenceNumber を用いる。  
834     ✧ TransactionFlag : 保持している TransactionFlag を用いる。

- 835            ✧ Request : Service Protocol のメッセージを設定する。
- 836            ✧ MessageDigest : 暗号化前の MessageDigest を除く Request message
- 837                            のパラメータから生成する。
- 838            ➤ Request message 作成後に保持している SequenceNumber を 1 増加する。
- 839            ➤ Request message を DRM サーバに送信する。
- 840            ➤ Request message 送信後のメッセージ受信待ちの状態に移る。
- 841

842    **(10).Request message 受信時の DRM サーバ処理**

- 843            ➤ [MIPTV], 4.1.4.5項の規定に従い、Request messageの検証を行う。
- 844            ✧ 検証が成功した場合、以下の処理を行い、(7)Reply message 送信時の
- 845                            DRM サーバ処理を実行する
- 846                            ● 保持している SequenceNumber を 1 増加する。
- 847                            ● 保持している TransactionFlag を受信した Request message の
- 848    TransactionFlag に変更して保持する。
- 849                            ● Request message から Service Protocol のメッセージを抽出する。
- 850            ✧ 検証が失敗した場合、以下の処理を行い、(17)Command が
- 851                            「ERROR」の Encrypted command message 送信時の DRM サーバ処
- 852                            理を実行する。
- 853                            ● 保持している SequenceNumber を 1 増加する。
- 854

855    **(11).Encrypted command message 送信時の受信機処理**

- 856            ➤ Encrypted command messageを作成する。SequenceNumber、
- 857                            TransactionFlag、Command、Status、MessageDigestはセッション鍵で暗
- 858                            号化する。Encrypted command messageのメッセージパラメータに関する
- 859                            処理を以下に示す。
- 860            ✧ SequenceNumber : 保持している SequenceNumber を用いる。
- 861            ✧ Command : 「Commit」を設定する。
- 862            ✧ Status : 「Success (0000h)」を設定する。
- 863            ✧ MessageDigest : 暗号化前の MessageDigest を除く Encrypted
- 864                            command message のパラメータから生成する。
- 865            ➤ Encrypted command message 作成後に保持している SequenceNumber を
- 866                            1 増加する。
- 867            ➤ Encrypted command messageをDRMサーバに送信する。
- 868            ➤ Encrypted command message送信後のメッセージ受信待ちの状態に移る。
- 869

870    **(12).Encrypted command message 受信時の DRM サーバ処理**

- 871            ➤ [MIPTV], 4.1.4.8 項の規定に従い、Encrypted command message の検証を
- 872                            行う。
- 873            ✧ 検証が成功した場合、以下の処理を行い、(13)Command が「ACK」の
- 874                            Encrypted command message 送信時の DRM サーバ処理を実行する。
- 875                            ● 保持している SequenceNumber を 1 増加する
- 876            ✧ 検証が失敗した場合、以下の処理を行い、(17)Command が
- 877                            「ERROR」の Encrypted command message 送信時の DRM サーバ処
- 878                            理を実行する。
- 879                            ● 保持している SequenceNumber を 1 増加する
- 880

881 **(13).Command が「ACK」の Encrypted command message 送信時の DRM サーバ処理**

- 882     ➤ Encrypted command messageを作成する。SequenceNumber、  
883     TransactionFlag、Command、Status、MessageDigestはセッション鍵で暗  
884     号化する。Encrypted command messageのメッセージパラメータに関する  
885     処理を以下に示す。  
886     ✧ SequenceNumber：保持している SequenceNumber を用いる。  
887     ✧ Command：「ACK」を設定する。  
888     ✧ Status：「Success (0000h)」を設定する。  
889     ✧ MessageDigest：暗号化前の MessageDigest を除く Encrypted  
890     command message のパラメータから生成する。  
891     ➤ Encrypted command messageを受信機に送信する。  
892     ➤ SACを終了して、SAC開始前の状態に移る。  
893

894 **(14).Encrypted command message 受信時の受信機処理**

- 895     ➤ [MIPTV], 4.1.4.8項の規定に従い、Encrypted command messageの検証を行  
896     う。  
897     ➤ SACを終了して、SAC開始前の状態に移る。  
898

899 **(15).Plain command message 受信時の受信機処理**

- 900     ➤ [MIPTV], 4.1.4.7 項の規定に従い、Plain command message の検証を行う。  
901     ➤ SAC を終了して、SAC 開始前の状態に移る。  
902

903 **(16).Plain command message 送信時の DRM サーバ処理**

- 904     ➤ Plain command message を作成する。Plain command message のメッセー  
905     ジパラメータに関する処理を以下に示す。  
906     ✧ Command：「ERROR」を設定する。  
907     ✧ Status：メッセージの検証で確定した Status を設定する。  
908     ➤ Plain command messageを受信機に送信する。  
909     ➤ SACを終了して、SAC開始前のメッセージ受信待ちの状態に移る。  
910

911 **(17).Command が「ERROR」の Encrypted command message 送信時の DRM サーバ処**  
912 **理**

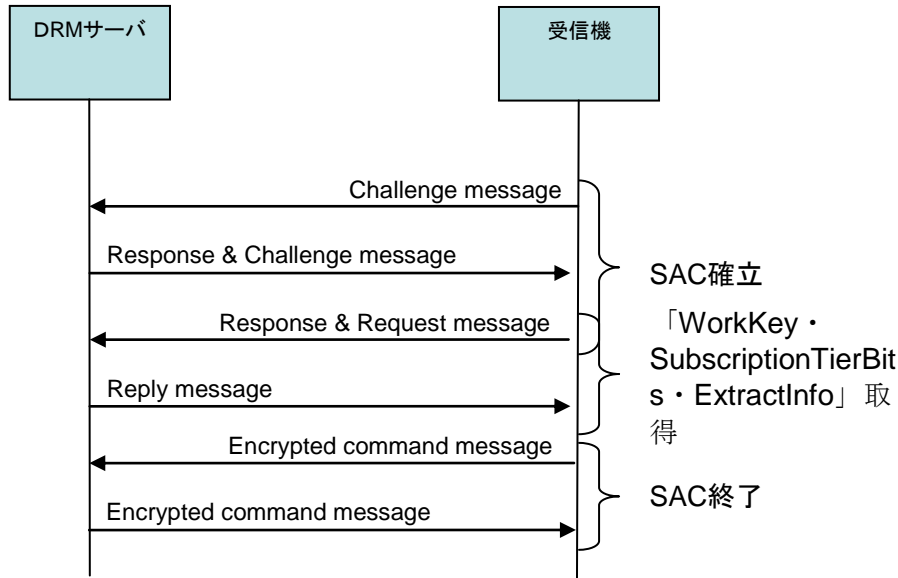
- 913     ➤ Encrypted command messageを作成する。SequenceNumber、  
914     TransactionFlag、Command、Status、MessageDigestはセッション鍵で暗  
915     号化する。Encrypted command messageのメッセージパラメータに関する  
916     処理を以下に示す。  
917     ✧ SequenceNumber：保持している SequenceNumber を用いる。  
918     ✧ Command：「ERROR」を設定する。  
919     ✧ Status：メッセージの検証で確定した Status を設定する。  
920     ✧ MessageDigest：暗号化前の MessageDigest を除く Encrypted  
921     command message のパラメータから生成する。  
922     ➤ Encrypted command messageを受信機に送信する。  
923     ➤ SACを終了して、SAC開始前のメッセージ受信待ちの状態に移る。  
924

## 925 **A.2 SAC と Service Protocol を用いたシーケンス**

926 本節では、SAC と Service Protocol を用いたシーケンスとして、「WorkKey ・  
927 SubscriptionTierBits ・ ExtractInfo」取得を解説する。

928 図 A-3 に DRM サーバ・受信機間の「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」  
929 取得シーケンスを示す。

930  
931



932

図 A-3 「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」取得シーケンス

933

934 「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」取得シーケンスに先立ち、受信機は  
935 UsageRuleReference と DRM サーバの URI を取得・保持する。なお、  
936 UsageRuleReference や DRM サーバの URI の取得に関する仕様は本書では規定し  
937 ない。

938

939 1. SAC 確立 : [MIPTV], 4.1 節 Secure Authenticated Channel (SAC) Protocol で規  
940 定されるプロトコルにより、受信機は DRM サーバとの間で相互認証を行い、  
941 SAC を確立する。

942 2. 「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」取得 : [MIPTV], 4.2.1 項 Get  
943 Permission Protocol で規定されるプロトコルにより、受信機は「WorkKey ・  
944 SubscriptionTierBits ・ ExtractInfo」取得要求を行い、DRM サーバから  
945 「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」を取得する。なお、  
946 「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」取得と Datetime を同時に取得  
947 する場合、[MIPTV], 4.2.3 項 Packed Message Protocol で規定されるプロトコ  
948 ルを用いる。

949 3. SAC 終了 : 「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」取得後、[MIPTV],  
950 4.1 節 Secure Authenticated Channel (SAC) Protocol で規定されるプロトコ  
951 ルにより、DRM サーバは Command が ACK の Encrypted command message を  
952 送信した後に、受信機は Command が ACK の Encrypted command message  
953 を受信した後に、SAC を終了する。

954

955 以上の「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」取得シーケンスにより、受信  
956 機は「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」を取得する。

957 その後、受信機は、コンテンツサーバから地上デジタルテレビジョン放送 IP 再送信  
958 コンテンツをストリーミングで受ける。コンテンツに多重化された ECM は  
959 WorkKey で復号され、ECM に含まれるスクランブル鍵を用いてコンテンツは復号  
960 されて再生される。この際、ECM に含まれる RenderingObligation に従った処理を  
961 行う必要がある。

962 また、WorkKey の扱いに関しては、[RTDBCR], 4 章の規定を参照のこと。  
963

### 964 A.3 コンテンツの利用シーケンス

965 本節では、コンテンツの利用に関して、受信機と DRM サーバとの間、および受信  
966 機とコンテンツを配信するコンテンツサーバとの間のシーケンスの概要について説  
967 明する。

968 コンテンツの利用シーケンスは、以下に示す 3 つの処理により構成される。

969 ① 「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」の取得処理

970 ② コンテンツ受信時の ECM 処理

971 ③ 「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」の更新処理

972 なお、以下の処理の事前に契約処理が行われ、受信機が「WorkKey ・  
973 SubscriptionTierBits ・ ExtractInfo」の取得のための UsageRuleReference ・ DRM サ  
974 ーバの URI を取得済みであることを前提とする。

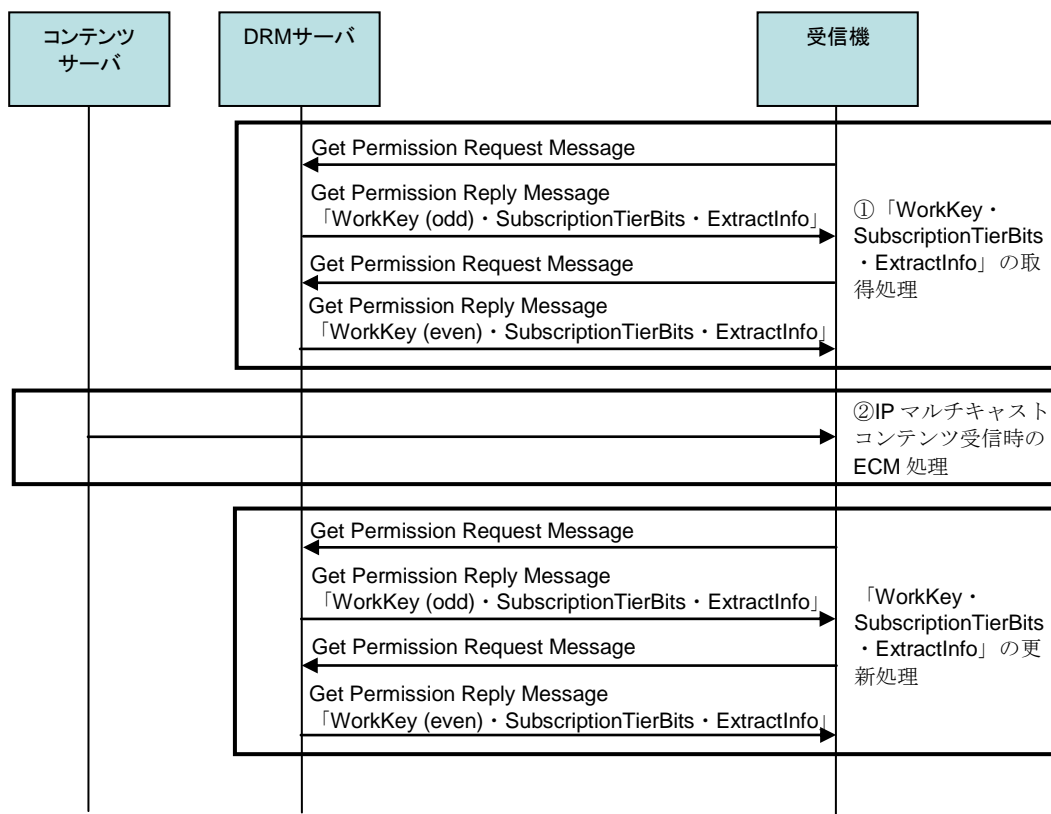


図 A-4 コンテンツの利用シーケンス

### 1001 **A.3.1 「WorkKey・SubscriptionTierBits・ExtractInfo」の取得処理**

1002 受信機は、[MIPTV], 4.1 節で規定される Secure Authenticated Channel (SAC)  
1003 Protocol により SAC を確立した上で、[MIPTV], 4.2.1 項で規定される Get  
1004 Permission Protocol により、DRM サーバから「WorkKey・SubscriptionTierBits・  
1005 ExtractInfo」を取得する。

1006 なお、Get Permission Protocol によるメッセージシーケンスは、A.2 節の  
1007 「WorkKey・SubscriptionTierBits・ExtractInfo」取得シーケンスと同様である。  
1008

#### 1009 **A.3.1.1ティアビット**

1010 「WorkKey・SubscriptionTierBits・ExtractInfo」に関して、DRM サーバと受信機と  
1011 の間の通信回数・受信機での保持数の削減のため、サービス事業者の 1 以上のチャ  
1012 ネルに対して同一の WorkKey を適用する運用が想定される。そこで、受信機が有す  
1013 る契約に応じたチャンネルの視聴制御のため、ティアビットとよばれる 64 ビットのビ  
1014 ット列を用いる。ティアビットの各ビットには、各チャンネルに対する契約が対応づ  
1015 けられる。この対応づけは、サービス事業者の運用依存である。  
1016 DRM サーバは、WorkKey とともに、ティアビットのうち受信機が有する契約に対  
1017 応するビットを示す“SubscriptionTierBits”を受信機に送信する。一方、ECM には、  
1018 ティアビットのうち当該チャンネルに対する契約を示す“ChannelTierBits”が設定さ  
1019 れる。受信機は、SubscriptionTierBits と ChannelTierBits との照合により、受信機が  
1020 契約を有するチャンネルのみを視聴可とするよう制御する。受信機でのティアビット  
1021 による視聴制御については A.3.2 項を参照のこと。  
1022

#### 1023 **A.3.1.2WorkKeyManagementID**

1024 A.3.1.1 項で説明したティアビットを用いる運用を実現するため、サービス事業者は、  
1025 複数チャンネルに適用する WorkKey と、当該複数チャンネルに対する契約を対応づけた  
1026 ティアビットとの対応を管理する。この WorkKey およびティアビットの管理単位を  
1027 特定する識別子を“WorkKeyManagementID”とよぶ。WorkKeyManagementID は、  
1028 サービス事業者ごと (“ServiceProviderID”で識別される) に割り当てる識別子で  
1029 ある。  
1030 受信機は、ServiceProviderID および WorkKeyManagementID の単位で、DRM サー  
1031 バから「WorkKey・SubscriptionTierBits・ExtractInfo」を取得する。したがって、受  
1032 信機が取得時に指定する UsageRuleReference には、ServiceProviderID と  
1033 WorkKeyManagementID とが含まれる。なお、WorkKeyManagementID および  
1034 UsageRuleReference の運用例については、A.4 節を参照のこと。  
1035

#### 1036 **A.3.1.3WorkKey (odd/even)**

1037 コンテンツに対して、一定期間などで ECM を暗号化する WorkKey を更新する運用  
1038 が想定される。  
1039 このような運用において、WorkKey の更新時でも受信機が ECM を連続して復号し、  
1040 継続視聴が可能となるように、DRM サーバは同時期に一对となる 2 つの WorkKey  
1041 を発行する。この一对となる 2 つの WorkKey を、“WorkKey (odd)” および  
1042 “WorkKey (even)” とよぶ。WorkKey (odd) と WorkKey (even) は、  
1043 WorkKeyID に含まれる WorkKeyVersion の LSB の値 (odd/evenID) により識別でき  
1044 る。WorkKeyVersion の詳細については[MIPTV], 4.2.1.5.1 項を、運用例については  
1045 A.4 節を参照のこと。

1046 また、DRM サーバは、一対の WorkKey (odd) ・ WorkKey (even) として、  
1047 WorkKey の送出時点で ECM を暗号化する WorkKey と、ECM を暗号化する  
1048 WorkKey の次回更新後の WorkKey とを送信する。  
1049 受信機は、2 つの Get Permission Protocol を用いて、DRM サーバから odd/even の  
1050 「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」を個別に取得する。したがって、  
1051 DRM サーバが、odd/even のどちらの WorkKey を発行すればよいかを特定できるよ  
1052 うに、UsageRuleReference には要求する WorkKey の odd/evenID の値が設定され  
1053 る。  
1054

#### 1055 **A.3.1.4受信機における「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」の管理**

1056 受信機は、ServiceProviderID ・ WorkKeyManagementID の単位で、取得した一対の  
1057 「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」を管理する。受信機は  
1058 「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」として、少なくとも WorkKey、  
1059 WorkKeyID、更新開始日時オフセット (PrivateData)、SubscriptionTierBits、  
1060 NotBefore/NotAfter を管理する。  
1061 [MIPTV], 4.2.4.4 項で規定される通り、受信機は、保持する WorkKey と  
1062 “ServiceProviderID”、“ReservedByte”、“WorkKeyManagementID”、および  
1063 “odd/evenID (WorkKeyVersion の LSB)” の4つの値が一致する WorkKey を新  
1064 たに取得した場合、取得した「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」で、保  
1065 持する「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」を更新する。このとき、受信  
1066 機は、ServiceProviderID ・ WorkKeyManagementID の単位での「WorkKey ・  
1067 SubscriptionTierBits ・ ExtractInfo」の管理にあわせて、一対の「WorkKey ・  
1068 SubscriptionTierBits ・ ExtractInfo」の単位で更新する。ただし、更新前後で値が同一  
1069 であるパラメータについては更新しなくても良い。  
1070 なお、取得した「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」を不揮発性記憶領域  
1071 に記録するか否かについては、受信機の実装依存である。ただし、受信機は、揮発  
1072 性記憶領域に保持する「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」の電源断など  
1073 による消失を防止し、DRM サーバからの再取得を不要とするため、不揮発性記憶領  
1074 域に記録することが望ましい。  
1075

#### 1076 **A.3.1.5RenderingObligation による EXTRACT ・ RECORD ・ EXPORT**

1077 受信機は、[MIPTV], 6.1.2 項で規定される ECM の RenderingObligation に基づき、  
1078 コンテンツの EXTRACT、RECORD、および EXPORT をおこなう。受信機は、  
1079 [MIPTV], 4.2.1 項で規定される RecordInfo および ExportInfo は取得しない。  
1080 RenderingObligation の送出に関する遵守規則、および、RenderingObligation に基づ  
1081 く受信機における EXTRACT ・ RECORD ・ EXPORT に関する遵守規則については、  
1082 [RTDBCR], 2 章を参照のこと。  
1083

#### 1084 **A.3.1.6Packed Message Protocol による「WorkKey ・ SubscriptionTierBits ・ 1085 ExtractInfo」の取得**

1086 受信機は、[MIPTV], 4.2.3 項で規定される Packed Message Protocol を用いて、1 ま  
1087 たは複数の “一対の「WorkKey (odd) ・ SubscriptionTierBits ・ ExtractInfo」と  
1088 「WorkKey (even) ・ SubscriptionTierBits ・ ExtractInfo」” とを同時に取得するこ  
1089 とができる。

1090 また、受信機は、1 または複数の “一対の「WorkKey (odd) ・  
1091 SubscriptionTierBits ・ ExtractInfo」と「WorkKey (even) ・ SubscriptionTierBits ・  
1092 ExtractInfo」 ” とともに、Datetime を同時に取得することもできる。  
1093

### 1094 **A.3.2 コンテンツ受信時の ECM 処理**

1095 受信機は、コンテンツサーバからコンテンツを受信し、ECM を抽出する。受信機は、  
1096 ECM 毎に ECM の WorkKeyID で指定される WorkKey の NotBefore/NotAfter と  
1097 [RTDBCR], 3.2 節の受信機内で保持する時刻（以下、本項では “受信機内で保持す  
1098 る時刻” と記す）とを比較し、(NotBefore) ≤ (受信機内で保持する時刻) ≤  
1099 (NotAfter) を満たす場合、受信機は当該 WorkKey を用いて ECM を復号する。但  
1100 し、NotBefore の値が FFFFFFFFh の場合は、NotBefore と受信機内で保持する時刻  
1101 との比較は不要である。同様に NotAfter の値が FFFFFFFFh の場合は、NotAfter と  
1102 受信機内で保持する時刻との比較は不要である。  
1103 復号した ECM の検証後、[MIPTV], 6.1.3 項の規定の通り、受信機は ECM の  
1104 ChannelTierBits と WorkKey の SubscriptionTierBits とを照合し、同一位置のビット  
1105 が共に 1b となるビットが存在する場合に視聴可と判定する。  
1106 受信機は、視聴可と判定した場合、ECM から抽出した ScrambleKey によりコンテ  
1107 ンツを復号し、ECM に設定された RenderingObligation に従った利用をおこなう。  
1108

### 1109 **A.3.3 「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」の更新処理**

1110 受信機は、「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」が更新される運用の場合  
1111 に、保持する「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」の更新を制御する。  
1112

#### 1113 **A.3.3.1 更新の有無と更新開始日時オフセット**

1114 DRM サーバは、送信する「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」の  
1115 StatusExtension の PrivateData（上位 23 バイト目から上位 24 バイト目）の値によ  
1116 り、次回の更新の有無、および、NotAfter から次回の更新が可能となる日時（更新  
1117 開始日時）までのオフセット時間を受信機に通知する。この 2 バイトの値を “更新  
1118 開始日時オフセット” とよぶ。  
1119 DRM サーバは、次回の更新をおこなう場合には、送信する「WorkKey ・  
1120 SubscriptionTierBits ・ ExtractInfo」の更新開始日時オフセットの値に 0001h~FFFFh  
1121 （単位は “分”）を設定する。  
1122 一方、DRM サーバは、契約の解約時など、次回の更新をおこなわない場合には、更  
1123 新開始日時オフセットの値に 0000h を設定する。  
1124

#### 1125 **A.3.3.2 「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」の更新**

1126 「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」が更新される運用（更新開始日時オ  
1127 フセットの値が 0001h~FFFFh）の場合、受信機は更新開始日時以降に更新するこ  
1128 とができる。更新のための「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」の取得は、  
1129 A.3.1 項と同等である。  
1130 一方、DRM サーバは、更新開始日時以降に受信機から「WorkKey ・  
1131 SubscriptionTierBits ・ ExtractInfo」の取得要求を受信した場合、少なくとも NotAfter  
1132 の値を更新した「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」を送信する。



1133 「WorkKey・SubscriptionTierBits・ExtractInfo」の更新に関して、DRM サーバ・受  
1134 信機の実装は以下の事項が考慮されることが望ましい。

- 1135
- 1136 ● DRM サーバ
- 1137   ➢ 受信機が短期間に繰り返し更新しないように、DRM サーバが更新後の  
1138   NotAfter に設定する値は、（「WorkKey・SubscriptionTierBits・  
1139   ExtractInfo」の送信日時）＋（更新開始日時オフセット）よりも大きな値と  
1140   することが望ましい。
- 1141 ● 受信機
- 1142   ➢ NotAfter と更新開始日時オフセットとに基づいて更新するか否かについては、  
1143   受信機の実装依存であるが、更新された場合に継続して視聴できるように  
1144   するため、受信機は更新開始日時以降、速やかに更新することが望ましい。
- 1145   ➢ DRM サーバと受信機とで保持する時刻に誤差が生じている場合などは、  
1146   DRM サーバが NotAfter を更新する前に、受信機が更新のために  
1147   「WorkKey・SubscriptionTierBits・ExtractInfo」を取得してしまう場合が考  
1148   えられる。このような場合、受信機は短期間に繰り返し更新しないように、  
1149   間隔をあけて再度更新するなどの制御をおこなうことが望ましい。
- 1150

## 1151 **A.4 WorkKeyID および UsageRuleReference の運用例**

1152 本節では、WorkKeyID および UsageRuleReference の運用例を示す。

1153

### 1154 **A.4.1 ティアビット・WorkKeyID・UsageRuleReference の関係**

1155 サービス事業者が運用するティアビット、WorkKeyID、および  
1156 UsageRuleReference の関係について説明する。

1157

#### 1158 **A.4.1.1 WorkKeyID とティアビットとの関係**

1159 WorkKeyID は、“ServiceProviderID”、“ReservedByte”、  
1160 “WorkKeyManagementID”、および“WorkKeyVersion”から構成される。  
1161 WorkKeyID のうち、DRM サーバおよび受信機における WorkKey の管理単位となる  
1162 ID が、ServiceProviderID および WorkKeyManagementID である。サービス事業者  
1163 は、64 ビットからなるティアビットを管理し、当該ティアビットに対して  
1164 WorkKeyManagementID を対応づける。したがって、サービス事業者は、64 ビット  
1165 からなるティアビットを運用する個数分だけ、WorkKeyManagementID を管理する。  
1166 たとえば、サービス事業者が 64 ビットからなるティアビットを 1 個用いる運用をお  
1167 こなう場合、当該サービス事業者は 1 つの WorkKeyManagementID を管理する。ま  
1168 た、64 ビットからなるティアビットを 2 個用いて、64 を超える契約を当該 2 個の  
1169 ティアビットに対応づける運用をおこなう場合、サービス事業者は 2 つの  
1170 WorkKeyManagementID を管理する。  
1171 また、DRM サーバは、同一 ServiceProviderID・同一 WorkKeyManagementID の  
1172 WorkKey について、WorkKey 自体を更新する場合に限り、WorkKeyVersion を更新  
1173 する。すなわち、SubscriptionTierBits または ExtractInfo の値を更新する場合であつ  
1174 ても、WorkKey 自体を更新しない限り、WorkKeyVersion は更新しない。  
1175 WorkKeyID の概要については A.3.1.2 項および A.3.1.3 項を、詳細については  
1176 [MIPTV], 4.2.1.5.1 項を参照のこと。

1177

1178 **A.4.1.2 UsageRuleReference と WorkKeyID との関係**

1179 UsageRuleReference は、“ServiceProviderID”、“ReservedByte”、  
1180 “WorkKeyManagementID”、および“odd/evenID”から構成される。  
1181 ServiceProviderID・WorkKeyManagementIDには、UsageRuleReference に対応す  
1182 る WorkKey の ServiceProviderID・WorkKeyManagementID と同一の値が設定され  
1183 る。  
1184 また、odd/evenID には、UsageRuleReference に対応する WorkKey の  
1185 WorkKeyVersion の LSB と同一の値が設定される。

1186 **A.4.1.3 ティアビットと WorkKeyID・UsageRuleReference の値との関係の例**

1187 サービス事業者が運用するティアビットと、WorkKeyID・UsageRuleReference の  
1188 値との関係の例を、表 A-3 に示す。  
1189 ただし、WorkKeyID および UsageRuleReference の ReservedByte は固定値である  
1190 ので、表 A-3 では記載を省略する。  
1191 また、3.1.3 項で規定する通り、UsageRuleReference の下位 10 バイトの値はサー  
1192 ビス事業者の運用依存であるため、表 A-3 では上位 6 バイトの値のみを記載する。  
1193

表 A-3 サービス事業者が運用するティアビットと  
WorkKeyID・UsageRuleReference の値との関係の例

サービス事業者	サービス事業者が運用するティアビット		WorkKeyID の値：16 進表記			UsageRuleReference の値：16 進表記		
			ServiceProviderID	WorkKeyManagementID	WorkKeyVersion	ServiceProviderID	WorkKeyManagementID	odd/evenID
サービス事業者 1	1 個 (*1)	一つ目のティアビット	0001	0001	01, 03, ...(odd)	0001	0001	01(odd)
					02, 04, ...(even)			00(even)
サービス事業者 2	2 個 (*2)	一つ目のティアビット	0002	0001	01, 03, ...(odd)	0002	0001	01(odd)
					02, 04, ...(even)			00(even)
	二つ目のティアビット	0002	01, 03, ...(odd)	0002	0002	01(odd)		
			02, 04, ...(even)			00(even)		

1194  
1195 \*1：64 ビットからなるティアビットを 1 個用いる運用をおこなうサービス事業者の  
1196 例である。  
1197 \*2：64 ビットからなるティアビットを 2 個用いて、64 を超える契約を当該 2 個の  
1198 ティアビットに対応づける運用をおこなうサービス事業者の例である。

1199 **A.4.2 WorkKey を更新する運用における WorkKeyID と**  
1200 **UsageRuleReference との関係の例**

1201 DRM サーバは、ServiceProviderID・WorkKeyManagementID ごとに WorkKey のバ  
1202 ージョン (WorkKeyVersion) を管理する。DRM サーバは、受信機から WorkKey を  
1203 要求された場合、一対の WorkKey (odd)・WorkKey (even) として、WorkKey の  
1204 送信時点で ECM を暗号化する WorkKey と、ECM を暗号化する WorkKey の次回更  
1205 新後の WorkKey とを送信する。よって、DRM サーバは、ServiceProviderID・  
1206 WorkKeyManagementID ごとに少なくとも一対の WorkKeyVersion を管理する。  
1207  
1208 WorkKey を更新する運用において、ECM を暗号化する WorkKey の WorkKeyID の  
1209 値、Get Permission Protocol で DRM サーバから送信する WorkKey の WorkKeyID

1210 の値、Get Permission Protocol で WorkKey を要求する際の UsageRuleReference の  
 1211 値の関係の例を、表 A-4 に示す。  
 1212  
 1213 表 A-4 において、Get Permission Protocol で送信する WorkKey の WorkKeyVersion  
 1214 は、ECM を暗号化する WorkKey の更新以降に送信する値を示すものである。DRM  
 1215 サーバは、ECM を暗号化する WorkKey の更新後、Get Permission Protocol で送信  
 1216 する一対の WorkKey (odd) ・ WorkKey (even) も速やかに更新する。  
 1217  
 1218 なお、A.4.1.3 項の表 A-3 と同様に、表 A-4 において WorkKeyID および  
 1219 UsageRuleReference の ReservedByte は記載を省略する。また、  
 1220 UsageRuleReference の下位 10 バイトの値についても、同様に記載を省略する。  
 1221

表 A-4 WorkKey を更新する運用における  
WorkKeyID と UsageRuleReference との値の関係の例

ECM を暗号 化する WorkK ey の 更新 (*1)	ECM を暗号化する WorkKey の WorkKeyID の値 : 16 進表 記			Get Permission Protocol で送 信する WorkKey の WorkKeyID の値 : 16 進表記			UsageRuleReference の値 : 16 進表記							
	Service ProviderID	WorkK eyMan agement ID	WorkKey Version	Service Provide rID	WorkK eyMan agement ID	WorkKey Version	Service ProviderID	WorkK eyMan agement ID	odd/even ID					
運用 開始	0001	0001	01(odd)	0001	0001	01(odd)	0001	0001	01(odd)					
1 回目 の更新 後			02(even)			02(even)			03(odd)	00(even)				
			2 回目 の更新 後			03(odd)			02(even)	04(even)	01(odd)			
...						...			...	...				
254 回 目の更 新後			FF(odd)			FF(odd)			00(even)	00(even)				
			255 回 目の更 新後			00(even)			01(odd)	00(even)	01(odd)			

1222  
 1223 \*1 : 256 回目以降の更新では、表 A-4 における “運用開始～255 回目の更新後” の  
 1224 繰り返しとなる。  
 1225

## 1226 A.5 「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」の更新運 1227 用の例

1228 本節では、図 A-5 を用いて「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」の更新  
 1229 運用の例を示す。  
 1230 以下、図 A-5 のイベントの囲い数字、および、DRM サーバ ・ 事業者サーバの括弧書  
 1231 きの数字に対応して、新規契約から解約までの DRM サーバ ・ 事業者サーバ ・ 受信  
 1232 機の処理の概要について説明する。なお、事業者サーバは、契約 ・ 解約などをおこ  
 1233 なるサービス事業者のポータルサーバなどである。

1234 本節では、以下を前提とした更新運用の例を示す。

1235

- 1236 ● 特定の ServiceProviderID・WorkKeyManagementID の「WorkKey・
- 1237 SubscriptionTierBits・ExtractInfo」に関して、特定の受信機についての例を示す。
- 1238 ● WorkKey の期限 (NotBefore/NotAfter) を定期的に更新する、期限延長の運用を
- 1239 基本とする。したがって、更新の度に「WorkKey・SubscriptionTierBits・
- 1240 ExtractInfo」の全てが更新されるとは限らないことに留意されたい。
- 1241 ● 期限の更新周期は 1 年とする。DRM サーバは、受信機ごとの契約月の 1 年後に
- 1242 更新をおこなうため、NotAfter を契約月の 1 年後の月末日に設定する。なお、
- 1243 NotBefore には期限なし (FFFFFFFFh) を設定する。
- 1244 図 A-5 に示す NotAfter の年月日の表記は、受信機が取得する WorkKey の
- 1245 NotAfter の値を示し、点線の矢印は、受信機が当該 NotAfter の値を保持する期
- 1246 間を示す。
- 1247 ● 期限延長における更新開始日時は NotAfter の 14 日前とし、更新開始日時オフセ
- 1248 ットに 14 日 (20160 分) を設定する場合の例を示す。なお、以下では、更新開
- 1249 始日時から NotAfter までを“更新期間”と記す。
- 1250 図 A-5 に示す更新期間の年月日の表記は、受信機が NotAfter の値と更新開始日
- 1251 時オフセットとから算出する更新期間 (更新開始日時～NotAfter) を示し、点線
- 1252 の矢印は、受信機が当該更新期間の値を保持する期間を示す。
- 1253 ● ECM を暗号化する WorkKey の WorkKeyVersion の初期値は 1 とする。DRM サ
- 1254 ーバは、図 A-5 に示す“⑦ ECM を暗号化する WorkKey の更新”において、
- 1255 WorkKeyVersion を 1 から 2 に更新する。
- 1256 ● 事業者サーバは、契約時や解約時などに「WorkKey・SubscriptionTierBits・
- 1257 ExtractInfo」の取得のための UsageRuleReference・DRM サーバの URI (DRM
- 1258 サーバ URI) を含むファイルを提供する。以下、当該ファイルを“メタファイル
- 1259 ”と記す。
- 1260

年	2008												2009											
月	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3	4	5	6	7	8	9	10	11	12
イベント				△①新規契約												△②期限延長						△③契約追加		
DRMサーバ				(2)												(3)						(5)		
事業者サーバ				(1)																		(4)		
受信機				⇓												⇐						⇓		
NotAfter				2009/4/30												2010/4/30								
更新期間				2009/4/16~2009/4/30												2010/4/16~2010/4/30								

年	2010												2011											
月	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3	4	5	6	7	8	9	10	11	12
イベント				△④期限延長				△⑤契約一部解約申し込み	△⑥契約一部解約				△⑦ECMを暗号化する WorkKeyの更新			△⑧期限延長						△⑨全解約		
DRMサーバ				(6)				(8)	(9)							(10)						(12)		
事業者サーバ								(7)														(11)		
受信機				⇐				⇓	⇓							⇐						⇓		
NotAfter				2011/4/30				2010/9/3	2011/4/30							2012/4/30						2011/10/31		
更新期間				2011/4/16~2011/4/30				2010/9/1~2010/9/3	2011/4/16~2011/4/30							2012/4/16~2012/4/30								

1261  
1262

図 A-5 「WorkKey・SubscriptionTierBits・ExtractInfo」の更新運用の例

1263

1264 ① 新規契約

1265 受信機は、2008/4 に事業者サーバに新規に契約を申し込む。受信機は、事業者サー  
1266 バから取得したメタファイルを用いて、DRM サーバから一対の「WorkKey・  
1267 SubscriptionTierBits・ExtractInfo」を取得する。

1268  
1269 (1)事業者サーバからのメタファイルの取得

- 1270 ● 受信機は、事業者サーバに新規契約の要求を送信する。
- 1271 ● 事業者サーバは、受信機からの契約申し込みを受け付け、DRM サーバ  
1272 に契約内容を送信するなど、新規契約の受け付け処理を完了する。
- 1273 ● 受信機は、事業者サーバからメタファイルを取得し、不揮発性記憶領域  
1274 に記録する。

1275 (2)DRM サーバからの「WorkKey・SubscriptionTierBits・ExtractInfo」の取得

- 1276 ● 受信機は、メタファイルで指定される UsageRuleReference (odd) を  
1277 設定した Get Permission Request message を作成して、メタファイル  
1278 の DRM サーバ URI で指定される DRM サーバに送信する。
- 1279 ● DRM サーバは、UsageRuleReference (odd) に対応する「WorkKey  
1280 (odd)・SubscriptionTierBits・ExtractInfo」を設定した Get  
1281 Permission Reply message を作成して、受信機に送信する。
  - 1282 ➤ NotAfter には、加入月の 1 年後の月末の日時 (2009/4/30) を設定  
1283 する。
  - 1284 ➤ 送信する「WorkKey・SubscriptionTierBits・ExtractInfo」を次回に  
1285 更新することを通知するため、更新開始日時オフセットの値には  
1286 14 日 (20160 分) を設定する。
  - 1287 ➤ WorkKey (odd) には、WorkKey の送信時点で ECM を暗号化する  
1288 WorkKey (WorkKeyVersion が 1) を設定する。
- 1289 ● 受信機は、DRM サーバから受信した Get Permission Reply message か  
1290 ら「WorkKey (odd)・SubscriptionTierBits・ExtractInfo」を取得する。
- 1291 ● 受信機は、メタファイルで指定される UsageRuleReference (even) を  
1292 設定した Get Permission Request message を作成して、メタファイル  
1293 の DRM サーバ URI で指定される DRM サーバに送信する。
- 1294 ● DRM サーバは、UsageRuleReference (even) に対応する「WorkKey  
1295 (even)・SubscriptionTierBits・ExtractInfo」を設定した Get  
1296 Permission Reply message を作成して、受信機に送信する。
  - 1297 ➤ SubscriptionTierBits・ExtractInfo には、WorkKey (odd) の  
1298 SubscriptionTierBits・ExtractInfo と同一の値を設定する。また、更  
1299 新開始日時オフセットには、WorkKey (odd) の更新開始日時オフ  
1300 セットと同一の値を設定する。
  - 1301 ➤ WorkKey (even) には、ECM を暗号化する WorkKey の次回更新  
1302 後の WorkKey (WorkKeyVersion が 2) を設定する。
- 1303 ● 受信機は、DRM サーバから受信した Get Permission Reply message か  
1304 ら「WorkKey (even)・SubscriptionTierBits・ExtractInfo」を取得する。
- 1305 ● 受信機は、取得した一対の「WorkKey・SubscriptionTierBits・  
1306 ExtractInfo」を不揮発性記憶領域に記録する。
- 1307 ● 受信機は、更新開始日時オフセットの値が 14 日 (更新される) である  
1308 ことから、NotAfter と更新開始日時オフセットとを用いて次回の更新期  
1309 間 (②の更新期間: 2009/4/16~2009/4/30) を算出し、次回更新の制御  
1310 をおこなう。

1311  
1312  
1313

1314 [備考]

1315 • DRM サーバ

- 1316 ▶ DRM サーバは、WorkKey (odd) または WorkKey (even) のいずれかとし  
1317 て、WorkKey の送信時点で ECM を暗号化する WorkKey を送信する。他方  
1318 は、ECM を暗号化する WorkKey の次回更新後の WorkKey を送信する。  
1319 ▶ DRM サーバは、受信機が有する契約の終了日時を経過するまでは、受信機  
1320 に「WorkKey・SubscriptionTierBits・ExtractInfo」を送信する。送信回数は  
1321 制限しない。  
1322 ▶ DRM サーバは、更新開始日時オフセットまたは NotAfter の値を受信機ごと  
1323 などに異なる値とすることで、更新時の受信機からのアクセスを分散させ  
1324 ることができる。

1325 • 受信機

- 1326 ▶ メタファイルから取得した UsageRuleReference・DRM サーバ URI は、以  
1327 降の「WorkKey・SubscriptionTierBits・ExtractInfo」の更新時に必要となる  
1328 ため、受信機は不揮発性記憶領域に記録しておくことが望ましい。また、  
1329 この場合、受信機は新たにメタファイルを取得した場合、当該メタファイ  
1330 ルに含まれる DRM サーバ URI の値で、不揮発性記憶領域に記録した DRM  
1331 サーバ URI の値を更新することが望ましい。  
1332 ▶ 受信機は、同一 ServiceProviderID・同一 WorkKeyManagementID の  
1333 odd/even の「WorkKey・SubscriptionTierBits・ExtractInfo」を一对の組と  
1334 して取得する。メタファイルでは、一对の「WorkKey・  
1335 SubscriptionTierBits・ExtractInfo」に対応する 2 つの UsageRuleReference  
1336 (odd)・UsageRuleReference (even) を組として、1 組以上の  
1337 UsageRuleReference が指定される (1 サービス事業者が 64 ビットからなる  
1338 ティアビットを複数個用いる運用をおこなう場合などに、2 組以上の  
1339 UsageRuleReference が指定される場合がある)。

1340 受信機は、更新開始日時オフセットの値が 0000h (更新されない) である場合は、  
1341 更新をおこなわない。

1342

1343 ② 期限延長

1344 受信機は、②の更新期間 (2009/4/16~2009/4/30) に、期限延長された一对の  
1345 「WorkKey・SubscriptionTierBits・ExtractInfo」を取得する。

1346

1347 (3) DRM サーバからの「WorkKey・SubscriptionTierBits・ExtractInfo」の取得

- 1348 • 受信機は、不揮発性記憶領域に記録したメタファイルで指定される  
1349 UsageRuleReference (odd) を設定した Get Permission Request  
1350 message を作成して、メタファイルの DRM サーバ URI で指定される  
1351 DRM サーバに送信する。  
1352 • DRM サーバは、UsageRuleReference (odd) に対応する「WorkKey  
1353 (odd)・SubscriptionTierBits・ExtractInfo」を設定した Get  
1354 Permission Reply message を作成して、受信機に送信する。  
1355 ▶ NotAfter には、1 年後の月末の日時 (2010/4/30) を設定する。  
1356 ▶ 引き続き次回に更新をおこなうことを通知するため、更新開始日時  
1357 オフセットの値には 14 日 (20160 分) を設定する。  
1358 ▶ WorkKey (odd) には、WorkKey の送信時点で ECM を暗号化する  
1359 WorkKey (WorkKeyVersion が 1) を設定する。  
1360 • 受信機は、DRM サーバから受信した Get Permission Reply message か  
1361 ら「WorkKey (odd)・SubscriptionTierBits・ExtractInfo」を取得する。

- 1362
- 1363
- 1364
- 1365
- 1366
- 1367
- 1368
- 1369
- 1370
- 1371
- 1372
- 1373
- 1374
- 1375
- 1376
- 1377
- 1378
- 1379
- 1380
- 1381
- 1382
- 1383
- 1384
- 受信機は、メタファイルで指定される UsageRuleReference (even) を設定した Get Permission Request message を作成して、メタファイルの DRM サーバ URI で指定される DRM サーバに送信する。
  - DRM サーバは、UsageRuleReference (even) に対応する「WorkKey (even) ・ SubscriptionTierBits ・ ExtractInfo」を設定した Get Permission Reply message を作成して、受信機に送信する。
    - SubscriptionTierBits ・ ExtractInfo には、WorkKey (odd) の SubscriptionTierBits ・ ExtractInfo と同一の値を設定する。また、更新開始日時オフセットには、WorkKey (odd) の更新開始日時オフセットと同一の値を設定する。
    - WorkKey (even) には、ECM を暗号化する WorkKey の次回更新後の WorkKey (WorkKeyVersion が 2) を設定する。
  - 受信機は、DRM サーバから受信した Get Permission Reply message から「WorkKey (even) ・ SubscriptionTierBits ・ ExtractInfo」を取得する。
  - 受信機は、取得した一対の「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」で、不揮発性記憶領域に記録した同一 ServiceProviderID ・ 同一 WorkKeyManagementID の一対の「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」を更新する。
  - 受信機は、更新開始日時オフセットの値が 14 日 (更新される) であることから、NotAfter と更新開始日時オフセットとを用いて次回の更新期間 (④の更新期間：2010/4/16～2010/4/30) を算出し、次回更新の制御をおこなう。

1385 [備考]

- 1386
- 1387
- 1388
- 1389
- 1390
- 1391
- 1392
- 1393
- 1394
- 1395
- 1396
- DRM サーバ
    - DRM サーバは、更新開始日時 (上記では 2009/4/16) 以降に受信機から「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」の要求を受信した場合、NotAfter を更新 (上記では 2010/4/30) した「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」を送信する。
  - 受信機
    - メタファイルの DRM サーバ URI が変更される場合を考慮し、受信機は、期限延長時などにメタファイルを取得し、取得したメタファイルに設定された DRM サーバ URI の値で、不揮発性記憶領域に記録した DRM サーバ URI の値を更新することが望ましい。

1397 ③ 契約追加

1398 受信機は、2009/9 に事業者サーバに契約の一部追加を申し込む。受信機は一部の契

1399 約が追加された一対の「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」を取得する。

1400

1401 (4) 事業者サーバからのメタファイルの取得

- 1402
- 1403
- 1404
- 1405
- 1406
- 受信機は、事業者サーバに対して契約の一部追加の要求を送信する。
  - 事業者サーバは、受信機からの契約追加の申し込みを受け付け、DRM サーバに契約内容を送信するなど、契約追加の受け付け処理を完了する。
  - 受信機は、事業者サーバからメタファイルを取得し、受信したメタファイルで不揮発性記憶領域に記録したメタファイルを更新する。

1407 (5) DRM サーバからの「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」の取得

- 1408
- 1409
- 1410
- 受信機は、メタファイルで指定される UsageRuleReference (odd) を設定した Get Permission Request message を作成して、メタファイルの DRM サーバ URI で指定される DRM サーバに送信する。

- 1411           • DRM サーバは、UsageRuleReference (odd) に対応する「WorkKey  
1412            (odd)・SubscriptionTierBits・ExtractInfo」を設定した Get  
1413           Permission Reply message を作成して、受信機に送信する。  
1414           ➤ SubscriptionTierBits には、②で設定した SubscriptionTierBits のう  
1415           ち、追加した契約に対応するビットを 1b に変更したビット列を設定  
1416           する。  
1417           ➤ NotAfter には、引き続き②で設定した NotAfter (2010/4/30) と同  
1418           一の値を設定する。また、更新開始日時オフセットにも、引き続き  
1419           ②で設定した更新開始日時オフセット (14 日) と同一の値を設定  
1420           する。  
1421           ➤ WorkKey (odd) には、WorkKey の送信時点で ECM を暗号化する  
1422           WorkKey (WorkKeyVersion が 1) を設定する。  
1423           • 受信機は、DRM サーバから受信した Get Permission Reply message から  
1424           「WorkKey (odd)・SubscriptionTierBits・ExtractInfo」を取得する。  
1425           • 受信機は、メタファイルで指定される UsageRuleReference (even) を  
1426           設定した Get Permission Request message を作成して、メタファイル  
1427           の DRM サーバ URI で指定される DRM サーバに送信する。  
1428           • DRM サーバは、UsageRuleReference (even) に対応する「WorkKey  
1429           (even)・SubscriptionTierBits・ExtractInfo」を設定した Get  
1430           Permission Reply message を作成して、受信機に送信する。  
1431           ➤ SubscriptionTierBits・ExtractInfo には、WorkKey (odd) の  
1432           SubscriptionTierBits・ExtractInfo と同一の値を設定する。また、更  
1433           新開始日時オフセットには、WorkKey (odd) の更新開始日時オフ  
1434           セットと同一の値を設定する。  
1435           ➤ WorkKey (even) には、ECM を暗号化する WorkKey の次回更新  
1436           後の WorkKey (WorkKeyVersion が 2) を設定する。  
1437           • 受信機は、DRM サーバから受信した Get Permission Reply message から  
1438           「WorkKey (even)・SubscriptionTierBits・ExtractInfo」を取得する。  
1439           • 受信機は、取得した一対の「WorkKey・SubscriptionTierBits・  
1440           ExtractInfo」で、不揮発性記憶領域に記録した同一  
1441           ServiceProviderID・同一 WorkKeyManagementID の一対の  
1442           「WorkKey・SubscriptionTierBits・ExtractInfo」を更新する。  
1443           • 受信機は、更新開始日時オフセットの値が 14 日 (更新される) である  
1444           ことから、NotAfter と更新開始日時オフセットとを用いて次回の更新期  
1445           間 (④の更新期間：2010/4/16～2010/4/30) を算出し、次回更新の制御  
1446           をおこなう。

#### ④ 期限延長

- 1449 受信機は、④の更新期間 (2010/4/16～2010/4/30) に、期限延長された一対の  
1450 「WorkKey・SubscriptionTierBits・ExtractInfo」を取得する。

#### (6) DRM サーバからの「WorkKey・SubscriptionTierBits・ExtractInfo」の取得

- 1452           • 受信機は、不揮発性記憶領域に記録したメタファイルで指定される  
1453           UsageRuleReference (odd) を設定した Get Permission Request  
1454           message を作成して、メタファイルの DRM サーバ URI で指定される  
1455           DRM サーバに送信する。  
1456           • DRM サーバは、UsageRuleReference (odd) に対応する「WorkKey  
1457           (odd)・SubscriptionTierBits・ExtractInfo」を設定した Get  
1458           Permission Reply message を作成して、受信機に送信する。  
1459



- 1460           ➤ NotAfter には、1 年後の月末の日時 (2011/4/30) を設定する。  
1461           ➤ 引き続き次回に更新をおこなうことを通知するため、更新開始日時  
1462           オフセットの値には 14 日 (20160 分) を設定する。  
1463           ➤ WorkKey (odd) には、WorkKey の送信時点で ECM を暗号化する  
1464           WorkKey (WorkKeyVersion が 1) を設定する。  
1465           ● 受信機は、DRM サーバから受信した Get Permission Reply message から  
1466           「WorkKey (odd) ・ SubscriptionTierBits ・ ExtractInfo」を取得する。  
1467           ● 受信機は、メタファイルで指定される UsageRuleReference (even) を  
1468           設定した Get Permission Request message を作成して、メタファイル  
1469           の DRM サーバ URI で指定される DRM サーバに送信する。  
1470           ● DRM サーバは、UsageRuleReference (even) に対応する「WorkKey  
1471           (even) ・ SubscriptionTierBits ・ ExtractInfo」を設定した Get  
1472           Permission Reply message を作成して、受信機に送信する。  
1473           ➤ SubscriptionTierBits ・ ExtractInfo には、WorkKey (odd) の  
1474           SubscriptionTierBits ・ ExtractInfo と同一の値を設定する。また、更  
1475           新開始日時オフセットには、WorkKey (odd) の更新開始日時オフ  
1476           セットと同一の値を設定する。  
1477           ➤ WorkKey (even) には、ECM を暗号化する WorkKey の次回更新  
1478           後の WorkKey (WorkKeyVersion が 2) を設定する。  
1479           ● 受信機は、DRM サーバから受信した Get Permission Reply message から  
1480           「WorkKey (even) ・ SubscriptionTierBits ・ ExtractInfo」を取得する。  
1481           ● 受信機は、取得した一対の「WorkKey ・ SubscriptionTierBits ・  
1482           ExtractInfo」で、不揮発性記憶領域に記録した同一  
1483           ServiceProviderID ・ 同一 WorkKeyManagementID の一対の  
1484           「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」を更新する。  
1485           ● 受信機は、更新開始日時オフセットの値が 14 日 (更新される) である  
1486           ことから、NotAfter と更新開始日時オフセットとを用いて次回の更新期  
1487           間 (⑧の更新期間：2011/4/16~2011/4/30) を算出し、次回更新の制御  
1488           をおこなう。  
1489

#### ⑤ 契約一部解約申し込み

- 1491 受信機は、2010/8 初旬に事業者サーバに一部の契約の解約を申し込む。DRM サー  
1492 バは、解約するチャンネルが当月末まで視聴可能となるように、SubscriptionTierBits  
1493 は解約前と同一で、期限が来月初めに設定された一対の「WorkKey ・  
1494 SubscriptionTierBits ・ ExtractInfo」を受信機に送信する。  
1495 受信機は、当該「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」を取得し、NotAfter  
1496 に基づき、一部の契約が解約された一対の「WorkKey ・ SubscriptionTierBits ・  
1497 ExtractInfo」を、来月初めに取得するための制御をおこなう。  
1498

#### (7) 事業者サーバからのメタファイルの取得

- 1500           ● 受信機は、事業者サーバに対して契約の一部解約の要求を送信する。  
1501           ● 事業者サーバは、受信機からの契約一部解約の申し込みを受け付け、  
1502           DRM サーバに解約内容を送信するなど、契約一部解約の受付処理を完  
1503           了する。  
1504           ● 受信機は、事業者サーバからメタファイルを取得し、受信したメタファ  
1505           イルで不揮発性記憶領域に記録したメタファイルを更新する。  
1506  
1507  
1508

- 1509 (8)DRM サーバからの「WorkKey・SubscriptionTierBits・ExtractInfo」の取得  
1510 ● 受信機は、メタファイルで指定される UsageRuleReference (odd) を  
1511 設定した Get Permission Request message を作成して、メタファイル  
1512 の DRM サーバ URI で指定される DRM サーバに送信する。  
1513 ● DRM サーバは、UsageRuleReference (odd) に対応する「WorkKey  
1514 (odd)・SubscriptionTierBits・ExtractInfo」を設定した Get  
1515 Permission Reply message を作成して、受信機に送信する。  
1516 ➤ SubscriptionTierBits には、引き続き、③および④で設定した契約  
1517 の一部解約前の SubscriptionTierBits を設定する。  
1518 ➤ 2010/9/1 からの一部解約のための更新を指定し、かつ、更新期間  
1519 を 3 日間とするため、NotAfter には 2010/9/3、更新開始日時オフセ  
1520 ットには 3 日 (4320 分) を設定する。  
1521 ➤ WorkKey (odd) には、WorkKey の送信時点で ECM を暗号化する  
1522 WorkKey (WorkKeyVersion が 1) を設定する。  
1523 ● 受信機は、DRM サーバから受信した Get Permission Reply message か  
1524 ら「WorkKey (odd)・SubscriptionTierBits・ExtractInfo」を取得する。  
1525 ● 受信機は、メタファイルで指定される UsageRuleReference (even) を  
1526 設定した Get Permission Request message を作成して、メタファイル  
1527 の DRM サーバ URI で指定される DRM サーバに送信する。  
1528 ● DRM サーバは、UsageRuleReference (even) に対応する「WorkKey  
1529 (even)・SubscriptionTierBits・ExtractInfo」を設定した Get  
1530 Permission Reply message を作成して、受信機に送信する。  
1531 ➤ SubscriptionTierBits・ExtractInfo には、WorkKey (odd) の  
1532 SubscriptionTierBits・ExtractInfo と同一の値を設定する。また、更  
1533 新開始日時オフセットには、WorkKey (odd) の更新開始日時オフ  
1534 セットと同一の値を設定する。  
1535 ➤ WorkKey (even) には、ECM を暗号化する WorkKey の次回更新  
1536 後の WorkKey (WorkKeyVersion が 2) を設定する。  
1537 ● 受信機は、DRM サーバから受信した Get Permission Reply message か  
1538 ら「WorkKey (even)・SubscriptionTierBits・ExtractInfo」を取得する。  
1539 ● 受信機は、取得した一対の「WorkKey・SubscriptionTierBits・  
1540 ExtractInfo」で、不揮発性記憶領域に記録した同一  
1541 ServiceProviderID・同一 WorkKeyManagementID の一対の  
1542 「WorkKey・SubscriptionTierBits・ExtractInfo」を更新する。  
1543 ● 受信機は、更新開始日時オフセットの値が 3 日 (更新される) である  
1544 ことから、NotAfter と更新開始日時オフセットとを用いて次回の更新期  
1545 間 (⑥の更新期間：2010/9/1～2010/9/3) を算出し、次回更新の制御を  
1546 おこなう。

1547  
1548 ⑥ 契約一部解約

1549 受信機は、⑥の更新期間 (2010/9/1～2010/9/3) に、一部の契約が解約された一対  
1550 の「WorkKey・SubscriptionTierBits・ExtractInfo」を取得する。

- 1551  
1552 (9)DRM サーバからの「WorkKey・SubscriptionTierBits・ExtractInfo」の取得  
1553 ● 受信機は、不揮発性記憶領域に記録したメタファイルで指定される  
1554 UsageRuleReference (odd) を設定した Get Permission Request  
1555 message を作成して、メタファイルの DRM サーバ URI で指定される  
1556 DRM サーバに送信する。

- 1557           • DRM サーバは、UsageRuleReference (odd) に対応する「WorkKey  
1558            (odd)・SubscriptionTierBits・ExtractInfo」を設定した Get  
1559           Permission Reply message を作成して、受信機に送信する。  
1560           ➤ SubscriptionTierBits には、⑤で設定した SubscriptionTierBits のう  
1561           ち、解約した契約に対応するビットを 0b に変更したビット列を設定  
1562           する。  
1563           ➤ NotAfter には、⑧の更新期間での期限延長を継続するため、④の期  
1564           限延長で設定した NotAfter (2011/4/30) と同一の値を設定する。  
1565           また、更新開始日時オフセットにも、引き続き④で設定した更新開  
1566           始日時オフセット (14 日) と同一の値を設定する。  
1567           ➤ WorkKey (odd) には、WorkKey の送信時点で ECM を暗号化する  
1568           WorkKey (WorkKeyVersion が 1) を設定する。  
1569           • 受信機は、DRM サーバから受信した Get Permission Reply message から  
1570           「WorkKey (odd)・SubscriptionTierBits・ExtractInfo」を取得する。  
1571           • 受信機は、メタファイルで指定される UsageRuleReference (even) を  
1572           設定した Get Permission Request message を作成して、メタファイル  
1573           の DRM サーバ URI で指定される DRM サーバに送信する。  
1574           • DRM サーバは、UsageRuleReference (even) に対応する「WorkKey  
1575           (even)・SubscriptionTierBits・ExtractInfo」を設定した Get  
1576           Permission Reply message を作成して、受信機に送信する。  
1577           ➤ SubscriptionTierBits・ExtractInfo には、WorkKey (odd) の  
1578           SubscriptionTierBits・ExtractInfo と同一の値を設定する。また、更  
1579           新開始日時オフセットには、WorkKey (odd) の更新開始日時オフ  
1580           セットと同一の値を設定する。  
1581           ➤ WorkKey (even) には、ECM を暗号化する WorkKey の次回更新  
1582           後の WorkKey (WorkKeyVersion が 2) を設定する。  
1583           • 受信機は、DRM サーバから受信した Get Permission Reply message から  
1584           「WorkKey (even)・SubscriptionTierBits・ExtractInfo」を取得する。  
1585           • 受信機は、取得した一対の「WorkKey・SubscriptionTierBits・  
1586           ExtractInfo」で、不揮発性記憶領域に記録している同一  
1587           ServiceProviderID・同一 WorkKeyManagementID の一対の  
1588           「WorkKey・SubscriptionTierBits・ExtractInfo」を更新する。  
1589           • 受信機は、更新開始日時オフセットの値が 14 日 (更新される) である  
1590           ことから、NotAfter と更新開始日時オフセットとを用いて次回の更新期  
1591           間 (⑧の更新期間: 2011/4/16~2011/4/30) を算出し、次回更新の制御  
1592           をおこなう。

1593  
1594 [備考]

- 1595           • DRM サーバ  
1596           ➤ DRM サーバは、「WorkKey・SubscriptionTierBits・ExtractInfo」の送信に  
1597           対して、受信機から以下の SAC 層のメッセージを受信することにより、受  
1598           信機での「WorkKey・SubscriptionTierBits・ExtractInfo」の更新を確認でき  
1599           る。  
1600           ✧ Command に「Commit」が設定された Encrypted command message  
1601           ✧ Request message  
1602           したがって、DRM サーバは、odd/even の一対の「WorkKey・  
1603           SubscriptionTierBits・ExtractInfo」の送信に対して、上記メッセージを受信  
1604           することにより契約の一部解約処理を完了できる。なお、上記は Get  
1605           Permission Protocol と Packed Message Protocol とで共通である。

1606 ⑦ ECM を暗号化する WorkKey の更新

1607 DRM サーバは、現在送出中の ECM を暗号化する WorkKey が更新された場合、Get  
1608 Permission Protocol で受信機に送信する WorkKey を更新する。

1609 DRM サーバは、ECM を暗号化する WorkKey を、WorkKeyVersion が 1 の WorkKey  
1610 (odd) から WorkKeyVersion が 2 の WorkKey (even) に更新する。以降、DRM サ  
1611 ーバは、WorkKeyVersion が 2 の WorkKey (even) と、WorkKeyVersion が 3 の  
1612 WorkKey (odd) とを受信機に送信する。

1613

1614 [備考]

1615 • DRM サーバは、ECM を暗号化する WorkKey の更新後、Get Permission  
1616 Protocol で送信する一対の WorkKey (odd) ・ WorkKey (even) も速やかに更  
1617 新する。

1618 • DRM サーバが WorkKey を更新するタイミングは、サービス事業者の運用依存  
1619 とする。

1620 • DRM サーバは、WorkKey 自体を定期的に更新する運用をおこなう場合、ECM  
1621 を暗号化する WorkKey の更新前に、全ての受信機が更新後に ECM の暗号化に  
1622 用いる WorkKey を取得できるように各受信機の更新期間を設定する。

1623

1624 ⑧ 期限延長

1625 受信機は、⑧の更新期間 (2011/4/16~2011/4/30) に、期限延長された一対の  
1626 「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」を取得する。

1627

1628 (10) DRM サーバからの「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」  
1629 の取得

1630 • 受信機は、不揮発性記憶領域に記録したメタファイルで指定される  
1631 UsageRuleReference (odd) を設定した Get Permission Request  
1632 message を作成して、メタファイルの DRM サーバ URI で指定される  
1633 DRM サーバに送信する。

1634 • DRM サーバは、UsageRuleReference (odd) に対応する「WorkKey  
1635 (odd) ・ SubscriptionTierBits ・ ExtractInfo」を設定した Get  
1636 Permission Reply message を作成して、受信機に送信する。

1637 ▶ NotAfter には、1 年後の月末の日時 (2012/4/30) を設定する。

1638 ▶ 引き続き次回に更新することを通知するため、更新開始日時オフセ  
1639 ットの値には 14 日 (20160 分) を設定する。

1640 ▶ WorkKey (odd) には、ECM を暗号化する WorkKey の次回更新後  
1641 の WorkKey (WorkKeyVersion が 3) を設定する。

1642 • 受信機は、DRM サーバから受信した Get Permission Reply message から  
1643 「WorkKey (odd) ・ SubscriptionTierBits ・ ExtractInfo」を取得する。

1644 • 受信機は、メタファイルで指定される UsageRuleReference (even) を  
1645 設定した Get Permission Request message を作成して、メタファイル  
1646 の DRM サーバ URI で指定される DRM サーバに送信する。

1647 • DRM サーバは、UsageRuleReference (even) に対応する「WorkKey  
1648 (even) ・ SubscriptionTierBits ・ ExtractInfo」を設定した Get  
1649 Permission Reply message を作成して、受信機に送信する。

1650 ▶ SubscriptionTierBits ・ ExtractInfo には、WorkKey (odd) の

1651 SubscriptionTierBits ・ ExtractInfo と同一の値を設定する。また、更  
1652 新開始日時オフセットには、WorkKey (odd) の更新開始日時オフ  
1653 セットと同一の値を設定する。

- 1654           ➤ WorkKey (even) には、WorkKey の送信時点で ECM を暗号化する  
1655           WorkKey (WorkKeyVersion が 2) を設定する。  
1656           ● 受信機は、DRM サーバから受信した Get Permission Reply message から「WorkKey (even) ・ SubscriptionTierBits ・ ExtractInfo」を取得する。  
1657           ● 受信機は、取得した一对の「WorkKey ・ SubscriptionTierBits ・  
1658           ExtractInfo」で、不揮発性記憶領域に記録した同一  
1659           ServiceProviderID ・ 同一 WorkKeyManagementID の一对の  
1660           「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」を更新する。  
1661           ● 受信機は、更新開始日時オフセットの値が 14 日（更新される）である  
1662           ことから、NotAfter と更新開始日時オフセットとを用いて次回の更新期  
1663           間（2012/4/16～2012/4/30、図 A-5 に図示せず）を算出し、次回更新の  
1664           制御をおこなう。  
1665

1666  
1667 ⑨ 全解約

1668 受信機は、2011/10 初旬に事業者サーバに当該 WorkKey に関する全契約の解約を申  
1669 し込む。受信機は、解約日を期限とした一对の「WorkKey ・ SubscriptionTierBits ・  
1670 ExtractInfo」を取得する。  
1671

1672 (11) 事業者サーバからのメタファイルの取得

- 1673           ● 受信機は、事業者サーバに対して全契約の解約の要求を送信する。  
1674           ● 事業者サーバは、受信機からの全解約申し込みを受け付け、DRM サー  
1675           バに解約内容を送信するなど、全解約の受付処理を完了する。  
1676           ● 受信機は、事業者サーバからメタファイルを取得し、受信したメタファ  
1677           イルで不揮発性記憶領域に記録したメタファイルを更新する。

1678 (12) DRM サーバからの「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」  
1679 の取得

- 1680           ● 受信機は、メタファイルで指定される UsageRuleReference (odd) を  
1681           設定した Get Permission Request message を作成して、メタファイル  
1682           の DRM サーバ URI で指定される DRM サーバに送信する。  
1683           ● DRM サーバは、UsageRuleReference (odd) に対応する「WorkKey  
1684           (odd) ・ SubscriptionTierBits ・ ExtractInfo」を設定した Get  
1685           Permission Reply message を作成して、受信機に送信する。  
1686           ➤ NotAfter には、解約月の月末の日時（2011/10/31）を設定する。  
1687           ➤ 当該受信機の以降の更新を停止するため、更新開始日時オフセット  
1688           の値には 0000h を設定する。  
1689           ➤ WorkKey (odd) には、ECM を暗号化する WorkKey の次回更新後  
1690           の WorkKey (WorkKeyVersion が 3) を設定する。  
1691           ● 受信機は、DRM サーバから受信した Get Permission Reply message から  
1692           「WorkKey (odd) ・ SubscriptionTierBits ・ ExtractInfo」を取得する。  
1693           ● 受信機は、メタファイルで指定される UsageRuleReference (even) を  
1694           設定した Get Permission Request message を作成して、メタファイル  
1695           の DRM サーバ URI で指定される DRM サーバに送信する。  
1696           ● DRM サーバは、UsageRuleReference (even) に対応する「WorkKey  
1697           (even) ・ SubscriptionTierBits ・ ExtractInfo」を設定した Get  
1698           Permission Reply message を作成して、受信機に送信する。  
1699           ➤ SubscriptionTierBits ・ ExtractInfo には、WorkKey (odd) の  
1700           SubscriptionTierBits ・ ExtractInfo と同一の値を設定する。また、更  
1701           新開始日時オフセットには、WorkKey (odd) の更新開始日時オフ  
1702           セットと同一の値を設定する。

- 1703           ➤ WorkKey (even) には、WorkKey の送信時点で ECM を暗号化する  
1704           WorkKey (WorkKeyVersion が 2) を設定する。  
1705           ● 受信機は、DRM サーバから受信した Get Permission Reply message から「WorkKey (even) ・ SubscriptionTierBits ・ ExtractInfo」を取得する。  
1706           ● 受信機は、取得した一対の「WorkKey ・ SubscriptionTierBits ・  
1707           ExtractInfo」で、不揮発性記憶領域に記録した同一  
1708           ServiceProviderID ・ 同一 WorkKeyManagementID の一対の  
1709           「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」を更新する。  
1710           ● 受信機は、更新開始日時オフセットの値が 0000h (更新されない) である  
1711           ことから、次回以降の更新はおこなわない。  
1712

1713

1714 [備考]

- 1715 ● DRM サーバ  
1716     ➤ DRM サーバは、「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」の送信に  
1717     対して、受信機から以下の SAC 層のメッセージを受信することにより、受  
1718     信機の「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」の更新を確認できる。  
1719     ☆ Command に「Commit」が設定された Encrypted command message  
1720     ☆ Request message  
1721     したがって、DRM サーバは、odd/even の一対の「WorkKey ・  
1722     SubscriptionTierBits ・ ExtractInfo」の送信に対して、上記メッセージを受信  
1723     することにより解約処理を完了できる。なお、上記は Get Permission  
1724     Protocol と Packed Message Protocol とで共通である。  
1725 ● 受信機  
1726     ➤ 「WorkKey ・ SubscriptionTierBits ・ ExtractInfo」の削除は、受信機の実装依  
1727     存とする。ただし、更新開始日時オフセットの値が 0000h (更新されな  
1728     い) である場合、NotAfter を経過した「WorkKey ・ SubscriptionTierBits ・  
1729     ExtractInfo」を削除することが望ましい。このとき、不揮発性記憶領域に記  
1730     録した、対応するメタファイルも削除することが望ましい。  
1731

## 1732 A.6 メッセージの例

### 1733 A.6.1 HTTP のメッセージの例

1734 本項では受信機と DRM サーバとの通信の HTTP ヘッダの例を SAC の処理毎に示す。  
1735 本項で示す例では、以下を想定している。

- 1736 ● DRM サーバのホスト名は [www.iptv.jp](http://www.iptv.jp)  
1737 ● Cookie は 0000000000000001

1738 なお、Cookie を用いない場合は、以降に示す例の Set-Cookie および Cookie は不要  
1739 である。

1740

1741 受信機から Challenge message を送信する場合

1742     POST / HTTP/1.1  
1743     Host: www.iptv.jp  
1744     Content-type: application/octet-stream  
1745     Content- Length: 1039  
1746

1747 DRM サーバから Response & Challenge message を送信する場合

1748     HTTP/1.1 200 OK  
1749     Set-Cookie: JSESSIONID=0000000000000001

1750 Cache-Control: no-cache  
 1751 Content-type: application/octet-stream  
 1752 Content- Length: 1321  
 1753  
 1754 受信機から Challenge message 以外のメッセージを送信する場合  
 1755 POST / HTTP/1.1  
 1756 Host: www.iptv.jp  
 1757 Cookie: JSESSIONID=0000000000000001  
 1758 Content-type: application/octet-stream  
 1759 Content- Length: 236  
 1760  
 1761 DRM サーバから Response & Challenge message 以外のメッセージを送信する場合  
 1762 HTTP/1.1 200 OK  
 1763 Cache-Control: no-cache  
 1764 Content-type: application/octet-stream  
 1765 Content- Length: 124  
 1766

1767 **A.6.2 SAC のメッセージの例**

1768 本項では SAC のメッセージの例を示す。  
 1769 なお、サイズの大きいパラメータ (SinkCertificate、SourceCertificate など) と演算  
 1770 により生成するパラメータ (SourceEC-DHPhase1Value、Signature など) は、値  
 1771 の記述を省略した。また、以下の表中のハッチング部に記載されている値は暗号化  
 1772 前の値を示す。  
 1773

1774 **A.6.2.1 Challenge message**

1775 1001byte の SinkCertificate を格納した Challenge message の例を表 A-5 に示す。  
 1776

表 A-5 Challenge message の例

Byte index	パラメータ名	値 : 16 進表記
0-3	ProtocolID	49505456 (固定値)
4-5	ProtocolVersion	0100 (固定値)
6-13	SenderID.	1001000000000000
14-15	PayloadType	0001 (固定値)
16-19	PayloadSize	0000034C
20-35	SinkRandomNumber	(省略)
36-37	SinkCertificateSize	03E9
38-1038	SinkCertificate	(省略)

1777

1778 **A.6.2.2 Response & Challenge message**

1779 1171byte の SourceCertificate を格納した Response & Challenge message の例を表  
 1780 A-6 に示す。  
 1781  
 1782  
 1783  
 1784  
 1785

表 A-6 Response & Challenge message の例

Byte index	パラメータ名	値：16 進表記
0-3	ProtocolID	49505456 (固定値)
4-5	ProtocolVersion	0100 (固定値)
6-13	SenderID	0000000000000000 (固定値)
14-15	PayloadType	0002 (固定値)
16-19	PayloadSize	000003BC
20-35	SourceRandomNumber	(省略)
36-91	SourceEC-DHPhase1Value	(省略)
92-147	Signature	(省略)
148-149	SourceCertificateSize	0493
150-1320	SourceCertificate	(省略)

1786

1787 **A.6.2.3 Response & Request message**

1788 GetPermission Request を格納した Response & Request message の例を表 A-7 に  
1789 示す。

1790

表 A-7 Response & Request message の例

Byte index	パラメータ名	値：16 進表記
0-3	ProtocolID	49505456 (固定値)
4-5	ProtocolVersion	0100 (固定値)
6-13	SenderID	1001000000000000
14-15	PayloadType	0003 (固定値)
16-19	PayloadSize	000000D8
20-75	SinkEC-DHPhase1Value	(省略)
76-131	Signature	(省略)
132-135	EncryptedDataSize	00000064
136-138	SequenceNumber	000001 (固定値)
139	TransactionFlag	00 (固定値)
140-203	Request	A.6.3.1.2 項を参照のこと
204-235	MessageDigest	(省略)

1791

1792 **A.6.2.4 Request message**

1793 複数の Request を連続送信する場合の GetPermission Request を格納した最初の  
1794 Request message の例を表 A-8 に示す。

1795

1796

1797

1798

1799

1800

1801

1802



表 A-8 Request message の例

Byte index	パラメータ名	値：16 進表記
0-3	ProtocolID	49505456 (固定値)
4-5	ProtocolVersion	0100 (固定値)
6-13	SenderID	1001000000000000
14-15	PayloadType	0004 (固定値)
16-19	PayloadSize	00000068
20-23	EncryptedDataSize	00000064
24-26	SequenceNumber	000003
27	TransactionFlag	01
28-91	Request	A.6.3.1.2 項を参照のこと
92-123	MessageDigest	(省略)

1803

1804 **A.6.2.5 Reply message**

1805 Get Permission Reply を格納した Reply message の例を表 A-9 に示す。

1806

表 A-9 Reply message の例

Byte index	パラメータ名	値：16 進表記
0-3	ProtocolID	49505456 (固定値)
4-5	ProtocolVersion	0100 (固定値)
6-13	SenderID	0000000000000000 (固定値)
14-15	PayloadType	0005 (固定値)
16-19	PayloadSize	0000004A
20-23	EncryptedDataSize	00000046
24-26	SequenceNumber	000002
27	TransactionFlagRecordFlag	00
28-61	Reply	A.6.3.1.3 項を参照のこと
62-93	MessageDigest	(省略)

1807

1808 **A.6.2.6 Plain command message**

1809 Status として Error other than the below を格納した Plain command message の例  
1810 を表 A-10 に示す。

1811

1812

1813

1814

1815

1816

1817

1818

1819

1820

1821

1822

表 A-10 Plain command message の例

Byte index	パラメータ名	値：16 進表記
0-3	ProtocolID	49505456 (固定値)
4-5	ProtocolVersion	0100 (固定値)
6-13	SenderID.	0000000000000000 (固定値)
14-15	PayloadType	0006 (固定値)
16-19	PayloadSize	00000004 (固定値)
20-21	Command	0002 (固定値)
22-23	Status	8001

1823

1824 **A.6.2.7 Encrypted command message**

1825 Status として Error other than the below を格納した複数の Request を連続送信する  
1826 場合の最初の Request message に対する Encrypted command message の例を表  
1827 A-11 に示す。

1828

表 A-11 Encrypted command message の例

Byte index	パラメータ名	値：16 進表記
0-3	ProtocolID	49505456 (固定値)
4-5	ProtocolVersion	0100 (固定値)
6-13	SenderID.	0000000000000000 (固定値)
14-15	PayloadType	0007 (固定値)
16-19	PayloadSize	0000002C (固定値)
20-23	EncryptedDataSize	00000028 (固定値)
24-26	SequenceNumber	000004
27	TransactionFlag	00
28-29	Command	0002
30-31	Status	8001
32-63	MessageDigest	(省略)

1829

1830 **A.6.3 Service Protocol のメッセージ例**

1831 本項では[MIPTV], 4.2 節で規定される Service Protocol のメッセージの例を示す。

1832

1833 **A.6.3.1 Get Permission Protocol**

1834 本項では、[MIPTV], 4.2.1 項で規定される Get Permission Protocol のメッセージ例  
1835 を示す。

1836 **A.6.3.1.1. DeviceInformation**

1837 [MIPTV], 3.2.2 項で規定される DeviceInformation の例を表 A-12 に示す。この情報  
1838 は Get Permission Request メッセージに格納される。

1839

表 A-12 DeviceInformation の例

Byte index	パラメータ名	値：16進表記
0	Marlin IPTV-ES SpecificationVersionMajor	01 (固定値)
1	Marlin IPTV-ES SpecificationVersionMinor	00 (固定値)
2	Capabilities	00 (固定値)
3-4	Manufacturer	1001
5-6	ManufacturerModel	0000 (固定値)
7	ManufacturerModelVersion Major	00 (固定値)
8	ManufacturerModelVersion Minor	00 (固定値)
9-11	Reserved	000000 (固定値)

1840

1841 **A.6.3.1.2. Get Permission Request message**

1842 [MIPTV], 4.2.1.2 項で規定される Get Permission Request message の例を、表 A-13  
1843 に示す。

表 A-13 Get Permission Request message の例

バイト インデックス	パラメータ名	値：16進表記
0-1	ProtocolVersion	0100 (固定値)
2-3	MessageID	0001 (固定値)
4-15	DeviceInformation	A.6.3.1.1 項参照
16-31	UsageRuleReference	0001000001010000000000 0000000000
32	ActionID	02 (固定値)
33	ActionParameter	FF (固定値)
34-35	SpecificCRID	0001 (固定値)
36	PrivateDataTag	00 (固定値)
37-63	PrivateData	全て 00 (固定値)

1844

1845 **A.6.3.1.3. Get Permission Reply message**

1846 [MIPTV], 4.2.1.3 項で規定される Get Permission Reply message の例を、表 A-14 に  
1847 示す。

1848

表 A-14 Get Permission Reply message の例

バイト インデックス	パラメータ名	値：16進表記
0-1	ProtocolVersion	0100 (固定値)
2-3	MessageID	0002 (固定値)
4-5	Status	0000
6-21	WorkKey	00112233445566778899A ABBCCDDEEFF
22-27	WorkKeyID	000100000101
28-29	PrivateData	4EC0
30-37	SubscriptionTierBits	8000000000000000
38-39	ExtractInfoSize	0A (固定値)
40-43	NotBefore	FFFFFFFF
44-47	NotAfter	4B3CCAAF
48-49	RenderingObligation	0000 (固定値)

1849

1850 **A.6.3.2 Get Trusted Time Protocol**

1851 本項では、[MIPTV], 4.2.2 項で規定される Get Trusted Time Protocol のメッセージ  
1852 例を示す。

1853

1854 **A.6.3.2.1 Get TrustedTime Request**

1855 [MIPTV], 4.2.2.2 項で規定される Get Trusted Time Request メッセージの例を表  
1856 A-15 に示す。

1857

1858

表 A-15 *Get Trusted Time Request* メッセージの例

Byte index	パラメータ名	値：16 進表記
0-1	ProtocolVersion	0100 (固定値)
2-3	MessageID	0003 (固定値)

1859

1860 **A.6.3.2.2 *Get TrustedTime Reply***

1861 [MIPTV], 4.2.2.3 で規定される *Get Trusted Time Reply* メッセージの例を表 A-16 に  
1862 示す。

1863

表 A-16 *Get Trusted Time Reply* メッセージの例

Byte index	パラメータ名	値：16 進表記
0-1	ProtocolVersion	0100 (固定値)
2-3	MessageID	0004 (固定値)
4-5	Status	0000
6-9	Datetime	4B3CCABD

1864

1865 **A.6.3.3 Packed Message Protocol**

1866 本項では、[MIPTV], 4.2.3 項で規定される *Packed Message Protocol* のメッセージ例  
1867 を示す。

1868

1869 **A.6.3.3.1 *Packed Message Request message***

1870 [MIPTV], 4.2.3.2 項で規定される *Packed Message Request* に格納するリクエストメ  
1871 ッッセージが A.6.3.1.2 項の場合の例を、表 A-17 に示す。

1872

表 A-17 *Packed Message Request message* の例

バイト インデックス	パラメータ名	値：16 進表記
0-1	ProtocolVersion	0100 (固定値)
2-3	MessageID	0101 (固定値)
4-5	NumberOfRequest MessageBoxes	0002
6-7	RequestMessageSize	0040
8-71	RequestMessage	A.6.3.1.2 項参照
72-73	RequestMessageSize	0040
74-137	RequestMessage	A.6.3.1.2 項参照

1873

1874 **A.6.3.3.2 *Packed Message Reply message***

1875 [MIPTV], 4.2.3.3 項で規定される *Packed Message Reply message* で、A.6.3.3.1 項  
1876 に対して応答する場合の例を、表 A-18 に示す。

1877

1878

1879

表 A-18 Packed Message Reply message の例

バイト インデックス	パラメータ名	値 : 16 進表記
0-1	ProtocolVersion	0100 (固定値)
2-3	MessageID	0102 (固定値)
4-5	Status	0000
6-7	NumberOfReplyMessage Boxes	0002
8-9	ReplyMessageSize	0032
10-59	ReplyMessage	A.6.3.1.3 項参照
34-35	ReplyMessageSize	0032
36-85	ReplyMessage	A.6.3.1.3 項参照

1880