

# AI inside Cube コンソールツール API 説明書 (全文 OCR)

**Ver 1.0.0**

作成者	AI inside 株式会社
作成日	2024 年 11 月 11 日
最終更新日	2024 年 11 月 11 日

## 改訂履歴

改定日	版数	章・節・項	改定内容
2024/11/11	1.0.0		初版

## 目次

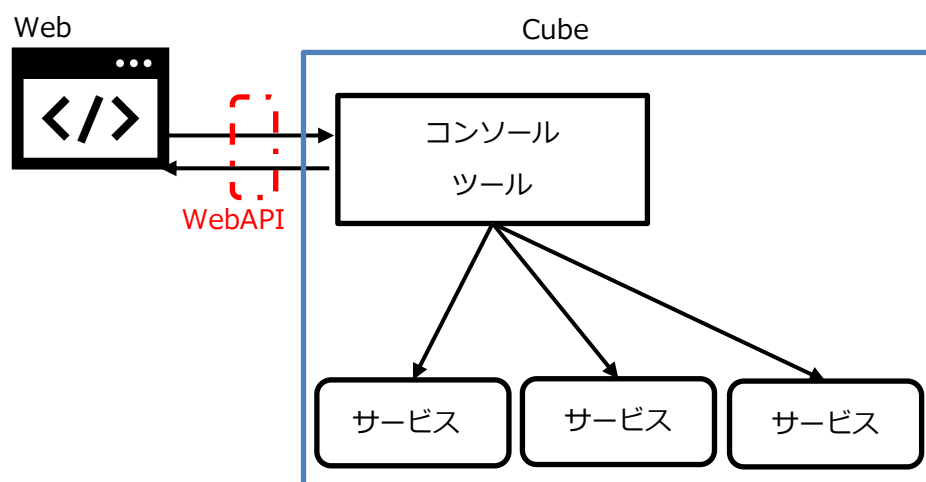
改訂履歴	1
目次	2
1 はじめに	3
2 API 概要一覧	4
3 API 共通の仕様	5
3.1 API の概要	5
3.2 リクエスト URI	5
3.3 認証	5
3.4 正常時の応答	6
3.5 エラー発生時の応答	6
4 各 API 個別の仕様	7
4.1 ウイルススキャン実行	7
4.2 MAC アドレス照会	8
4.3 サービスステータスチェック	9
4.4 サービス起動停止	11
4.5 サービス有効無効	13
4.6 clamav パターンファイルアップロード	15
4.7 CubeConsoleTool アクセスログダウンロード	17
4.8 Cube 状態監視	18
6 サポート窓口	21

## 1 はじめに

コンソールツール API は、Cube アプリケーション（コンソールツール）の制御を Web API から利用します。本書では、API 説明に関して記載いたします。

コンソールツールに関しましては、「AI inside Cube アプリケーション説明書（全文 OCR）」をご参照ください。

API に関しては、「2 API 概要一覧」をご参照ください。



## 2 API 概要一覧

提供している API は以下です。

API 名称	処理内容	レスポンス タイミング
ウイルススキャン実行	2:00AM に実行しているウイルススキャンを即時で実行します。	処理完了後
MAC アドレス参照	MAC アドレスを参照します。	処理完了後
ログダウンロード	各種ログファイルを zip 形式で取得します。	処理完了後
サービスステータスチェック	パラメータで指定されたサービスの状態を確認します。 <ul style="list-style-type: none"> <li>● 現在の起動状態（起動／停止）</li> <li>● デフォルトの起動設定（有効／無効）</li> </ul>	処理完了後
サービス起動停止	パラメータで指定されたサービスについて、以下の操作を実行します。 <ul style="list-style-type: none"> <li>● 起動</li> <li>● 停止</li> <li>● 再起動</li> </ul>	処理完了後
サービス有効無効	パラメータで指定されたサービスについて、以下の操作を実行します。 <ul style="list-style-type: none"> <li>● 有効化</li> <li>● 無効化</li> </ul>	処理完了後
ClamAV パターンファイルアップロード	ClamAV のパターンファイルをアップロードします。	処理完了後
Cube ConsoleTool アクセス ログダウンロード	Cube ConsoleTool のアクセスログをダウンロードします。	処理完了後

レスポンスタイミングについて

処理完了後：サービスの実行完了を待って API がレスポンスを返却します。

処理受付後：処理の受付後、API がレスポンスを返却します。

## 3 API 共通の仕様

### 3.1 API の概要

項目	説明
プロトコル	HTTP
コンテンツタイプ	multipart/form-data
レスポンスデータ形式	JSON フォーマット (Content-Type: text/html)
文字コード	UTF-8

### 3.2 リクエスト URI

コマンド実行による基本書式

```
$ curl -H "X-Cube-Apikey: <API キー>" -X <メソッド> "http:// <IP アドレス>:<ポート番号>/ <パス>"
```

項目	説明
X-Cube-Apikey	認証用ヘッダー (必須)
API キー	認証用 API キー (必須)
メソッド	POST メソッドを使用
IP アドレス	Cube に設定された IP アドレス
ポート番号	8000 番固定
パス	各 API のリクエスト URL 「4 各 API 個別の仕様」 参照

### 3.3 認証

認証には、32 バイトの 16 進数ランダム API キーを利用します。

API キーについては、AI inside Cube コンソールツールから発行をお願いします。

### 3.4 正常時の応答

正常に処理された結果は、コンテンツボディとして応答されます。コンテンツは、application/json となります。

正常時のコンテンツボディの形式は API 毎の仕様を参照してください。

### 3.5 エラー発生時の応答

エラー時は HTTP ステータスコード 4xx/5xx を応答します。エラーの詳細はコンテンツボディに JSON 形式で返されます。エラー発生時の JSON は以下の形式となります。

項目	必須	型	値
status	○	文字列	エラー情報
errorCode	○	文字列	エラー状態を示す ID （2 桁の整数からなる文字列）
message	○	文字列	エラーメッセージ
info		オブジェクト	処理対象の状態

## 4 各 API 個別の仕様

### 4.1 ウイルススキャン実行

ウイルススキャンを実施します。

パス	/Clamscan		
メソッド	POST		
レスポンス	正常の場合は、以下の構造の JSON。		
	項目	型	説明
	status	文字列	success : 成功
	errorCode	文字列	正常を示す ID
	message	文字列	メッセージ
	エラーの場合は、エラーコードを含む以下の構造の JSON。		
	項目	型	説明
	status	文字列	error : エラー
	errorCode	文字列	エラー状態を示す ID
	message	文字列	メッセージ
エラーコード	HTTP ステータスコードと、返却される JSON の errorCode にセットされるコードは以下です。		
	errorCode	説明	
	00	処理は正常に受け付けられた、又は、正常に終了した	
	01	対象サービスが既に実行している	
	10	対象サービスが見つからない	

コマンド実行例

```
$ curl -H "X-Cube-Apikey: dummy-key" -X POST "http://0.0.0.0:8000/Clamscan"
{
  "status": "success",
  "errorCode": "00",
  "message": ""
```



```
}
```

## 4.2 MAC アドレス照会

任意の NW インタフェースの MAC アドレスを取得します。

パス	/MACadd	
メソッド	POST	
リクエスト パラメータ	<b>device</b> パラメータに取得したいネットワークインタフェース名を指定	
	対象筐体	device 設定値
	Cube	eno1
	Cube Pro	eno1
	Cube Pro 4U / 4U+	eno1 eno2
※IPMI ポートには対応していません。		
レスポンス	正常の場合は、以下の構造の JSON。	
	項目	型 説明
	status	文字列 success : 成功
	errorCode	文字列 正常を示す ID
	message	文字列 取得した MAC アドレス
	エラーの場合は、エラーコードを含む以下の構造の JSON。	
	項目	型 説明
	status	文字列 error : エラー
	errorCode	文字列 エラー状態を示す ID
	message	文字列 メッセージ

エラーコード	HTTP ステータスコードと、返却される JSON の errorCode にセットされるコードは以下です。	
	errorCode	説明
	00	処理は正常に受け付けられた、又は、正常に終了した
	10	API キーが正しくありません
	11	ネットワークインタフェースが指定されていない
	20	mac アドレスが取得できなかった

#### コマンド実行例

```
$ curl -H "X-Cube-Apikey: dummy-key" -X POST -F "device=eno1"
"http://0.0.0.0:8000/MACadd"
{
  "status": "success",
  "errorCode": "00",
  "message": "3c:ec:ef:ba:64:de"
}
```

### 4.3 サービスステータスチェック

パラメータで指定されたサービスの状態を確認します。

現在の起動状態（起動／停止）とデフォルトの起動設定（有効／無効）が確認可能です。

パス	/ServiceStatus	
メソッド	POST	
リクエスト パラメータ	<b>svc</b> パラメータに状態確認を行いたいサービスを指定してください。 対象となるサービスが下記となります。	
	サービス名	説明
	FullocrAPI	AI-OCR を実行する為のサービス
	snmpd	監視サービス
	clamav-freshclam	アンチウィルスパターンファイル自動アップデートサービス

	systemd-timesyncd	時刻同期サービス		
レスポンス	正常の場合は、以下の構造の JSON。			
	項目		型	説明
	status		文字列	success : 成功
	errorCode		文字列	正常を示す ID
	message		文字列	メッセージ
	info	service_name	文字列	svc パラメータで指定したサービス名
		condition	文字列	現在の起動状態
	値			説明
	active			稼働中
	inactive			停止中
	default	文字列	デフォルトの起動設定値	
			値	説明
			enable	有効（サービス起動）
			disable	無効（サービス停止）
エラーの場合は、エラーコードを含む以下の構造の JSON。				
項目		型	説明	
status		文字列	error : エラー	
errorCode		文字列	エラー状態を示す ID	
message		文字列	メッセージ	
info	service_name	文字列	svc パラメータで指定したサービス名	
エラーコード	返却される JSON の errorCode にセットされるコードは以下です。			
	errorCode	説明		
	00	処理は正常に受け付けられた、又は、正常に終了した		
	10	API キーが正しくありません		
	30	サービス名の指定が正しくありません		

## コマンド実行例

```
$ curl -H "X-Cube-APIkey: dummy-key" -X POST -F "svc=FullocrAPI"
"http://0.0.0.0:8000/ServiceStatus"
{
  "status": "success",
  "errorCode": "00",
  "message": "",
  "info": {
    "service_name": "FullocrAPI",
    "condition": "active",
    "default": "enable"
  }
}
```

### 4.4 サービス起動停止

パラメータで指定されたサービスの操作を実施します。

サービスの起動／停止／再起動の操作が可能です。

パス	/ServiceStartStop	
メソッド	POST	
リクエスト パラメータ	<b>ctl</b> パラメータに下記の操作アクションを指定	
	操作内容	説明
	start	停止しているサービスを起動する
	stop	起動しているサービスを停止する
	restart	起動しているサービスを再起動する
	<b>svc</b> パラメータに操作を適応させるサービスを指定してください。 対象となるサービスは下記となります。	
	サービス名	説明
	FullocrAPI	AI-OCR を実行する為のサービス

	snmpd	監視サービス	
	clamav-freshclam	アンチウィルスパターンファイル自動アップデートサービス	
	systemd-timesyncd	時刻同期サービス	

レスポンス	正常の場合は、以下の構造の JSON。			
	項目		型	説明
	status		文字列	success : 成功
	errorCode		文字列	正常を示す ID
	message		文字列	メッセージ
	info	service_name	文字列	svc パラメータで指定したサービス名
		result	文字列	success
	エラーの場合は、エラーコードを含む以下の構造の JSON。			
	項目		型	説明
	status		文字列	error : エラー
	errorCode		文字列	エラー状態を示す ID
	message		文字列	メッセージ
	info	service_name	文字列	svc パラメータで指定したサービス名
		result	文字列	fail

エラーコード	返却される JSON の errorCode にセットされるコードは以下です。	
	errorCode	説明
	00	処理は正常に受け付けられた、又は、正常に終了した
	10	API キーが正しくありません
	20	ctl パラメータが正しくありません
	30	サービス名の指定が正しくありません
	40	その他のエラー

## コマンド実行例

```
$ curl -H "X-Cube-Apikey: dummy-key" -X POST -F "ctl=restart" -F "svc=FullocrAPI"
"http://0.0.0.0:8000/ServiceStartStop"
{
  "status": "success",
  "errorCode": "00",
  "message": "",
  "info": {
    "service_name": "FullocrAPI",
    "result": "success"
  }
}
```

## 4.5 サービス有効無効

パラメータで指定されたサービスのデフォルトの起動状態を変更します。

サービスのデフォルトの起動状態は有効／無効に変更可能です。

有効に変更した場合はサービスが起動し、無効に変更した場合はサービスが停止します。

パス	/ServiceEnableDisable	
メソッド	POST	
リクエスト パラメータ	<b>ctl</b> パラメータに下記の操作アクションを指定	
	操作内容	説明
	enable	サービスのデフォルト起動を有効にし、サービスを起動する
	disable	サービスのデフォルト起動を無効にし、サービスを停止する
	<b>svc</b> パラメータに操作を適応させるサービスを指定してください。 対象となるサービスは下記となります。	
	サービス名	説明
	snmpd	監視サービス
	clamav-freshclam	アンチウィルスパターンファイル自動アップデートサービス

	systemd-timesyncd	時刻同期サービス																							
レスポンス	正常の場合は、以下の構造の JSON。																								
	<table><tr><th colspan="2">項目</th><th>型</th><th>説明</th></tr><tr><td colspan="2">status</td><td>文字列</td><td>success : 成功</td></tr><tr><td colspan="2">errorCode</td><td>文字列</td><td>正常を示す ID</td></tr><tr><td colspan="2">message</td><td>文字列</td><td>メッセージ</td></tr><tr><td rowspan="2">info</td><td>service_name</td><td>文字列</td><td>svc パラメータで指定したサービス名</td></tr><tr><td>result</td><td>文字列</td><td>success</td></tr></table>		項目		型	説明	status		文字列	success : 成功	errorCode		文字列	正常を示す ID	message		文字列	メッセージ	info	service_name	文字列	svc パラメータで指定したサービス名	result	文字列	success
	項目		型	説明																					
	status		文字列	success : 成功																					
	errorCode		文字列	正常を示す ID																					
	message		文字列	メッセージ																					
	info	service_name	文字列	svc パラメータで指定したサービス名																					
		result	文字列	success																					
	エラーの場合は、エラーコードを含む以下の構造の JSON。																								
	<table><tr><th colspan="2">項目</th><th>型</th><th>説明</th></tr><tr><td colspan="2">status</td><td>文字列</td><td>error : エラー</td></tr><tr><td colspan="2">errorCode</td><td>文字列</td><td>エラー状態を示す ID</td></tr><tr><td colspan="2">message</td><td>文字列</td><td>メッセージ</td></tr><tr><td rowspan="2">info</td><td>service_name</td><td>文字列</td><td>svc パラメータで指定したサービス名</td></tr><tr><td>result</td><td>文字列</td><td>fail</td></tr></table>		項目		型	説明	status		文字列	error : エラー	errorCode		文字列	エラー状態を示す ID	message		文字列	メッセージ	info	service_name	文字列	svc パラメータで指定したサービス名	result	文字列	fail
項目		型	説明																						
status		文字列	error : エラー																						
errorCode		文字列	エラー状態を示す ID																						
message		文字列	メッセージ																						
info	service_name	文字列	svc パラメータで指定したサービス名																						
	result	文字列	fail																						
エラーコード	返却される JSON の errorCode にセットされるコードは以下です。																								
	errorCode	説明																							
	00	処理は正常に受け付けられた、又は、正常に終了した																							
	10	API キーが正しくありません																							
	20	ctl パラメータが正しくありません																							
	30	サービス名の指定が正しくありません																							
	40	その他のエラー																							

#### コマンド実行例

```
$ curl -H "X-Cube-Apikey: dummy-key" -X POST -F "ctl=enable" -F "svc=FullocrAPI"
"http://0.0.0.0:8000/ServiceEnableDisable"
```

## 4.6 clamav パターンファイルアップロード

アンチウイルスソフト（ClamAV）のパターンファイルをアップロードします。

パス	/ClamavPatternUpload																											
メソッド	POST																											
リクエスト パラメータ	File パラメータにアップロードするファイル名を設定。同時に 3 ファイルまで送信可能。 以下の名称のファイルのみ指定可能です。																											
	ファイル名																											
	bytecode.cld（または bytecode.cvd）																											
	daily.cld（または daily.cvd）																											
	main.cvd（または main.cld）																											
レスポンス	正常の場合は、以下の構造の JSON。																											
	<table><tr><th colspan="2">項目</th><th>型</th><th>説明</th></tr><tr><td colspan="2">status</td><td>文字列</td><td>success : 成功</td></tr><tr><td colspan="2">errorCode</td><td>文字列</td><td>正常を示す ID</td></tr><tr><td colspan="2">message</td><td>文字列</td><td>メッセージ</td></tr><tr><td rowspan="3">info</td><td>bytecode.cld version</td><td>文字列</td><td rowspan="2">“既存ファイルのバージョン番号”/“アップロー ドしたファイルのバージョン番号”</td></tr><tr><td>daily.cld version</td><td>文字列</td></tr><tr><td>main.cvd version</td><td>文字列</td><td>※アップロードしていない場合は “-” で表示される</td></tr></table>			項目		型	説明	status		文字列	success : 成功	errorCode		文字列	正常を示す ID	message		文字列	メッセージ	info	bytecode.cld version	文字列	“既存ファイルのバージョン番号”/“アップロー ドしたファイルのバージョン番号”	daily.cld version	文字列	main.cvd version	文字列	※アップロードしていない場合は “-” で表示される
	項目		型	説明																								
	status		文字列	success : 成功																								
	errorCode		文字列	正常を示す ID																								
	message		文字列	メッセージ																								
	info	bytecode.cld version	文字列	“既存ファイルのバージョン番号”/“アップロー ドしたファイルのバージョン番号”																								
		daily.cld version	文字列																									
		main.cvd version	文字列	※アップロードしていない場合は “-” で表示される																								
	エラーの場合は、エラーコードを含む以下の構造の JSON。																											
	<table><tr><th colspan="2">項目</th><th>型</th><th>説明</th></tr><tr><td colspan="2">status</td><td>文字列</td><td>error : エラー</td></tr><tr><td colspan="2">errorCode</td><td>文字列</td><td>エラー状態を示す ID</td></tr><tr><td colspan="2">message</td><td>文字列</td><td>メッセージ</td></tr></table>			項目		型	説明	status		文字列	error : エラー	errorCode		文字列	エラー状態を示す ID	message		文字列	メッセージ									
	項目		型	説明																								
status		文字列	error : エラー																									
errorCode		文字列	エラー状態を示す ID																									
message		文字列	メッセージ																									



エラーコード	返却される JSON の errorCode にセットされるコードは以下です。	
	errorCode	説明
	00	処理は正常に受け付けられた、又は、正常に終了した
	10	API キーが正しくありません
	20	アップロードされたファイルの数が正しくありません
	30	アップロードされたファイルが正しくありません
	40	その他のエラー

#### コマンド実行例

```
$ curl -H "X-Cube-Apikey: dummy-key" -X POST -F "file=@bytecode.cld" -F "file=@daily.cld" -F "file=@main.cvd" "http://0.0.0.0:8000/ClamavPatternUpload"
{
  "status": "success",
  "errorCode": "00",
  "message": "",
  "info": {
    "bytecode.cld version": "335/335",
    "daily.cld version": "27461/27461",
    "main.cvd version": "62/62"
  }
}
```

## 4.7 CubeConsoleTool アクセスログダウンロード

CubeConsoleTool のアクセスログをダウンロードします。

パス	/ConsoleToolAccessLog																						
メソッド	POST																						
リクエスト パラメータ	<p><b>date_from</b> パラメータに以下を設定</p> <p>date_from パラメータのみ指定した場合は、指定した日付から当日日付まで</p> <table><tr><td>ログ取得開始日の日付</td></tr><tr><td>YYYYMMDD 形式</td></tr></table> <p><b>date_to</b> パラメータに以下を設定</p> <p>date_to パラメータのみ指定した場合は、ログが存在する最初の日付から指定した日付まで</p> <table><tr><td>ログ取得終了日の日付</td></tr><tr><td>YYYYMMDD 形式</td></tr></table>	ログ取得開始日の日付	YYYYMMDD 形式	ログ取得終了日の日付	YYYYMMDD 形式																		
ログ取得開始日の日付																							
YYYYMMDD 形式																							
ログ取得終了日の日付																							
YYYYMMDD 形式																							
レスポンス	<p>正常の場合は、CSV ファイルがテキストで返却されます。</p> <p>適切に保存先の指定（〇〇〇〇.csv 等）を行ってください。</p> <p>※ エラー時のメッセージも保存先ファイルに出力されます。</p> <p>出力ファイルのフィールドは以下</p> <table><tr><th>項目</th><th>説明</th></tr><tr><td>id</td><td>ログ ID</td></tr><tr><td>created_at</td><td>作成タイムスタンプ（ログが生成された日時）</td></tr><tr><td>updated_at</td><td>更新タイムスタンプ（ログが更新された日時：通常は空白）</td></tr><tr><td>deleted_at</td><td>削除タイムスタンプ（ログが削除された日時：通常は空白）</td></tr><tr><td>action_started_at</td><td>操作日付（ログの操作が実行された日時）</td></tr><tr><td>user_id</td><td>ユーザーID（ログの操作を実行したユーザ名）</td></tr><tr><td>ip_address</td><td>IP アドレス（ログの操作を実行したクライアントの IP アドレス）</td></tr><tr><td>screen_name</td><td>画面名（ログの操作を実行した画面の名称）</td></tr><tr><td>action</td><td>操作内容（実行された操作の内容）</td></tr><tr><td>action_info</td><td>補足情報（その他の補足情報）</td></tr></table>	項目	説明	id	ログ ID	created_at	作成タイムスタンプ（ログが生成された日時）	updated_at	更新タイムスタンプ（ログが更新された日時：通常は空白）	deleted_at	削除タイムスタンプ（ログが削除された日時：通常は空白）	action_started_at	操作日付（ログの操作が実行された日時）	user_id	ユーザーID（ログの操作を実行したユーザ名）	ip_address	IP アドレス（ログの操作を実行したクライアントの IP アドレス）	screen_name	画面名（ログの操作を実行した画面の名称）	action	操作内容（実行された操作の内容）	action_info	補足情報（その他の補足情報）
項目	説明																						
id	ログ ID																						
created_at	作成タイムスタンプ（ログが生成された日時）																						
updated_at	更新タイムスタンプ（ログが更新された日時：通常は空白）																						
deleted_at	削除タイムスタンプ（ログが削除された日時：通常は空白）																						
action_started_at	操作日付（ログの操作が実行された日時）																						
user_id	ユーザーID（ログの操作を実行したユーザ名）																						
ip_address	IP アドレス（ログの操作を実行したクライアントの IP アドレス）																						
screen_name	画面名（ログの操作を実行した画面の名称）																						
action	操作内容（実行された操作の内容）																						
action_info	補足情報（その他の補足情報）																						

	エラーの場合は、エラーコードを含む以下の構造の JSON。		
	項目	型	説明
	status	文字列	error : エラー
	errorCode	文字列	エラー状態を示す ID
	message	文字列	メッセージ
エラーコード	返却される JSON の errorCode にセットされるコードは以下です。		
	errorCode	説明	
	10	API キーが正しくありません	

#### コマンド実行例

```
$ curl -H "X-Cube-Apikey: dummy-key" -X POST -F "date_from=20241001" -F "date_to=20241031" "http://0.0.0.0:8000/ConsoleToolAccessLog" -o ./access-log.csv
```

## 4.8 Cube 状態監視

Cube のアクティベーションの状況、ディスク容量、CPU 温度などの情報を取得します。

パス	/CubeStatus				
メソッド	POST				
レスポンス	正常の場合は、以下の構造の JSON。				
	項目			型	説明
	status			文字列	success : 成功 warning : 警告
	errorCode			文字列	正常を示す ID、または警告を示す ID
	message			文字列	メッセージ
	info	activation	result	文字列	OK : アクティベーション済み NG : 未アクティベーション
			expirationDate	文字列	アクティベーションの有効期限
		hw sensors	datetime	文字列	状態監視の時間

		cpuTemp	文字列	CPU 温度
		boardTemp	文字列	マザーボード温度
		gpuTemp	文字列	GPU 温度
		cpu%	文字列	CPU 使用率
		gpu%	文字列	GPU 使用率
		memFree%	文字列	空きメモリー率
		useStorate%	文字列	ストレージ使用率
		gpu2Temp	文字列	GPU 2 の温度（Cube を除く）
		gpu2%	文字列	GPU 2 の使用率（Cube を除く）

正常終了時、メッセージに表示される内容は下記の通り。

errorCode	message	説明
00	(なし)	正常終了、警告なし
01	The expiration date of activation is less than 20 days.	アクティベーションの有効期限が 20 日以内に到来します。
02	The useage of disk is above 70%.	ストレージの使用率が 70%を超えています。

エラーの場合は、エラーコードを含む以下の構造の JSON。

項目	型	説明
status	文字列	error : エラー
errorCode	文字列	エラー状態を示す ID
message	文字列	メッセージ

エラーコード

返却される JSON の errorCode にセットされるコードは以下です。

errorCode	説明
00	処理は正常に受け付けられた。警告なし。
01	警告（アクティベーションキーの有効期限が 20 日以内に到来する）
02	警告（ストレージの使用率が 70%を超えた）
10	API キーが正しくありません
20～70	その他のエラー

## コマンド実行例

```
$ curl -H "X-Cube-Apikey: dummy-key" -X POST "http://0.0.0.0:8000/CubeStatus"
{
  "status": "success",
  "errorCode": "00",
  "message": "",
  "info": {
    "activation": {
      "result": "OK",
      "expirationDate": "2025-10-01"
    },
    "hw sensors": {
      "datetime": "20241119 19:28:01",
      "cpuTemp": "32",
      "boardTemp": "36",
      "gpuTemp": "32",
      "cpu%": "1%",
      "gpu%": "0 %",
      "memFree%": "71.4%",
      "usedStorage%": "2%"
    }
  }
}
```

## 6 サポート窓口

お問い合わせ対応やサポートは、サービス仕様書をご参照ください。

このマニュアルには、AI inside 株式会社の秘密情報（ないし機密情報）が含まれています。  
同社による事前の許諾なく、これら情報の一部または全部を複製、使用、公開することはいかなる場合も  
禁じられております。