

mijin デプロイメントガイド

Release 1.3(日本語)

Techbureau, Corp

2025年07月28日

Contents:

1	Aboı	ut															1
	1.1	mijin	Catapult	v.2) について											•		1
		1.1.1	mijin C	atapult(v.2) とは	?										•		1
		1.1.2	ユース・												•		2
		1.1.3	パブリ	, クブロックチ ェ	· ーン Svm	bol $\mathcal{V}\sigma$	違い										4
	12	主 亜 -	ニータベー	ス製品との比較	ン ア 田 徐		~ • •	•••	•••	•••	•••		•••	•••	•	•	5
	1.2		・一一一一	へ表面との比較		• • • • •	• • • •	• • •	•••	•••	•••		• •	• •	•	•	5
	1 2	۲.۷.۱ د ب	土女ノ			• • • • •	• • • •	• • •	•••	•••	•••		• •	• •	•	•	5
	1.3	<u> デー</u> ヨ	- テクナキ	とスペック安件			• • • •	• • •	•••	•••	•••	•••	• •	• •	•	•	6
		1.3.1	mijin C	atapult(v.2)の構	成について	Ç		• • •	•••		•••		• •	• •	•	•	6
		1.3.2	PEER >	ードの役割とス	ペック要件	F					•••				•	•	6
		1.3.3	APIノ-	・ドの役割とスペ	ック要件										•	•	7
_	_																_
2	Depl	loy	о. I.														8
	2.1	mijin	Catapult	v.2) のテノロイフ	」法			• • •	•••	•••	••		•••	• •	•	•	8
		2.1.1	mijin Ø	デプロイ方法に	ついて .						•••		• •	• •	•	•	8
			2.1.1.1	AWS MarketPlac	ce を使用し	、たデプ	ロイ・								•	•	8
			2.1.1.2	テックビューロ	による構築	ē									•		8
	2.2	AWS	MarketPla	ce											•		8
		2.2.1	AWS M	irketplace を使っ	ったデプロ	イ準備											9
			2211	AWSアカウント	の進備						•••			• •	•	•	9
			2.2.1.1		•>		• • • •	•••	•••	•••	•••		•••	•••	•	•	q
			2.2.1.2	ヘルレႽ にデプロイ	オスための	\cdots	・・・ いト 佐		•••	•••	•••		• •	• •	•	•	11
			2.2.1.3	AWS にナノロイ)))))))) 	ノト催	. 219	•••	•••	•••		• •	• •	•	•	11
			2.2.1.4	AWSにおりるIT	iijin Catap	ult(V.2)	712	ンス	•••	•••	•••		• •	• •	•	•	13
			2.2.1.5	AWS利用料金	· • • • • • •			• • •	•••	•••	•••		•••	• •	•	•	13
			2.2.1.6	トライアル版							••		• •	• •	•	•	14
			2.2.1.7	製品版	. .						••				•	•	16
			2.2.1.8	有償サポートに [.]	ついて .										•		17
			2.2.1.9	AWSのサービス	クォータに	こよる制	限につ	いて							•		18
		2.2.2	新規VF	C を作成し、mii	inをデプロ	コイする											19
			2221	デプロイによっ	TAWS H	こ構築す	-るサー	-ビス-	-皆								20
			2222	View Network				<u> </u>	90	•••	•••		•••	•••	•	•	20
			2.2.2.2	Stop 1	, .	• • • • •	• • • •	• • •	•••	•••	••	•••	•••	• •	•	•	20
			2.2.2.3	Step.1	•••••	••••	• • • •	• • •	•••	•••	•••	•••	• •	• •	•	•	22
			2.2.2.4	Step.2	•••••	• • • • •		• • •	•••	•••	••	•••	• •	• •	•	•	22
			2.2.2.5	Step.3	· • • • • • •			• • •	•••	•••	••		•••	• •	•	•	23
			2.2.2.6	Step.4			• • • •	• • •	• • •		••		• •	• •	•	•	24
			2.2.2.7	Step.5	. .										•	•	25
			2.2.2.8	Step.6											•		26
			2.2.2.9	Step.7											•		27
			2.2.2.10	Step.8													30
			22211	Sten 9								•••	•••		•		31
			2.2.2.11	Stop 10	•••••	••••		• • •	•••	•••	•••	•••	•••	•••	•	•	37
			2.2.2.12	$Step. 10 \dots$		••••		• • •	•••	•••	••		• •	• •	·	•	J∠ 22
			Z.Z.Z.13	JCCD.TT	. .									• •	•	•	22

	2.2.2.14	Step.12	34
	2.2.2.15	mijin エンドポイントと確認項目	35
2.2.3	既存 VF	℃上に、mijin をデプロイする..................................	36
	2.2.3.1	デプロイによって AWS 上に構築するサービス一覧	36
	2232	Fire VPC のサブネットの作成	36
	2.2.3.2		36
	2.2.3.3		20
	2.2.3.4	Step.1	38
	2.2.3.5	Step.2	38
	2.2.3.6	Step.3	39
	2.2.3.7	Step.4	40
	2.2.3.8	Step.5	41
	2.2.3.9	Step.6	42
	2.2.3.10	Step.7	43
	22311	Step 8	47
	2 2 3 12	Step 9	48
	2.2.3.12	Stop 10	10
	2.2.3.13	Step 11	49
	2.2.3.14	Step.11	50
	2.2.3.15	Step.12	51
2.2.4	トフイ	アル版の mijin をテフロイする	53
	2.2.4.1	AWS 使用サービス	53
	2.2.4.2	View Network	53
	2.2.4.3	Step.1	54
	2.2.4.4	Step.2	54
	2.2.4.5	Step.3	55
	2246	Step 4	56
	2.2.1.0 2.2.1.7	Stop 5	56
	2.2.7.1	Stop 6	57
	2.2.4.0	Step.0	51
	2.2.4.9	Step.7	29
	2.2.4.10	Step.8	60
	2.2.4.11	Step.9	61
	2.2.4.12	Step.10	61
	2.2.4.13	Step.11	62
2.2.5	AWS M	arketPlace テクニカル資料	64
	2.2.5.1	AWS Marketplace Cloudformation パラメーター比較表	64
	2.2.5.2	AWS MarketPlace Cloudformation 什様	66
	2253	AWS MarketPlace mijin Catapult(v 2) アーキテクチャパターンによるリカ	
	2.2.0.0	バーI 能略	76
226	miiin (アフロスロート	20
2.2.0	2261	ataputt(v.2) アフロイ 夜の AWS 改定	00
	2.2.0.1		00 05
	2.2.6.2	mijin Catapuit(V.2)のノートストレーンの暗号化	85
	2.2.6.3	mijin Catapult(v.2)の定期的なノードのハックアップ	101
	2.2.6.4	mijin Catapult(v.2) 手数料ありモード有効時の残高アカウントの残高移動 .	103
	2.2.6.5	mijin Catapult(v.2) ノードの投票権ファイルの更新方法	109
	2.2.6.6	[アーカイブ] mijin Catapult(v.2) 手数料ありモード有効時の残高アカウン	
		トの残高移動	115
2.2.7	AWS の	トラブルシューティング	123
	2271	バックアップしたスナップショットからリストアする	123
	2272	アベイラビリティゾーン (A7) 暗実時の対応方法	129
228		ット・シーン・シーン (パー) r+日-マジンボルシリム ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	122
2.2.0	2201		122
	2.2.0.1		124
2 2 2	2.2.8.2	ノリートノ1 ブル版 FAU 一見	134
2.2.9	AWS M	arketPtace mijin Cataputt(V.2) 利用科金比較衣	132

3 Technical

ii

3.1	mijin Catapult(v.2)の基礎知識138
	3.1.1 mijin Catapult(v.2) のアクセス方法 138
	3.1.2 mijin Catapult(v.2) ステータス確認方法 139
	3.1.2.1 フロック高を確認する 139
	3.1.2.2 RESTのバージョンを確認する14(
	3.1.2.3 ノード情報を確認する 14(
	3.1.2.4 接続しているノードを確認する
	3.1.2.5 総トランザクション数、総アカワント数を確認する
	3.1.2.6 ネットワークダイフを確認する
	3.1.2.7 ノートのコンナナの状況を確認する14: 2.1.2.0 ブロックチューン(人体の))中た 変詞する
	3.1.2.8 ノロックナエーノ主体の設定を確認する
วา	3.1.2.9 トノノリンショノナ奴科を確認する143 milin Catapult(v_2) を撮作する
3.2	1111JIII Cataputi(V.2) を探FF 9 る 145
	5.2.1 IIIIJIII アカワンドF成 140
	$3.2.1.1$ Houels χO yall $\alpha \uparrow \gamma \land \land \land \neg \land \land$
	3.2.1.2 IIIIJII-Catapuit-10015 のインストール
	3.2.1.3 アガランドのIF成
	3.2.1.4 MOSAICをFRのし転と
	3.2.1.3 (田内) 子妖科のクビード時の、本和通貨の移動力広 $\dots \dots \dots$
	3.2.2 Infinite Cataputit(v.2) of V (V=V) of V (V=
	3.2.2.1 Step 1
	3.2.2.2 Step 3 152
	3 2 2 4 Step 4 154
	3.2.2.5 Step 5
	3.2.2.6 Step 6
	3.2.2.7 Step.7
	3.2.3 [アーカイブ] mijin アカウント作成 (>=1.0.0.0)
	3.2.3.1 symbol-cliのインストール
	3.2.3.2 アカウントの作成 157
	3.2.3.3 Mosaic を作成し転送 159
	3.2.3.4 (番外)手数料ありモード時の、基軸通貨の移動方法
3.3	トラブルシューティング
	3.3.1 mijin Catapult(v.2) のノード再同期 169
	3.3.1.1 対象....................................
	3.3.1.2 Step.1
	3.3.1.3 Step.2
	3.3.1.4 Step.3
	3.3.1.5 Step.4
	3.3.1.6 Step.5
	3.3.1.7 Step.6
	3.3.1.8 Step.7
	3.3.2 mijin Catapult(v.2) のノードログを確認する 178
	3.3.2.1 対象
	3.3.2.2 Step.1
	3.3.2.3 Step.2
	3.3.2.4 Step.3
	3.3.2.5 Step.4
	3.3.2.6 Step.5
	3.3.3 ノート間の暗号化通信の更新
	3.3.3.1 mijin Catapult(V.2) のノード間の暗号化通信について
2 4	3.3.3.2 ノートの SSL 証明音の史新方法
3.4	- mijin Cataputt(V.2) アーダテイレクトリ 悟道
	- 3.4.1 テーメ配直のティレクトリ

	3.4.2 mijin パッケージの構造 1	185
	3.4.2.1 APIノード	185
	3.4.2.2 PEERノード 1	187
	3.4.3 ブロックデータの構造1	190
	3.4.4 Mongo データ構造	190
3.5	mijin Catapult(v.2) 環境構築オプション表 1	192

About

1.1 mijin Catapult(v.2) について

1.1.1 mijin Catapult(v.2) とは?

mijin Catapult(v.2) は最小限のリソースで実行可能なプライベートブロックチェーンを構築することがで きます。

既存のデータベースを置き換えるプライベートブロックチェーン製品として、複数の暗号署名機能を活用 し、多くのプロジェクトで利用されています。

取引の原子性を損なうことなく、mijin のアカウントエンジンは、高いセキュリティを前提とした完全分 散型、ゼロダウンタイムのネットワークとして稼働し、アプリケーションの開発やメンテナンスの必要性 を最小化することができます。

簡単な定義であらゆるデータや資産をトークン化し、すべての取引を安全なワンタイムのスマートコント ラクトとして処理し、1秒間に数千件の取引を行う速度を実現します。

プライベートブロックチェーン"mijin"とは

クラウド上や自社データセンター内に、企業内で利用可能な プライベートブロックチェーン環境を手軽に構築できるプラットフォーム



ンとデータ自体を分散管理。ゼロダウ ンタイムの実現が可能です。

なマルチシグ(多重署名)をネイティ ブサポート。人に限らず、モノ・コトを トリガーとしても活用いただけます。

造上で定義することができます。一つ のブロックチェーン上で整合性の保持

が可能です。

mijin Catapult(v.2) で新たに提供されるのは、**アグリゲートトランザクション**(1,000 トランザクション を原子性を持って同時に実現)や **マルチレイヤーシグネチャ**(3 段階のマルチシグ認証を実現)による原 子性が保証された完全分散型のアトミックスワップ技術です。

それら先進的な機能が最低限の知識と工数で実装可能となったのは、すべて V1 における実用ケースで5年以上の年月をかけてお客様から得てきたフィードバックの賜物です。

それに NEM という長年の稼働実績があるパブリックブロックチェーンでの経験と技術が合わさったもの が、今回 AWS マーケットプレイスにて初めて日本発の商用ブロックチェーン製品として提供開始される mijin Catapult(v.2) です。

以下のサイトでも mijin Catapult(v.2) の説明を詳しく紹介しています。

mijin Catapult(v.2) 説明ページ https://mijin.io/product/

1.1.2 ユースケース

mijin Catapult(v.2) はさまざまな分野で有効活用可能です。

注釈:

詳しく知りたい方は mijin Site のページの多様な適用分野の項目、又は問い合わせより実例資料を入手することができます。



1.1.3 パブリックブロックチェーン Symbol との違い

mijin Catapult(v.2) はパブリックブロックチェーンとしてローンチされている Symbol と同じコアエンジン (Catapult) を使用しているため、Symbol でできることは mijin Catapult(v.2) でも利用可能です。 ここでは主に Symbol との違いを示します。

項目	mijin Catapult(v.2)	Symbol
チェーン	プライベートブロックチェーン	パブリックブロックチェーン
ネットワーク名	MIJIN, MIJIN_TEST	MAIN_NET, TEST_NET
ジェネシスブロックエ ポック	1560294000s (Tue 11 Jun 2019 23:00:00 UTC)	1615853185s (Tue 16 Mar 2021 00:06:25 AM UTC)
基軸通貨名	cat.currency	symbol.xym
基軸通貨 ID	mijin Catapult(v.2) 構築時に ID 作成	6BED913FA20223F8
基軸通貨発行量	8,998,999,998.000000	8,164,233,299.724038
ハーベスト通貨名	cat.harvest	symbol.xym(基軸通貨と兼用)
ハーベスト通貨 ID	mijin Catapult(v.2) 構築時に ID 作成	6BED913FA20223F8
ハーベスティング通貨 発行量	15,000,000	8,164,233,299.724038 (symbol.xym)
ブロック生成間隔	10~60 秒(構築時にカスタム可能)	30 秒
1 ブロックあたりの最 大トランザクション数	6,000 10,000 20,000 50,000 100,000	5,000
トランザクション手数 料	あり なし(構築時に指定可能)	あり
Mosaic レンタル手数 料	あり(50cat.currency) なし(構築時に指定可能)	あり(50symbol.xym)
Namespace レンタル 手数料	あり(期間によって変動) なし(構築時に指定可能)	あり(期間によって変動)
VotingKey ファイルの 期限	約 547~3285 日(ブロック生成間隔により異な る)	約 180 日
1 アカウントあたりの 最大署名数	25、50、100、1000(選択可能)	25
ファイナライズ対応	決定的、確率的(構築時に選択可能)	決定的

表 1: mijin と symbol の比較

1.2 主要データベース製品との比較と用途

1.2.1 主要データベース製品との比較

項目	mijin v.1	mijin Catapult (v.2)	MySQL	Redis	Mongo	Neo4j	Apache Hbase
種類	ブロックチェ ーン	ブロックチェ ーン	RDBMS	NoSQL(KV 型)	NoSQL(ドキ ュメント型)	NoSQL(グラ フ型)	NoSQL(ワイ ドカラム型)
概要							
	ブロック単位 でデータを格 納し、 チェーンのよ うに連結する DB	同左	表形式で整 合性を保ち、 クエリで容 易に検索可 能	キーバリュ ー構造で 高速アクセ スが可能	JSON 形式 などをスキ ーマレスで 保存・運用 できる DB	関係性をグ ラフ構造で 表現し、 高速な結合 が可能	列ごとに異 なる構造を 持ち、 高速な集計 が可能
ライセン ス	商用	商用	GPL / 商用	BSD / 商用	SSPL / 商用	GPL / AGPL / 商用	Apache 2.0
リリース 日	2015年9月	2019年6月	1995年	2009年	2009年	2007年	2010年
開発言語	Java	C++	C++	С	C++	Java	Java
対応 OS	Linux, Windows	Linux, Windows (要 Docker)	Linux, Windows	Linux, Windows	Linux, Windows	Linux, Windows	Linux
クライアント	REST API, nem-sdk	REST API, symbol-sdk (Java, TS)	mysql- client, 各言語ライ ブラリ	redis-client, 各言語ライ ブラリ	mongo- client, 各言語ライ ブラリ	REST API, WEB UI 各言語ライ ブラリ	HBase client, RPC 対応ラ イブラリ
利点							
	改ざん耐性が 高く、 勘定系アセッ ト構築が容易	上記同様	クエリが強 力、整合性 保証 情報量が多 い	シンプル API、 メモリ常駐 で高速	スキーマレ ス、 水平スケー ラブル	複雑構造で も高速検索、 クエリ対応	列志向で集 計が高速、 ビッグデー タに強い
主な用途	ポイント管理、 監査ログ	ポイント管理、 P2P 取引	顧客管理	セ ッ シ ョ ン/メッセー ジ中継	ログ、ゲー ム、アンケー ト	レコメンド、 関係性分析	分析、統計、 集計用途

表 2: 各種データベースの比較

1.3 アーキテクチャとスペック要件

1.3.1 mijin Catapult(v.2) の構成について

mijin Catapult (v.2) はノードと呼ばれるサーバ郡を構成し、プライベートブロックチェーンネットワーク を構成します。

mijin では、2 台の API ノード、3 台の PEER ノードの構成を最小の推奨構成としています。



1.3.2 PEER ノードの役割とスペック要件

PEER ノードは、ブロックチェーンデータを生成、ブロックのコンセンサス機能を提供します。 PEER ノードだけではクライアントからはアクセスできず、API ノードが必要になります。

CPU	3.1 GHz 程度のプロセッサ (CPU コア 2 以上)
メモリ	4GB RAM 以上
ディスク	
	root 30GB 以上
	block 500GB 以上 3000IOPS 以上推奨
OS	Docker が稼働する Linux(推奨 Ubuntu 20.04 以上)

注釈:

AWS MarketPlace の mijin Catapult(v.2) では、推奨以上の使用可能なスペックのみを選択することができます。

ディスク容量は、ブロックチェーンへの保存量に比例して増加します。

1.3.3 API ノードの役割とスペック要件

API ノードは、ブロックチェーンデータを mongodb に書き込み、高速で読み込める API を提供します。 API ノード上にも、ブロックチェーンデータはありますが、ブロック生成機能はなく、単純なバックアッ プとして扱われます。

このブロックチェーンデータの生成機能を有効にすることもでき、PEER ノードの機能を有した DUAL モードとして機能することができます。

CPU	3.1 GHz 程度のプロセッサ (CPU コア 2 以上)
メモリ	8GB RAM 以上
ディスク	root 30GB 以上 mongo 300GB 3000IOPS 以上推奨 block 500GB 3000IOPS 以上推奨
OS	Docker が稼働する Linux(推奨 Ubuntu 20.04 以上)

注釈:

AWS MarketPlace の mijin Catapult(v.2) では、推奨以上の使用可能なスペックのみを選択することができます。

ディスク容量は、ブロックチェーンへの保存量に比例して増加します。

Deploy

2.1 mijin Catapult(v.2) のデプロイ方法

2.1.1 mijin のデプロイ方法について

mijin Catapult(v.2)を構築する方法は、現在二通りとなります。

- 1. AWS MarketPlace を使用したデプロイ
- 2. テックビューロによる構築

2.1.1.1 AWS MarketPlace を使用したデプロイ

クラウドベンダー最大手の一つである Amazon Web Service(AWS) のマーケットプレイスにて、mijin Catapult(v.2) を容易にデプロイすることができる製品を提供しています。

AWS MarketPlace https://aws.amazon.com/marketplace

詳しくは、AWS MarketPlace を使ったデプロイ準備 を参照してください。

2.1.1.2 テックビューロによる構築

テックビューロにて、mijin 構築を行います。 詳しくは、mijin Site の問い合わせページよりお問い合わせください。

2.2 AWS MarketPlace

AWS MarketPlace は、サードパーティーのソフトウェア、サービス、およびデータの調達、プロビジョン する AWS のサービスとなります。 この章では、AWS MarketPlace に展開する mijin Catapult(v.2) の説明をしています。

2.2.1 AWS Marketplace を使ったデプロイ準備

AWS Marketplace にて提供している mijin Catapult(v.2) を使用するには、Amazon Web Services(AWS) の アカウントおよび操作の知識が必要になります。 本章では、デプロイまで事前に準備が必要となる内容を理解することができます。

2.2.1.1 AWS アカウントの準備

Amazon Web Services(AWS) はクラウド上にコンピューティング、ストレージ、データベースなどのイン フラストラクチャテクノロジーを作成するプラットフォームになります。 操作を行うには、AWS アカウント作成が必要になるため、以下のサイトを参照し、AWS アカウントを作 成してください。

https://aws.amazon.com/jp/register-flow/

警告:

mijin Catapult(v.2) をデプロイする際は ルートユーザーではなく管理用 IAM アカウントを作成しデ プロイすることを推奨します。 管理 IAM ユーザーの作成は以下を参照してください。 https: //docs.aws.amazon.com/ja_jp/IAM/latest/UserGuide/getting-started_create-admin-group.html ルートユーザーに関して知りたい方は、以下を参照してください。 https://docs.aws.amazon.com/ja_jp/IAM/latest/UserGuide/id_root-user.html

2.2.1.2 AWS の知識

mijin が使用する AWS サービスは主に以下となります。 Marketplace で提供している mijin Catapult(v.2) はデプロイ時に自動で以下を作成されます。

Amazon VPC

仮想ネットワークを構築します。mijin Catapult(v.2) はこの VPC 上のネットワークに配置され ます。 詳しくは以下のドキュメントを確認してください。 https://docs.aws.amazon.com/ja_jp/vpc/latest/userguide/what-is-amazon-vpc.html

Amazon EC2

コンピューティングマシンを作成します。mijin Catapult(v.2) のノード (サーバ) を作成します。 詳しくは以下のドキュメントを確認してください。 https://docs.aws.amazon.com/ja_jp/AWSEC2/latest/UserGuide/concepts.html

Amazon EBS

コンピューティングマシンにストレージを作成します。

https:

mijin Catapult(v.2) では、ブロックチェーンデータおよび mongo データを EBS 上に配置しま す。

mijin Catapult(v.2) では API ノードへのアクセスをロードバランサ経由にすることで冗長性を

詳しくは以下のドキュメントを確認してください。

https://docs.aws.amazon.com/ja_jp/AWSEC2/latest/UserGuide/AmazonEBS.html

コンピューティングマシンに中継する負荷分散装置を作成します。

指定し、 きめ細かいアクセス許可を一元管理します。 mijin Catapult(v.2) デプロイ時、AWS アカウントは IAM 権限が必要となります。(AWS にデプ

//docs.aws.amazon.com/ja_jp/AWSEC2/latest/UserGuide/iam-roles-for-amazon-ec2.html

パラメータストアに保存することで、万が一障害が発生した場合でも復元することが可能です。

・EC2 インスタンスから Secrets Manager への権限 ・EC2 インスタンスから SSM ログイン権限

を API,PEER ノードである EC2 インスタンスのプロファイルに割り当てます。

AWS Systems Manager Parameter Store

EC2のIAM ロールに関しては以下を参照してください。

SSM ログインについて詳しくは以下を参照してください。

session-manager-getting-started-instance-profile.html

パラメータストアに関しては、以下を参照してください。

systems-manager-parameter-store.html

mijin Catapult(v.2) デプロイ後に作成されたリソースに、

ロイするためのアカウント権限にて説明)

https://docs.aws.amazon.com/ja_jp/systems-manager/latest/userguide/

設定データ管理と機密管理のための安全な階層型ストレージとして利用します。

mijin Catapult(v.2) では、各ノードの設定情報や共通設定を保存します。

https://docs.aws.amazon.com/ja_jp/systems-manager/latest/userguide/

AWS の各サービス間など、AWS のサービスやリソースにアクセスできるユーザーやグループを

mijin Catapult(v.2) では、各ノード間を DNS で名前解決して接続しています。 詳しくは以下のドキュメントを確認してください。 https://docs.aws.amazon.com/ja jp/Route53/latest/DeveloperGuide/Welcome.html

コンピューティングマシンの名前解決に使用する DNS サービスを作成します。

Elastic Load Balancing

確保します。

Amazon Route 53

AWS IAM

what-is-load-balancing.html

https://docs.aws.amazon.com/ja jp/elasticloadbalancing/latest/userguide/

詳しくは以下のドキュメントを確認してください。

AWS CloudFormation

AWS サービスを自動で構築するオーケストレーションを提供します。 AWS Marketplace は AWS CloudFormation を使用して、AWS に mijin Catapult(v.2) を構築し ます。 詳しくは以下のドキュメントを確認してください。 https://docs.aws.amazon.com/ja_jp/AWSCloudFormation/latest/UserGuide/Welcome.html

2.2.1.3 AWS にデプロイするためのアカウント権限

AWS に mijin Catapult(v.2) をデプロイするためには、AWS アカウントの準備 で準備したアカウントに権 限を付与する必要があります。 権限としてはAWS の知識 の AWS リソースを作成する権限及び Marketplace で展開する mijin Catapult(v.2) のイメージを使用するための Subscribe する権限が必要です。

AWS Marketplace 権限およびデプロイ権限のみに絞った IAM ポリシーは、以下となります。 以下を参考に、IAM ポリシーを作成し、デプロイに使用する IAM アカウントに付与してください。 https://docs.aws.amazon.com/ja_jp/apigateway/latest/developerguide/ api-gateway-create-and-attach-iam-policy.html

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": [
                         "lambda.amazonaws.com",
                         "ec2.amazonaws.com"
                    1
                }
            }
        },
        {
            "Sid": "VisualEditor1",
            "Effect": "Allow",
            "Action": [
                "iam:*RolePolicy",
                "route53:*HostedZone",
                "iam:List*",
                "aws-marketplace:*",
                "elasticloadbalancing:RegisterTargets",
                "ec2:*RouteTable*",
                "iam:CreateRole".
                "elasticloadbalancing:DeleteLoadBalancer",
                "ec2:DescribeInternetGateways",
                "elasticloadbalancing:DescribeLoadBalancers",
                "ec2:*KeyPairs",
```

(continues on next page)

(continued from previous page)



2.2.1.4 AWS における mijin Catapult(v.2) ライセンス

AWS Marketplace の mijin Catapult(v.2) 製品版のライセンスは、Marketplace にある mijin Catapult(v.2) の EULA に同意することで使用可能になります。 EULA に関しては、以下を参照してください。 https://d7umqicpi7263.cloudfront.net/eula/product/d6b2653b-ee61-4a62-8fef-a9fa7930892e/ c255cb3f-6c72-412a-a899-42fa3f83fd71.pdf

AWS Marketplace の mijin Catapult(v.2) 製品版のライセンス費は、AWS 利用料金に合算され、**時間単位** で従量課金されます。

スペックや CPU コア数は関係なく、1 ノード単位です。

項目値mijin ライセンス費\$0.40/1 時間

注釈:

AWS MarketPlace Trial Version は mijin Catapult(v.2) ライセンス費用は発生しません。 AWS MarketPlace Enterprise x86_64 Version のノード起動は最低 5 台のため、1 ヶ月にかかるおよそのラ イセンス費の計算は以下となります。

■ 1時間単位
 \$0.4/1h * 5(台) = \$2.0/1h
 ■ 1日単位
 \$2.0/1h * 24(h) = \$48.0/1d
 ■ 1ヶ月 (30 日) 単位
 \$48/1d * 30(h) = \$1,440.0/1m

2.2.1.5 AWS 利用料金

mijin のライセンス費の他、AWS リソース使用料として、

- Amazon EC2
- Amazon EBS
- Elastic Load Balancing
- Amazon Route53
- Amazon VPC(Nat Gateway)
- ・パラメータストア

の従量課金の費用が発生します。 また、データ転送量などでも、料金が変動します。 AWS 利用料金に関しては、以下を参照してください。

https://aws.amazon.com/jp/pricing/

注釈:

AWS Marketplace で提供している mijin Catapult(v.2) は指定したパラメーターによって構成が変わり、費 用が変わります。

主にパラメーター指定によって、以下が変動することに留意してください。

- ・インスタンスタイプ
- ・ノード数 (インスタンス数)
- ・Elastic Load Balancing の必要可否
- ・VPC の作成有無
- ・EBS のブロックサイズ・IOPS

AWS Marketplace の製品説明

現在、テックビューロから提供している製品は以下となります。 各製品ごとに、AWS Marketplace 上に製品ページがあります。

- トライアル版 (AWS MarketPlace Trial Version)
- 製品版 x86_64 版 (AWS MarketPlace Enterprise x86_64 Version)
- 製品版 arm64 版 (AWS MarketPlace Enterprise arm64 Version)

AWS Marketplace で提供している mijin Catapult(v.2) は、構築に複雑な操作をする必要がありません。 デプロイ時に選んだパラメータによって、自動でネットワークを含んだ環境を構築し、mijin を含んだイ メージから安全かつ堅牢なブロックチェーンネットワークを構築します。

環境の構築される内容については、各デプロイ方法のページにて説明しています。

2.2.1.6 トライアル版

トライアル版は、簡易に起動できる mijin Catapult(v.2) として、ライセンス無償でお試しいただくことが 可能です。

冗長性はなく、単一稼働の mijin Catapult(v.2) になります。

製品版に比べ、カスタム性がなく、シングル AZ の配置ですが、試しに使ってみる環境や開発環境に適し ています。



表 1: 無償トライアル環境概要

項目	説明
マーケットプレイス製品ペー ジ	AWS MarketPlace Trial Version
デプロイ方法	
	新規の VPC に mijin Catapult(v.2) を構築
	トライアル版の mijin をテノロイする を参照
デプロイにかかる時間	約15分
サポート	
	基本サポートなし
	 AWS に関する問題は以下で問い合わせ:
	https://aws.amazon.com/jp/premiumsupport/
使用可能リージョン	
	│ │ 世界 21 リージョンで使用可能
	ap-northeast-1, us-west-1, us-west-2, us-east-1,
	us-east-2, eu-north-1, eu-west-1, eu-west-2,
	eu-west-3, eu-south-1, af-south-1, ap-south-1,
	ap-east-1, ap-northeast-2, ap-northeast-3,
	ap-sourceast-1, ap-sourceast-2, sa-east-1,
ノード数	1台(DUAL モード)
配置アベイラビリティゾーン	シングル AZ(1 つ)
ロードバランサー	なし

2.2.1.7 製品版

製品版は、エンタープライズ向けに本番運用を想定したカスタム性がある構成を自由に設定することがで き、耐障害性、高可用性のある構成を標準とし、且つ API へのアクセス方法もセキュアなアクセスを容易 に構築することが可能です。

また、mijin Catapult(v.2)を既存・新規環境を問わず設置することができ、ロードバランサを使った負荷 分散など様々な環境に適用することができます。



項目	説明
マーケットプレイス製品ペー ジ	x86_64版:AWS MarketPlace Enterprise x86_64 Version arm64版:AWS MarketPlace Enterprise arm64 Version
既存環境へのデプロイ方法	すでに既存サービスがあり、 同じ VPC 内で mijin Catapult(v.2) を管理したい場合は 既存 VPC 上に、mijin をデプロイする を参照
新規環境へのデプロイ方法	mijin Catapult(v.2) を新規に構築する、 初めて AWS を使う場合は 新規 VPC を作成し、mijin をデプロイする を参照
デプロイにかかる時間	約 30 分
サホート	デプロイ関連のお問い合わせは https://mijin.io/aws_contact/ ※ mijin Catapult(v.2) に関する技術的な質問は 有償サポート(チケット)で対応 AWS 関連は AWS サポートケースへ: https://aws.amazon.com/jp/premiumsupport/
使用可能リージョン	世界 21 リージョンで使用可能 ap-northeast-1, us-west-1, us-west-2, us-east-1, us-east-2, eu-north-1, eu-west-1, eu-west-2, eu-west-3, eu-south-1, af-south-1, ap-south-1, ap-east-1, ap-northeast-2, ap-northeast-3, ap-southeast-1, ap-southeast-2, sa-east-1, ca-central-1, eu-central-1, me-south-1
ノード数	API ノード ×2、PEER ノード ×3~7(合計 5~9 台)
配置 AZ (アベイラビリティゾ ーン)	マルチ AZ(2 つ) ※ シングル AZ は設定不可
ロードハラシッー	心女に心して政定り形

表 2: AWS デプロイ情報

2.2.1.8 有償サポートについて

mijin Catapult(v.2) のデプロイに関するサポートは無償で行いますが、mijin Catapult(v.2) に関する技術 問合せが必要な場合、有償のサポート (チケット制) を購入することでテックビューロ社より支援すること が可能です。

以下より、サポート購入の旨をお問合せください。

https://mijin.io/aws_contact/

サポート内容	
	mijin Catapult(v.2) に関する技術問合せ ノードの障害サポート (営業時間内) バージョンアップの通知・手順公開 AWS 環境におけるインフラ支援 など

2.2.1.9 AWS のサービスクォータによる制限について

AWS Marketplace mijin Catapult(v.2) は、以下の AWS サービスを使用するため、すでに環境が構築され ている AWS アカウントを利用する場合、サービスクォータによる、起動失敗する可能性があります。 サービスクォータに関しては、以下に説明があります。

https://docs.aws.amazon.com/ja_jp/general/latest/gr/aws_service_limits.html

以下に、mijin Catapult(v.2)のサービス作成数を説明します。

AWS サービス	制限に関連する内容
Amazon VPC	
	VPC: 1
	サブネット: 4(トライアルは 1)
	インターネットゲートウェイ: 1
	NAT Gateway: 2(トライアルは 0)
	ルートテーブル: 1
	セキュリティグループ:5
	参考: https://docs.aws.amazon.com/ja_jp/vpc/latest/userguide/amazon-vpc-limits.html
Amazon EC2	
	FC2 インスタンス: 5~9(トライアルは 1)
	インスタンスタイプによりクォータ変動あり
	EC2 制限: https://docs.aws.amazon.com/ja_jp/general/latest/gr/ec2-service.html
	オンデマンド制限: https://docs.aws.amazon.com/ja_jp/AWSEC2/latest/UserGuide/
	ec2-on-demand-instances.ntml#ec2-on-demand-instances-limits
	FRS- 630GR~
	https://docs.aws.amazon.com/ia_ip/general/latest/gr/ebs-service.html
	······································
Elastic Load Balancing	
	Network Load Balancer: 1(有効時)
	https://docs.aws.amazon.com/ja_jp/elasticloadbalancing/latest/network/
	load-balancer-limits.html
Amazon Route53	
	Internal ゾーン: 1
	https://docs.aws.amazon.com/ja_jp/Route53/latest/DeveloperGuide/DNSLimitations.html
AWSIAM	
	IAM ロール: 2(トライアルは 1)
	IAM ポリシー: 2(トライアルは 1)
	https://docs.aws.amazon.com/ja_jp/IAM/latest/UserGuide/reference_iam-quotas.html
Systems Manager Darameter	
Store	
	パラメーター数: 48~(トライアルは 15)
	https://docs.aws.amazon.com/ja_jp/general/latest/gr/ssm.html

表 3: AWS サービスと制限

2.2.2 新規 VPC を作成し、mijin をデプロイする

本章では、新規ネットワーク (VPC) に mijin Catapult(v.2) をデプロイする方法を示します。 製品版では、パラメーターを変更することで環境にあったネットワークを柔軟に構築することが可能です。

2.2.2.1 デプロイによって AWS 上に構築するサービス一覧

- Amazon EC2 (API ノード x 2 PEER ノード x 3~)
- Amazon EBS
- Elastic Load Balancing
- Amazon Route53
- Amazon VPC(Nat Gateway)
- ・パラメータストア

注釈:

Elastic Load Balancing は、デプロイ時に指定するパラメーターの設定により、作成しないケースがあります。

2.2.2.2 View Network

製品版では、パラメーターを変更することで環境にあったネットワークを構築することが可能です。以下 にパターン例を示します。

No	ロードバランサー	APIノード配置	PEER ノード配置	図
1	あり/公開 NW UseLoadBalancer: Yes LoadBalancerType: external	公開 NW ApiPlacementNetwork: Public	非公開 NW	
2	あり / 非公開 NW UseLoadBalancer: Yes LoadBalancerType: internal	非公開 NW ApiPlacementNetwork: ^{Private}	非公開 NW	
3	なし UseLoadBalancer: № LoadBalancerType は設定無 効	公開 NW ApiPlacementNetwork: Public	非公開 NW	
4	なし UseLoadBalancer: № LoadBalancerType は設定無 効	非公開 NW ApiPlacementNetwork: Private	非公開 NW	

表 4: ロードバランサーとノード配置パターン

注釈:

製品版では、高可用性を実現するため、シングル AZ で構成はできず、マルチ AZ のみの構成となっています。

必ずパブリックネットワーク2つ、プライベートネットワーク2つが必要になることに留意してください。 配置はシングルリージョンとなりますが、世界 20 リージョン毎にデプロイすることが可能です。

2.2.2.3 Step.1



mijin Catapult Enterprise の AMI を使用するためにサブスクライブする必要があります。赤枠のボタンを 押してください。

2.2.2.4 Step.2

miiin Catapult (v.2) Enterprise	Continu
	You must fi

< Product Detail Subscribe

Subscribe to this software

To create a subscription, review the pricing information and accept the terms for this software.

Terms and Conditions

Tech Bureau Holdings Corp. Offer

By subscribing to this software, you agree to the pricing terms and the seller's <u>End User</u> <u>License Agreement (EULA)</u> C ² . You also agree and acknowledge that AWS may share information about this transaction (including your payment terms) with the respective seller, reseller or underlying provider, as applicable, in accordance with the <u>AWS Privacy</u> <u>Notice</u> C ² . Your use of AWS services is subject to the <u>AWS Customer Agreement</u> C ³ or other agreement with AWS governing your use of such services.					
The following table shows pricing information for the listed software components. You're charged separately for your use of each component.					
mijin Catapult (v.2) Enterprise Additional taxes or fees may apply.					
mijin Catapult (v.2) Enterprise EC2 Instance Type Software/hr					

mijin Catapult Enterprise AMI を使用するため、使用の承認をしてください。

2.2.2.5 Step.3

mijin Ca	tapult (v.2) Ente	erprise		Continue to Configuration	
< Product Detail <u>Subscribe</u>					
Subscribe to this s	oftware				
You're subscribed to this software. Please see the terms and pricing details below or click the button above to configure your software.					
Terms and Conditions					
Tech Bureau Holdings Corp. O	ffer				
You have subscribed to this softwa and the seller's End User License A transaction (including your payme applicable, in accordance with the Customer Agreement ^C or other a	re and agreed that your greement (EULA) ^[2] . Yo nt terms) with the respe AWS Privacy Notice ^[2] . greement with AWS gov	use of this software is s u agreed that AWS may ctive seller, reseller or ur Your use of AWS services erning your use of such	ubject to the pricing terms share information about this iderlying provider, as remains subject to the AWS services.		
Product	Effective date	Expiration date	Action		

日: Config から起動するために、赤枠の文字をクリックしてください。

2.2.2.6 Step.4

mijin Catapult (v.2) Enterprise	(4) Continue to Launch
< Product Detail Subscribe <u>Configure</u>	
Configure this software	Pricing information
Choose a fulfillment option below to select how you wish to deploy the software, then enter the information required to configure the deployment.	This is an estimate of typical software and infrastructure costs based on your configuration. Your actual charges for each statement period may differ from
Delivery Method mijin Catapult Enterprise on New VPC CFT Software Version 1.0.0.0 (May 27, 2021)	this estimate. Software Pricing mijin Catapult \$0.40/hr (v.2) Enterprise running on t3.large
whats in this version mijin Catapult (v.2) Enterprise running on t3.large Learn more	
Region	
US East (N. Virginia) 🗸	
Use of Local Zones or WaveLength infrastructure deployment may alter your final pricing.	
Product code: cpkwiq119jldq4fuzr857563y Release notes (updated May 27, 2021)	

① の赤枠にて使用するテンプレートを選択します。ここでは新規ネットワークを作成するため「mijin Catapult Enterprise on New VPC CFT」を指定します。

② の赤枠ににて mijin のバージョンを指定します。

③ mijin を展開するリージョンを指定します。

④ の赤枠の「Continue to Launch」を押します。

2.2.2.7 Step.5

mijin Cata	apult (v.2) Enterprise			
< Product Detail Subscribe Configure	Launch			
Launch this softwar	re			
Review your configuration and choos	e how you wish to launch the software.			
Configuration Details				
Fulfillment Option	mijin Catapult Enterprise on New VPC CFT mijin Catapult (v.2) Enterprise			
Software Version	1.0.0.0			
Region	US East (N. Virginia)			
Usage Instructions				
Choose Action Launch CloudFormation	Choose this action to launch your configuration through the AWS CloudFormation console.			
	(2) Launch			

日: ① の赤枠にて、mijin の構築するサービスを指定します。ここでは「Launch CloudFormation」を指定 します。日: ② の赤枠の、「Launch」を押します

2.2.2.8 Step.6

Specify template	Create stack
Step 2 Specify stack details	Prerequisite - Prepare template
itep 3	Prepare template Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want to include in the stack.
	Template is ready Use a sample template Create template in Designer
itep 4 Review	
	Specify template A template is a JSON or YAML file that describes your stack's resources and properties.
	Template source Selecting a template generates an Amazon S3 URL where it will be stored.
	Amazon S3 URL Upload a template file
	Amazon S3 URL
	https://s3-ap-northeast-1.amazonaws.com/cf-templates-1301ubacqb2jo-ap-northeast-1/2020331YN8-template1b103xcr6zch
	Amazon S3 template URL
	S3 URL: https://s3-ap-northeast-1.amazonaws.com/cf-templates-1301ubacqb2jo-ap-northeast-1/2020331YN8-template1b10

日:とくに編集せず、赤枠の「Next」を押します。

2.2.2.9 Step.7

C Search	(Option+5) 🕑 👃 🧿 🙆 Adu Pediti (Takya) +	
E CloudFormation > Stacks	> Create stack	0 0
Create stack	Specify stack details	
Step 2 Specify stack details	Provide a stack name	
Step 3	Stack name	
 Configure stack options 	Enter a stack name	
 Step 4 Review and create 	Stack name must contain only initions (i-z, A-Z), numbers (0-9), and hyphens (-) and start with a letter. Max 128 characters. Character count: 0/128.	
	Parameters	
	Parameters are defined in your template and allow you to input custom values when you create or update a stack.	
	VPC Configuration	
	ServiceName Please Input Service Name.	
	(2) (MUIN-CATAPULT	
	AvailabilityZone1	
	Select AWS:EC2:AvailabilityZone::Name	•
	AvailabilityZone2	
	Piesse input Availability/Zone2	
	Security Group Configuration	
	Plauectocationin Plauec	
	Enter String	
	Node Configuration	
	DefaultUnixUser Please Input Default Unix User	
	(b) (ubuntu	
	KeyName Name of an astrition EC2 XexPainto enable SSN arrows to the ani and name instances	
	Select AWS:EC2:KeyPair:KeyName	•
	API Node Configuration	
	ApiPlacementNetwork	
	Presse select network Public	•
	AniinstanceTyne	
	APIEC2 Instance type	
	(aguirge	<u> </u>
	ApiRootVolumeSize Boot Volume Size	
	(10) (30	_
	ApiBlockVolumeSize API Block Volume Size(60)	
	(1) 500	•
	ApiBlockVolumelops	
	API Block Volume lops 3000	•
	AniMenneVolumaSize	
	12 Ar Mongo Volume Size(Gil)	
		<u> </u>
	AplMongoVolumelops Poot Volume lops	
	(14) (3000	•
	PEER Node Configuration	
	PeerNumberOfUnits Please input Number of Peer EC2 instances Unit/Net Autoscilling)	
	(15) (3	•
	PeerInstanceType	
	16 PERR EC2 Instance type 14g.large	_
	PearBontVolumeSize	
	PEER Root Volume Size(GB)	
	U (30	•

	PéerBlock Volume Size(GB)
8	500
_	PeerBlockVolumetops
9	3000
	milin Configuration
	CatapultVersion
20	vitaas sooet mijn cataput version v10038
9	CatapultShareMode
21)	Please select mijn Catapult Share Mode
9	
20	Cataputtvetwork Please select mijin Catapult Network Identifier
<u>u</u>	mjin
_	CatapultBlockGenerationTargetTime Please select mijn Cataputt Block Generation Target Time
23)	155
	CatapultEffectiveFee Name valueT min Catanut Effective Fee
24)	No
	MaxCosignedAccount
25)	Please select Max Cosiglined Account per Account 25
-9	
20	Please select mijin Catapult Finalization Type
9	Deterministic
_	MaxTransactionperBlock Please select Max Transacion per Block
27)	6:000
	RestThrottring Disease value / Barts Ilmuit
28	30tps
	UnconfirmCacheSize
29)	Please maximum size of the unconfirmed transactions cache. Small
-	
	loadbalancer Configuration UseLoadBalancer
30	Do you use a load balancer(network load balancer)? Yes
	LosfRalancerTune
21)	If a load balancer is enabled, spedifies the load balancer type
ົງ	external Imageld
	This is the alias of the Marketplace AMI that will be deployed as part of this stack. Ensure this parameter is set to the following value: /aws/service/marketplace/prod- ustqtwosgdbtw/1.0.3.8.
32)	/aws/service/marketplace/prod-ustqtwosgsbhw/1.0.3.8
	MPS3BucketName This is the name of the Amazon 53 bucket containing the nested templates that will be deployed as part of this stack. Ensure this parameter is set to the followino value: awsmo-
22	992382380361-1708727387563.
	Landra and Andrew
20	MMP>SDUCKETKEGION This is the AWS Region of the bucket containing the nested templates that will be deployed as part of this stack. Ensure this parameter is set to the following value: us-east-1.
54)	us-east-1
	MPS3KeyPrefix This is the prefix of the nested templates in the Amazon 53 bucket that will be deployed as part of this stack. Ensure this parameter is set to the following value: Occ0725b-75b3
35)	9886-55560202020, 0ecd725b-75b3-477c-98a8-553ebe25b208/
9	mijinStackAlreadyExist
	mijin Stack already exist? If set 'No', create Macro
6	(NO

日: パラメータを入力します。

表 5:	mijin	デプロ・	ィ用パラ	メーター	一覧	(簡易版)
------	-------	------	------	------	----	-------

No	パラメータ	説明	推奨値
1	Stack Name	このスタックの名前	•
2	Service Name	全リソースの冠名として利用されるサービス名	•
3	Availability Zone1	使用するアベイラビリティゾーン(Multi-AZ 構成)	•
4	Availability Zone2	AZ1 と異なる AZ を指定(Multi-AZ 構成)	•
5	Public Location IP	API 接続許可 IP アドレス(/24 等も可)	例: XX.XX.XX.XX/32
6	Default UnixUser	EC2 の標準 Unix ユーザー	ubuntu
7	KeyName	EC2 SSH 接続用鍵名	•
8	ApiPlacementNetwork	API ノードのネットワーク配置先	Public
9	ApiInstanceType	API ノードのインスタンスタイプ	c5n.2xlarge 以上
10	ApiRootVolumeSize	API ノードのルートディスク容量(Docker 等に使用)	30GB 以上
1	ApiBlockVolumeSize	mijin ブロックデータ格納用ディスク容量	500GB 以上
12	ApiBlockVolumeIops	上記ディスクの IOPS 設定	3000 以上
13	ApiMongoVolumeSize	mongo データ格納用ディスク容量	300GB以上

次のページに続く

No	パラメータ	説明	推奨値
(14)	ApiMongoVolumelops	mongo ディスクの IOPS 設定	3000以上
(15)	PeerNumberOfUnits	PEER ノードの台数	3
(16)	PeerInstanceType	PEER ノードのインスタンスタイプ	c5n.xlarge 以上
17	PeerRootVolumeSize	PEER ノードのルートディスク容量	30GB 以上
(18)	PeerBlockVolumeSize	PEER ノードのブロックデータ用ディスク	500GB 以上
(19)	PeerBlockVolumeIops	上記ディスクの IOPS 設定	3000以上
20	CatapultVersion	mijin のバージョン	v10038
21)	CatapultShareMode	設定保存方式(例:SSM)	ssm
22	CatapultNetwork	ネットワーク名	mijin
23	BlockGenerationTargetTime	ブロック生成間隔	15s
24)	EffectiveFee	手数料設定(あり/なし)	No
25	MaxCosignedAccount	署名可能最大アカウント数	25
26	FinalizationType	ファイナライゼーション方式	Deterministic
27)	MaxTransactionperBlock	最大トランザクション数(1 ブロック)	6000
28	RestThrottling	API 接続上限	30tps
29	UnconfirmCacheSize	未承認トランザクションキャッシュサイズ	•
30	UseLoadBalancer	NLB の使用有無	Yes
31)	LoadBalancerType	ロードバランサー種別	External
32)	ImageId	管理用 AMI の ID(変更不可)	変更不可
33	MPS3BucketName	S3 バケット名(変更不可)	変更不可
34)	MPS3BucketRegion	S3 リージョン(変更不可)	変更不可
35	MPS3KeyPrefix	S3 のプレフィックス(変更不可)	変更不可
36	mijinStackAlreadyExist	既存スタックの有無	No

表 5 – 前のページからの続き

日: パラメータの入力完了後、「Next」を押します。

2.2.2.10 Step.8

		4	Tokyo 👻 Sup	port 🕶	
CloudFormation > Stacks >	Create stack				
Step 1 Specify template	Configure stack option	15			
Step 2 Specify stack details	Tags You can specify tags (key-value pairs) to apply t	o resources in your stack. You can add up to 50 unique tags for	each stack. Learn more 🔀		
Step 3 Configure stack options	Key	Value		R e	
Step 4 Review				m o v e	
	Add tag				
	Permissions Choose an ILM role to explicitly define how CloudFormation can create, modify, or delete resources in the stack. If you don't choose a role, CloudFormation uses permissions based on your user credentials. Learn more 🖄				
	IAM role - optional Choose the IAM role for CloudFormation to use IAM role na Sample-role-nat	for all operations performed on the stack. ne	▼ Remove		
	Advanced options				
	You can set additional options for your stack	, like notification options and a stack policy. Learn me	ore [2]		
	You can set additional options for your stack Stack policy Defines the resources that you want to prote	; like notification options and a stack policy. Learn me ect from unintentional updates during a stack update.	ore 🖸		
	You can set additional options for your stack • Stack policy Defines the resources that you want to prote • Rollback configuration Specify alarms for CloudFormation to monit back. Learn more C	, like notification options and a stack policy. Learn me ext from unintentional updates during a stack update.	ches an alarm threshold, CloudFormation	rolls it	
	You can set additional options for your stack • Stack policy Defines the resources that you want to prote • Rollback configuration Specify alarms for CloudFormation to monit back. Learn more 1 • Notification options	, like notification options and a stack policy. Learn me set from unintentional updates during a stack update.	ches an alarm threshold, CloudFormation	rolls it	
	You can set additional options for your stack Stack policy Defines the resources that you want to prote Rollback configuration Specify alarms for Cloudformation to monit back. Learn more ? Notification options Stack creation options	, like notification options and a stack policy. Learn mo	ches an alarm threshold, CloudFormation	a rolls it	

日:とくに編集せず、赤枠の「Next」を押します。

2.2.2.11 Step.9

	Review MUIN-CATAPULT1		
	Step 1: Specify template		881
stash details			
	Telepiza		
er task splare.	Template URL Neos Destination in the template test of an excitate to a second and a	Contractor second features in cont	
	Tork decoutor		
	mijin Calapuli Brianpina with multiple submits VPC		
	tational call not particle		
	Step 2: Specify stack details		-
	Parameters (31)		
	G, Grand parameters		*
	No	 Value 	
	Apittodiviburinings		
	Apilited Video Circ	500	
	Apirotaccelgo	tharps	
	JaPingilianelar		
	ApPlasment/InterA	Public	
	Aptenteturitee		
	Avelability.com	ap-northeast hz	
	Availability2cmi2	ap-northeast-fia	
	Case-utilite distance of angeofine	*	
	Criseditive	-	
	CategolillowHede		
	CalipulPrevion	+10000	
	Sufault Holder	abartu	
	Reductorhol	Probabilities	
	Kephene	N21-02849-57	
	Loadblamp Tge	anternal	
	Refune formilies	27300	
	PeerWook/information		
	PeerBid/Islamitize		
	Permananyei	things	
	Pagnantaciti, Ma	3	
	Papriportialamidian		
	Public surries		
	La vialiant	HUR-COMPLETE	
	Unardentabeliae	Medium	
	Uniteditioner	744	
	mpraskekeepen	10	
	Stars I: Carolinear stark cardions.		101
	and a can report many oppose		
	Tags (2)		
	(a secolo		
	Ng . 10	•	
	No Taga Trans an es bass defined fo	r this shade	
	Permissions		
	There is no bill role amendated	with this stack	
	Stark policy		
	P		
	As much policy. There is no starting address	infrad	
	Baliback configuration		
	Rollback configuration		
	Bullack configuration Restance to:		
	Kulikask coofiguration Protocomp term - - Coofficient Jan J3N		
	Natikask configuration Recenting from - Candidatin Jaco J.M. -		
	Ralitack configuration Restances one 		
	Rullack configuration Rection proc Contraction date: ARR - Notification systems		
	hullack configuration Amounts on Constrained and AM Constrained		
	Multiack configuration Minimum rise	ee ma Admad	
	Mallask configuration Marana en - Configuration dan dan - Marification aptions Non-Mallane aptions Deer are conduction apti- Deer are conduction apti-	We are allowed	
	hullack configuration Manage for Construction down Alther Construction down Alther Marification applies Such construction applies	en andrea	
	Multitack configuration Straining tom	en adat	
	Multitude configuration Intervention of the Condition date of the Condition date of the Present condition of the Filter condition of the Fil	NA NA NA NA NA NA NA NA NA NA NA NA NA N	
	Malikak configuration Marina dan -	m man addrese	
	Multiack configuration Multiack configuration Straining to:	m m	
	Multiaxis configuration Hoursenance - - - Multiaxies appliese - <	me me added	
	Matikask configuration Matikask configuration Matikask configuration - - Matikask configuration - Matikask configuration - <td>Ma Markana Andre</td> <td></td>	Ma Markana Andre	
	Multiack configuration Berning ton - Contraction data AM - Determined to AM -	Ma and a second s	
	Milliok configuration International - Configuration and Million - Model of the Million Model of the Million Match on Million Model of the Million Model of Million Model of Million	see addred	
	Mallack configuration Manual procession	Ma man address	
	Millioks configuration Werning the - Configuration data disk - Million data disk -		
	Millioxis configuration International Internationality Internatio	me me endered	total Boy Low
1	Millioxis configuration Minutes procession	Mini mana andread The property set of the profit and with first sectors are profit that the sectors are sets as a sequent the project of the	
1		nes un a chical	total day-taon 1 2 3

日: ①の赤枠にて2つの項目にチェックを入れます。

日:②の赤枠の「Create Stack」を押します。エラーがなければ作成が始まります。
2.2.2.12 Step.10

CATAPULT-PRE1 Events Resources Outputs h events	Parameters Template Change sets		
CATAPULT-PRE1	Parameters Template Change sets		
Events Resources Outputs A events	Parameters Template Change sets		
1) h events			
th events			
P		Status	Status reason
6 17:48:36 UTC+0900	MUIN-CATAPULT-PRE1	CREATE_IN_PROGRESS	User Initiated
	6 17:48:36 UTC+0900	6 17:48:36 UTC-0900 MUIN-CATAPULT-PRE1	6 17.48.36 UTC-0900 MURI-CATAPULT-PRE1 (CREATE_IN_PROGRESS

日: Stack が始まり「CREATE_IN_PROGRESS」になっていることを確認してください。この状態はおよそ 20~30 分程度かかります。

2.2.2.13 Step.11

⊡ Stacks (6) ℃	MIJIN-CATAPULT1	Delete Update Stack actions V Create stack
Q MIJIN-CATAPULT1 X	Stack info Events Resources Outputs Parameters	Template Change sets
Active View Hester	Events (17)	C
NESTED MIJIN-CATAPULT1-loadBalancerNestStack -1HCA87RNUNK9Q	Q. Search events	6
2021-04-30 13:42:04 UTC+0900 CREATE_COMPLETE	Timestamp V Logical ID	Status Status reason
NESTED	2021-04-30 13:45:40 UTC+0900 MIJIN-CATAPULT1	⊘ CREATE_COMPLETE -
12223GK	2021-04-30 13:45:37 UTC+0900 loadBalancerNestStack	⊘ CREATE_COMPLETE -
CREATE_COMPLETE	2021-04-30 13:42:05 UTC+0900 loadBalancerNestStack	CREATE_IN_PROGRESS Resource creation Initiated
NESTED MIJIN-CATAPULT1-macroNestStack-TFT74	2021-04-30 13:42:04 UTC+0900 loadBalancerNestStack	CREATE_IN_PROGRESS -
GMT7QPN 2021-04-30 13:22:32 UTC+0900	2021-04-30 13:41:59 UTC+0900 mijinNestStack	⊘ CREATE_COMPLETE -
CREATE_COMPLETE	2021-04-30 13:25:41 UTC+0900 mijinNestStack	CREATE_IN_PROGRESS Resource creation Initiated
NESTED MIJIN-CATAPULT1-vpcNestStack-GGV80H	2021-04-30 13:25:40 UTC+0900 mijinNestStack	CREATE_IN_PROGRESS -
6B6NSL 2021-04-30 13:22:32 UTC+0900	2021-04-30 13:25:36 UTC+0900 vpcNestStack	⊘ CREATE_COMPLETE -
CREATE_COMPLETE	2021-04-30 13:23:33 UTC+0900 iamNestStack	⊘ CREATE_COMPLETE -
NESTED MIJIN-CATAPULT1-iamNestStack-1IU2EK1	2021-04-30 13:23:20 UTC+0900 macroNestStack	⊘ CREATE_COMPLETE -
RO8MA 2021-04-30 13-22-32 UTC+0900	2021-04-30 13:22:33 UTC+0900 macroNestStack	CREATE_IN_PROGRESS Resource creation Initiated
CREATE_COMPLETE	2021-04-30 13:22:32 UTC+0900 vpcNestStack	CREATE_IN_PROGRESS Resource creation Initiated
MIJIN-CATAPULT1	2021-04-30 13:22:32 UTC+0900 iamNestStack	CREATE_IN_PROGRESS Resource creation Initiated
CREATE_COMPLETE	2021-04-30 13:22:31 UTC+0900 vpcNestStack	CREATE_IN_PROGRESS -
	2021-04-30 13:22:31 UTC+0900 macroNestStack	CREATE_IN_PROGRESS -
	2021-04-30 13:22:31 UTC+0900 iamNestStack	CREATE_IN_PROGRESS -
	2021-04-30 13:22:24 UTC+0900 MIJIN-CATAPULT1	CREATE_IN_PROGRESS User Initiated

日: 「CREATE_COMPLATE」の状態であれば、mijin の作成が完了しました。

2.2.2.14 Step.12

MIJIN-CATA	VPULT1 Delete Update Sta	ck actions 🔻	Create stack v
Stack info Ev	rents Resources Outputs Parameters Template Change sets		
Outputs (6)			C
Q Search output	2		٢
Key 🔺	Value 🗸	Description	▼ Export name ⊽
chainInfo	http://MIJIN-CATAPULT1-nlb-rest-0a16468e0cca959b.elb.ap-northeast- 1.amazonaws.com:3000/chain/info	-	-
emptyAddress	https://ap-northeast-1.console.aws.amazon.com/systems-manager/parameters/MIJIN- CATAPULT1/shares/nemesis_addresses.json	-	-
harvestAddress	https://ap-northeast-1.console.aws.amazon.com/systems-manager/parameters/MIJIN- CATAPULT1/shares/nemesis_addresses_harvesting.json	-	-
mijinEndpoint	http://18.183.121.6:3000,http://54.95.9.192:3000	mijin Catapult Rest Endpoint	-
mijinLBENdpoint	http://MIJIN-CATAPULT1-nlb-rest-0a16468e0cca959b.elb.ap-northeast-1.amazonaws.com:3000	-	-
nodePeers	http://MIJIN-CATAPULT1-nlb-rest-0a16468e0cca959b.elb.ap-northeast- 1.amazonaws.com:3000/node/peers	-	-

日: 作成した Stack の「Outputs」を押すと、作成された mijin の設定情報を確認できます。

2.2.2.15 mijin エンドポイントと確認項目

	mijinLBEndpoint
	ロードバランサー経由の mijin API エンドポイントです。セッション維持設 定が有効で、ソース IP に基づいたスティッキーセッションとなります。詳 細はこちら
	mijinEndpoint API ノード (EC2) の直アクセス用の mijin API エンドポイントです。
Your Might * 10	chainInfo mijin の現在のブロック数を確認できます。ブロック数が「2」以上であれ ば正常です。
	harvestAddress AWS Systems Manager パラメータストアに登録された通貨分配用アドレ スのリンクです。
MULCIANCE CARACTERIZATION CONTRACTOR CONTRAC	emptyAddress AWS Systems Manager パラメータストアに登録された未使用アドレスの リンクです。
 Yeard, S., Yang, Y. & Sama, Y. & Sa	nodePeers mijin API からノードの接続状態が確認できます。API ノード1台と、設定 済みの PEER 台数が表示されていれば OK です。

表 6: mijin エンドポイントと確認項目

日: これで mijin Catapult を使用する準備が整いました。それでは次の項で操作を始めてみましょう!

2.2.3 既存 VPC 上に、mijin をデプロイする

本章では、既存ネットワーク (VPC) に mijin Catapult(v.2) をデプロイする方法を示します。 製品版では、パラメーターを変更することで環境にあったネットワークを柔軟に構築することが可能です。

2.2.3.1 デプロイによって AWS 上に構築するサービス一覧

- Amazon EC2 (API ノード x 2 PEER ノード x 3~)
- Amazon EBS
- Elastic Load Balancing
- Amazon Route53
- パラメータストア

注釈:

Elastic Load Balancing は、デプロイ時に指定するパラメーターの設定により、作成しないケースがあります。

2.2.3.2 既存 VPC のサブネットの作成

mijin をデプロイするうえで、高可用性を実現するため、Public ネットワークと Private ネットワークのサ ブネットが二つずつ必要になります。 サブネット数が足りない場合、以下を参考に作成してください。

<https://docs.aws.amazon.com/ja_jp/vpc/latest/userguide/working-with-subnets.html# create-subnets>

注釈: 片方の AZ に障害が発生してもサービスを継続できるように、複数のアベイラビリティゾーン (AZ) のサブネット二つを作成してください。

2.2.3.3 View Network

既存ネットワーク用では、パラメーターを変更することですでに存在する環境へ構築することが可能です。 以下にパターン例を示します。

No	ロードバランサー	APIノード配置	PEER ノード配置	図
1	あり(公開ネットワーク) VPC: Your vpc VpcCidrBlock: xx.xx.xx/xx Public1/2, Private1/2 UseLoadBalancer: Yes LoadBalancerType: external	公開 NW ApiPlacementNetwork: ^{Public}	非公開 NW	
2	あり(非公開ネットワーク) UseLoadBalancer: Yes LoadBalancerType: internal	非公開 NW ApiPlacementNetwork: Private	非公開 NW	
3	なし UseLoadBalancer: № LoadBalancerType は設定効 果なし	公開 NW ApiPlacementNetwork: Public	非公開 NW	
4	なし UseLoadBalancer: № LoadBalancerType は設定効 果なし	非公開 NW ApiPlacementNetwork: Private	非公開 NW	

表 7: API/PEER ネットワーク構成一覧

注釈:

製品版では、高可用性を実現するため、シングル AZ で構成はできず、マルチ AZ のみの構成となっています。

必ずパブリックネットワーク2つ、プライベートネットワーク2つが必要になることに留意してください。 配置はシングルリージョンとなりますが、世界 21 リージョン毎にデプロイすることが可能です。

2.2.3.4 Step.1



mijin Catapult (v.2) Enterprise By: Tech Bureau Holdings Corp. C Latest Version: 1.0.0.0 "mijin allows you to build a viable private blockchain with minimal resources" Linux/Unix



日: mijin Catapult Enterprise の AMI を使用するためにサブスクライブする必要があります。赤枠のボタ ンを押してください。

2.2.3.5 Step.2



mijin Catapult (v.2) Enterprise

You must fi

< Product Detail Subscribe

Subscribe to this software

To create a subscription, review the pricing information and accept the terms for this software.

Terms and Conditions

Tech Bureau Holdings Corp. Offer



日: mijin Catapult Enterprise AMI を使用するため、使用の承認をしてください。

2.2.3.6 Step.3

mijin Ca	tapult (v.2) Ente	erprise		Continue to Configur
< Product Detail <u>Subscribe</u>				
Subscribe to this s	oftware			
You're subscribed to this software. configure your software.	Please see the terms an	nd pricing details below	or click the button above to	
Terms and Conditions				
Tech Bureau Holdings Corp. O	ffer			
You have subscribed to this softwa and the seller's End User License A transaction (including your payme applicable, in accordance with the Customer Agreement ² or other a	re and agreed that your greement (EULA) C ² . Yo nt terms) with the respe AWS Privacy Notice C ² . greement with AWS gov	use of this software is a u agreed that AWS may ctive seller, reseller or u Your use of AWS service erning your use of such	subject to the pricing terms share information about this inderlying provider, as s remains subject to the AWS services.	
Product	Effective date	Expiration date	Action	

日: Config から起動するために、赤枠の文字をクリックしてください。

2.2.3.7 Step.4

mijin Catapult (v.2) Enterprise	(4) Continue to Launch
eventue teal where the tenting were Between Version Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Software Version Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Software Version Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Construction on Existing VPC CFT Image: Con	<section-header> Pricing information This is an estimate of typical software and infrastructure costs based on your configuration. Your actual charges for each statement period may differ from this estimate. Software Pricing mijin Catapult (v.2) Enterprise running on t3large</section-header>

日: ① の赤枠にて使用するテンプレートを選択します。ここでは既存ネットワークに作成するため「mijin Catapult Enterprise on Existing VPC CFT」を指定します。日: ② の赤枠ににて mijin のバージョンを指定します。日: ③ mijin を展開するリージョンを指定します。日: ④ の赤枠の「Continue to Launch」を押します。

2.2.3.8 Step.5

mijin Cata	oult (v.2) Enterprise
< Product Detail Subscribe Configure La	aunch
Launch this software	e
Review your configuration and choose	how you wish to launch the software.
Configuration Details	
Fulfillment Option	mijin Catapult Enterprise on Existing VPC CFT mijin Catapult (v.2) Enterprise running on t3.large
Software Version	1.0.0.0
Region	US East (N. Virginia)
Usage Instructions	
Choose Action	1
Launch CloudFormation	Choose this action to launch your configuration through the AWS CloudFormation console.
	(2) Launch

日: ① の赤枠にて、mijin の構築するサービスを指定します。ここでは「Launch CloudFormation」を指定 します。日: ② の赤枠の、「Launch」を押します

2.2.3.9 Step.6

Specify template	Create stack			
Step 2 Specify stack details	Prerequisite - Prepare template			
itep 3	Prepare template Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want to include in the stack.			
	Template is ready Use a sample template Create template in Designer			
itep 4 Review				
	Specify template A template is a JSON or YAML file that describes your stack's resources and properties.			
	Template source Selecting a template generates an Amazon S3 URL where it will be stored.			
	Amazon S3 URL Upload a template file			
	Amazon S3 URL			
	https://s3-ap-northeast-1.amazonaws.com/cf-templates-1301ubacqb2jo-ap-northeast-1/2020331YN8-template1b103xcr6zch			
	Amazon S3 template URL			
	S3 URL: https://s3-ap-northeast-1.amazonaws.com/cf-templates-1301ubacqb2jo-ap-northeast-1/2020331YN8-template1b10			

日:とくに編集せず、赤枠の「Next」を押します。

2.2.3.10 Step.7

tic Container Service	(obrout.st) (ii) 🔶 🖉 🦉 waaveere (oolog a	
CloudFormation > Stacks >	Create stack	0
reate stack	Specify stack details	
tep 2	Provide a start come	
- T	Frovide a static name	
infigure stack options	Enter a stack name	
p 4 view and create	Stack name must contain only letters (a-z, A-Z), numbers (3-4), and hyphens (-) and start with a letter. Max 128 characters. Character count: 0/128.	
	Parameters	
	Parameters are demined in your template and allow you to input custom values when you create or opdate a stack.	
	ServiceName	
(Plane Input Service Nerre.	
,	Please select the Vpc id to install mijin	
(Select AWS::EC2:VPC:/d	•
	VpcCidrillock Please input an P rance in this VPC. Normally, you can specify the Cidrillock of the VPC example. 10.0.0.07/6	
(4 Enter String	
	Public1	
(Please select Subart lies of Public network Floase select Subart	
· · · · · ·		
	Public2 Please select Subnet lids of Public network. Please specify a different network than Public1.	
(6 Select AWStatC2:Submetted	•
	Private 1 Place refers Subset into all Photos reference	
(7 Select MWScECE:Subnettid	•
	Private2	
(Please select Subset lish of Private notwork. Please specify a different network than Private1.	
```		
	InternalDomainName Please input domain name,luse internal) If you have the same domain in this VPC, you will need to set different values.	
(	9 mijn.internäl	
	Security Group Configuration	
	PublicLocationIP Please input range of IP addresses that can access mijlin rest. Do not use 0.0.0.0.0/D if CataputotHectiveFee is No.	
(	10 Enter String	
	Node Configuration	
	DefaultUnixUser	
(	by a set of the contract of th	
	KevName	
	Name of an existing EC2 KeyPair to enable SSH access to the api and peer instances	
(		•
	API Node Configuration	
	ApiPlacementNetwork Please select network	
(	Public	•
	AplinstanceType	
(	14 téglarge	
	ApiRootVolumeSize	
	Rect Volume Size	~
(	U (30	•
	ApiblockVolumeSize APi Block Volume Size(GB)	
(	16 500	•
	ApiBlockVolumelops	
(	APE Block Volume tops 3000	
	ApiMongovolumesize API Mongo Volume Size(60)	
(	300	•
	ApiMongoVolumelops	
(	19) 3000 3000	
	PEER Node Configuration PerchamberOflinits	
	Plase input Number of Peer IC2 instances Unit. (Not Autoscaling)	
(	🕐 🕐	

	PeerinstanceType
)	t4g.large
	PeerRootVolumeSize
	PETR Root Volume Size(Sil)
	30
	PeerBlockVolumeSize PEIR Block Volume Size(GB)
	500
	PeerBlockVolumelops
	7000
	mijin Configuration
	Cataput/version Please select mijn Catapult Version
	v10038
	CatapultShareMode Please whet mile Catanul Share Mode
	same and a second might same to some
	CataoultNetwork
	Please select mijn Catapult Network Identifier
	( mhu
	CatapultBlockGenerationTargetTime Please select mijn Catapult Block Generation Target Time
	15s
	CatapultEffectiveFee
	Please select mijn Catapult Effective Fee
	. m.
1	MaxCosignedAccount Vieses select Max Cosigined Account per Account
	25
ļ	FinalizationType
	mease select myn cataputt i maitatatan type Deterministic
	Lee Terrer allowed New Street
	Maxiransactionpersiock Please select Max Transacion per Black
	6000
	RestThrottring
	30tps
	UnconfirmCacheSize
	Please maximum size of the unconfirmed transactions cache.
	Small
	loadbalancer Configuration
	UseLoadBalancer Do you use a load balancer(network load balancer)?
	Yes
	LoadBalancerType
	If a Load balancer is enabled, specifies the load balancer type evternal
1	imageld
	This is the alias of the Marketplace AMI that will be deployed as part of this stack. Ensure this parameter is set to the following value: /aws/service/marketplace/prod- unitgtwosgsbirw/1.0.5.8.
	/aws/service/marketplace/prod-ustqtwosgsbhw/1.0.3.8
	MPS3BucketName
	This is the name of the Amazen 53 bucket containing the nested templates that will be deployed as part of this stack. Ensure this parameter is set to the following value: ave 99236230261-1708727387563.
	awsmp-cft-992382380361-1708727387563
	MPS3BucketRegion This is the AWS bridge of the backet certaining the prested templates that will be derived as part of this stack. Ensure this parameter is not to the following online on anti-
	useest-1
	MPSTKurðnafiv
	In a supervision. This is the prefix of the nested templates in the Amazon S3 bucket that will be deployed as part of this stack. Ensure this parameter is set to the following value: a051154-5 5977-ahtea374077.d.
	a05f1f54-9233-4eca-9577-afbcaa7a0f7c/
	mijinstackaireadyexist
	mijin Stack Already exist? il set 'No', create Macro

日: パラメータを入力します。

表 8: CloudFormation /	パラメー	ター覧
-----------------------	------	-----

No	Parameter	Describe	推奨値
1	Stack Name	このスタックにおける名前を記載してください。	•
2	Service Name		•
		スタックによって作成されるサービス名を記載してください。 全リソースの冠名として使用されます。	
3	VPC	お使いの VPC を指定してください。	•
4	VpcCidrBlock		•
		③ で指定した VPC の IP アドレスの範囲を記載してください。(CIDR Block) ③ で選択した () 内の IP アドレス範囲で問題ありません。	
5	Public1	③ で指定した VPC にある公開ネットワークを指定してください。	•
6	Public2		•
		③ で指定した VPC にある公開ネットワークを指定してください。 ⑤ と別のネットワークを指定する必要があります。	

次のページに続く

No	Parameter	Describe	堆将值
<i>W</i>	Private1	③で指定したVPCにめる非公開イットワークを指定してくたさい。	•
8	Private2		
		③で指定した VPC にめる非公開ネットワークを指定してくたさい。	
		⑦とは別のネットワークを指定する必要があります。	
(9)	InternalDomainName		mijin.internal
		ノード間で使用する名前解決用の DNS名を記載してください。	
		非小問ネットワーク田で、小問けされません	
		後数スタック備来時には一息にしてくたさい。	
10	Dublic Leasting ID		XX XX XX XX /22
0	Public Location IP		XX.XX.XX.XX/32 たど
		mijin Catapult の API への接続許可 IP アドレスを指定してください。	4 C
		IPレンジ指定可(/24など)	
(11)	Dofault Univelsor	作成する EC2 の 煙淮 Unix コーザー	ubuntu
10	Keyllamo		dbulltu
	Reyname		•
		EC2 の SSH 鍵を指定してください。	
		表示されない場合は事前作成が必要です。	
		655	
(i)	A mi Dia anno annt Niata una mir		
(13)	ApiPlacementNetwork	APIノートの配直ネットワークを迭折してくたさい。	•
(14)	ApilnstanceType		c5n.2xlarge 以上
			0
		APIノードのインスタンスタイプ	
		アーキテクチャとスペック要件 参照	
(15)	ApiRootVolumeSize		30GB 以上
		Docker ロクや System log に使用	
(16)	ApiBlockVolumeSize		500GB 以上
		mijin ブロックデータ格納田ディスクサイズ (GB)	
		gho ノ 1 × 2 径 利用	
17	AniBlack/olymolong		2000 IV/ F
<u></u>	ApiBlockVolumelops	mijin フロックテータ用 iOPS	3000 以上
(18)	ApiMongoVolumeSize		300GB 以上
		mongo データ格納田ディスクサイブ (GB)	
		Riorigo ア シ油樹所 イスフライス (3D) Blockchain デーク 取び出し 田	
		DIOCKCITAIIIテータ呼び出し用	
			200010.000 101 1
(19)	Apimongovolumelops	mongo テータ用 IOPS	3000IOPS 以上
(20)	PeerNumberOfUnits	PEERノードの台数	3以上
21)	PeerInstanceType		c5n.xlarge 以上
		FLLIN F FUT ノスダノスダイノ マーナニカチャレフィック西佐 名四	
		アニキテンテヤとスペツン安計 梦照	
			2005 111
(22)	PeerRootVolumeSize		30GB 以上
		PEER ノードのルートディスクサイズ (CR)	
		Dockor ログや System log に住田	
		DUCKEI 山ノア System lug に反用	
(1)	PeerPleels/shares Cint		EQCODIN 1
<u>(1</u> )	PeerBlockVolumeSize		500GB 以上
(24)	PeerBlockVolumelops	PEER ノードのフロックテータ用 IOPS	3000IOPS 以上
25	CatapultVersion	mijin のバージョン	v10038
26	CatapultShareMode		ssm
-			
		フロック生成情報保存先を指定	
		現在は AWS Systems Manager に固定	
27)	CatapultNetwork	mijin ネットワーク指定	mijin
(28)	Catapult	ブロック生成間隔	15s
	BlockGenerationTargetTime		
29	CatapultEffectiveFee	手数料有無の設定	No
-	aupunencenter		

表 8 – 前のページからの続き

次のページに続く

No	Parameter	Describe	推奨値
30	MaxCosignedAccount	最大署名数	25
31	FinalizationType	ファイナライゼーション方式	Deterministic
32	MaxTransactionperBlock	1ブロックの最大トランザクション数	6000
33	RestThrottring	API 接続数上限	30tps
34)	UnconfirmCacheSize	未承認トランザクション格納数	Small
35	UseLoadBalancer		Yes
		NLB 使用有無(プレビューは Yes 固定)	
36	LoadBalancerType	ロードバランサー配置タイプ	External
37)	Imageld	AMIID(変更不可)	変更不可
38	MPS3BucketName	S3 バケット名(変更不可)	変更不可
39	MPS3BucketRegion	S3 リージョン(変更不可)	変更不可
40	MPS3KeyPrefix	S3 キー Prefix(変更不可)	変更不可
(41)	mijinStackAlreadyExist		No
		他に mijin スタックがある場合、再作成時は「Yes」を選択	

表 8 - 前のページからの続き

日: パラメータの入力完了後、「Next」を押します。

# 2.2.3.11 Step.8

	esource Groups 👻 🦌	4	Tokyo 👻 Support 👻
CloudFormation > Stacks >	Create stack		
Step 1 Specify template	Configure stack optic	ons	
Step 2 Specify stack details	Tags You can specify tags (key-value pairs) to apply	y to resources in your stack. You can add up to 50 unique tags fo	or each stack. Learn more 🔀
Step 3 Configure stack options	Key	Vatue	R
Step 4 Review			m o v
	Add tag		6
	Permissions Choose an IAM role to explicitly define how Cl uses permissions based on your user credentia	loudFormation can create, modify, or delete resources in the sta als. Learn more 🖸	ick. If you don't choose a role, CloudFormation
	IAM role - optional Choose the IAM role for CloudFormation to u: IAM role na V Sample-role-n	se for all operations performed on the stack. arme	▼ Remove
	Advanced options		
	You can set additional options for your sta     Stack policy     Defines the resources that you want to pre-	ck, like notification options and a stack policy. Learn n otect from unintentional updates during a stack update.	nore L2
	Rollback configuration     Specify alarms for CloudFormation to more     back. Learn more	nitor when creating and updating the stack. If the operation bre	raches an alarm threshold, CloudFormation rolls it
	Notification options		
	Stack creation options		

日: とくに編集せず、赤枠の「Next」を押します。

2.2.3.12 Step.9

Pay langer     met       Image: Control     met       Image: Contro     met		Review MUIN-CATAPULT1		
Series a series a series of a series o		Step 1: Specify template		681
Notice of the second of the	starts details	Template		
<form><form>  Number of the second of the sec</form></form>	er slack spliters	Template URL		
		https://wijiv.staputi-christigatie-text.ch-ap-texthanir-Lanceorans.com/jireduci-	enterprise-energic/esponarult-pand	
International		Tash decaptor mje Calapuli brizystvenih multjale subrata VPC		
Projectorie       Description		Tallman call not wartable		
Pay 2-Service year of a service o				
<form>  Image: Imag</form>		Step 2: Specify stack details		0.61
Image:		Parameters (31)		
v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v		Q ₁ downly parameters		۰
National National   National		14 M	<ul> <li>Value</li> </ul>	
Abinorgin Bin   Abinorgin		Apittodivolumentari		
Wingtom       Big         Advances       Big		Apillodividumetica	500	
Participation Participation   Advitation Participation<		Aphropathianaings	11 Augu	
Indension Note   Indension Indension   Indension Indension <td></td> <td>ApiPingsiVelameTan</td> <td>-</td> <td></td>		ApiPingsiVelameTan	-	
Avinitionia memoria   Avinitionia memor		ApPlanmenthelarit	Public	
Antique generation de la persona de la pe		Aphotechencies		
A series of the second		Availability.com	sp-normali-10 sp-normali-14	
A series of a series o		Cataputition Kinear vion Fargerf ima	-	
		Grapublike Soulize	-	
main       mi         inputuo       mi <td></td> <td>Carloguillionweis</td> <td></td> <td></td>		Carloguillionweis		
Intermediate of the second		CatagolithumHole		
Additional of the second of		Default / Ind. Ser	abarta.	
Note of the second of		RealizationType	Probabilition	
A sequence of a sequen		Keyhane	101-0349-51	
Anderson and a set of the set		icethians Tge	ortemai	
A serie de la serie de		Record and a second and a secon		
Networksome in the second s		Per Bahistonian	100	
Networkson is a set of the set		PeerBod/Islamilie	100	
<pre>windows is in the second second</pre>		Permitanahas	things	
intentioned in a set of the se		Peortember (M.Nits	3	
Anional and and an a		Parkotister		
indiana de la construit de		Red Treating	200	
indexidation indexidantial		la vielane	HURI-CATAPULTI	
unique       N         unique       N         solutions       N         solution		UnarherGahillar	Mediam	
monitoring       monitoring         % 10 Critiqui etta etta etta etta etta etta etta ett		Test.collisteror	Test.	
Text - Strategy and and appropriate strategy and appropriete strategy and appropriate strategy and appropriate stra			10	
Fer 2         Immunity         Free many methods and		Step 3: Canfigure stack options		Dilli Dilli
Mentage     Y           Y		Tegs (0)		
Fg         • Max         •           Service		Q, local lap		
Second		Rep A Table		
Financial         Financial         Sea party		No taga		
Feederal           Second		Trace are no kap, defined for	Pri dali	
Image: Control of the second of the secon		Permissions		
Sciencing           Sciencing <td< td=""><td></td><td>These is no kild role associated as</td><td>ith this stack</td><td></td></td<>		These is no kild role associated as	ith this stack	
Field performance         Set approximation         Set approximation <t< td=""><td></td><td></td><td></td><td></td></t<>				
State and set of a set of		Stack policy		
Abids cardyaction      Keeperson      Keeperso		The starts policy		
		There is no shall policy in	Prost.	
		Builback configuration		
<pre>main set set set set set set set set set set</pre>		and the second se		
Image: Contract of the Contract		-		
Image: Control of the contro		Cinadelante alaren ANK		
Sectedate spaces           Bestimation optimal           Descention				
Bestender uppen           Bestender           Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestender Bestend		NetReation options		
Contract and the second and the		No application contemport	-	
Existence in equiplement     Existence in the initial ini		There are no initialize option	en defined	
Subcastical explorement     Subcastical explorement     Subcastical     S				
Relation to the Same Same Same Same Same Same Same Sam		Stack creation options		
Contract Contrect Contratt Contratt Contratt Contratt Contratt Contratt Contrac		Different and Mark		
The array of the		Builded		
bottom      class		Termuk		
balance      to the south that      the contrast of the south that the south the south the		Territation protection		
V DRAWNING		Dated		
Comparison      Compariso				
Port Heading Second 2 reads (california Date) (Second Second		> Suit-oursink		
the extension regard periodicity of the indication of the first order name within period the indication of the indi		» Dolb-oranitek Cagabilities		
Pro the surplus, BRI Doublement with register at an annuplus regulating DMBL/PLAND, DMBL/D Out for expetition of here means in service and the surplus and the surplus register at annuplus regulating DMBL/PLAND, DMBL/D Out for expetition of here         () Interviewing that DMD Could Provide register at annuplus regulating DMBL/PLAND, DMBL/D Out for expetition of here         () Interviewing that DMD Could Provide register at annuplus regulating DMBL/PLAND, DMBL/D Out for expetition of here         () Interviewing that DMD Could Provide register at annuplus regulating DMBL/PLAND, DMBL/D Out for expetition         () Interviewing that DMD Could Provide register at a more provide regulating that DMD Could Provide Register         () Interviewing		<ul> <li>Built-sensitivit</li> <li>Expanditions</li> <li>The following secondly regular spatialities (JMC) cloudlo-molect.cloud)</li> <li>The implies and also labelly and Learning second 2014 measure. Data its</li> </ul>	of you want to a make mathed these managem, and	hai Bey have
Ladioacting that BHI Cloud Investige register and AVI researce with outers names.     Ladioacting of an Arit Cloud Investige register for Internet prophyry     (ANAM TV, Arit C, BIONE		<ul> <li>Solid-section</li> <li>Petroleme secretal representation (MSC)/Solid-method/Solid Transporter secretal representation (Solid Construction Solid Society of the Solid Construction Solid Construction (Solid secretary construction)</li> </ul>	et yes and is a state matival from emouses and bat the calcular sames invariant million your MRI	hai Beylane
CANADJY, AUTO, DIANO	1	Solit-oraci list Equilities     Orthologia analysis (capital spatialities     Orthologia analysis (capital spatialities (orthologia))     Territologia analysis (capital spatialities (orthologia))     Territologia analysis (capital spatialities)     Territologia analysis (capitalities)     Territologia analysis     Territologia     Terri	of you and its analy natival fines encourse, and the callow stores are unique within you 1983 CAMBLETY, 6013, CAMBLE Oracle des supplities	Rai Beylaw I af these
	1	Exploration     Equilation     Containing sector (and or parts equilible (DMS classifier model))     The State sector (address exploration (DMS development))     The State sector (address exploration) and the sector (address exploration)     Containing address exploration (address exploration)     Containing address exploration     Containing address exploration     Containing address exploration     Containing address exploration	tel para antici la estate noti cui dinere encorren, en l' la l'De cattere stanes are artaga estate por Mitt GANALTY_MITO_DINALC_Ouch du sepatritivo antenenses.	And Broy Tanan 1 of These

日: ① の赤枠にて 2 つの項目にチェックを入れます。日: ② の赤枠の「Create Stack」を押します。エラー がなければ作成が始まります。

### 2.2.3.13 Step.10

CATAPULT-PRE1  Events Resources Outputs  h events	Parameters Template Change sets		
CATAPULT-PRE1	Parameters Template Change sets		
Events Resources Outputs     A events	Parameters Template Change sets		
1) h events			
th events			
P		Status	Status reason
6 17:48:36 UTC+0900	MUIN-CATAPULT-PRE1	CREATE_IN_PROGRESS	User Initiated
	6 17:48:36 UTC+0900	6 17:48:36 UTC-0900 MUIN-CATAPULT-PRE1	6 17.48.36 UTC-0900 MURI-CATAPULT-PRE1 (CREATE_IN_PROGRESS

日: Stack が始まり「CREATE_IN_PROGRESS」になっていることを確認してください。この状態はおよそ 20~30 分程度かかります。

# 2.2.3.14 Step.11

⊡ Stacks (6) ℃	MIJIN-CATAPULT1	Delete Update Stack actions <b>v</b> Create stack
Q MIJIN-CATAPULT1 X	Stack info Events Resources Outputs Parameters	Template Change sets
Active View Hester	Events (17)	C
NESTED MIJIN-CATAPULT1-loadBalancerNestStack -1HCA87RNUNK9Q	Q. Search events	6
2021-04-30 13:42:04 UTC+0900 CREATE_COMPLETE	Timestamp V Logical ID	Status Status reason
NESTED	2021-04-30 13:45:40 UTC+0900 MIJIN-CATAPULT1	⊘ CREATE_COMPLETE -
12223GK	2021-04-30 13:45:37 UTC+0900 loadBalancerNestStack	⊘ CREATE_COMPLETE -
CREATE_COMPLETE	2021-04-30 13:42:05 UTC+0900 loadBalancerNestStack	CREATE_IN_PROGRESS     Resource creation Initiated
NESTED MIJIN-CATAPULT1-macroNestStack-TFT74	2021-04-30 13:42:04 UTC+0900 loadBalancerNestStack	CREATE_IN_PROGRESS -
GMT7QPN 2021-04-30 13:22:32 UTC+0900	2021-04-30 13:41:59 UTC+0900 mijinNestStack	⊘ CREATE_COMPLETE -
CREATE_COMPLETE	2021-04-30 13:25:41 UTC+0900 mijinNestStack	CREATE_IN_PROGRESS     Resource creation Initiated
NESTED MIJIN-CATAPULT1-vpcNestStack-GGV80H	2021-04-30 13:25:40 UTC+0900 mijinNestStack	CREATE_IN_PROGRESS -
6B6NSL 2021-04-30 13:22:32 UTC+0900	2021-04-30 13:25:36 UTC+0900 vpcNestStack	⊘ CREATE_COMPLETE -
CREATE_COMPLETE	2021-04-30 13:23:33 UTC+0900 iamNestStack	⊘ CREATE_COMPLETE -
NESTED MIJIN-CATAPULT1-iamNestStack-1IU2EK1	2021-04-30 13:23:20 UTC+0900 macroNestStack	⊘ CREATE_COMPLETE -
RO8MA 2021-04-30 13-22-32 UTC+0900	2021-04-30 13:22:33 UTC+0900 macroNestStack	CREATE_IN_PROGRESS     Resource creation Initiated
CREATE_COMPLETE	2021-04-30 13:22:32 UTC+0900 vpcNestStack	CREATE_IN_PROGRESS     Resource creation Initiated
MIJIN-CATAPULT1	2021-04-30 13:22:32 UTC+0900 iamNestStack	CREATE_IN_PROGRESS     Resource creation Initiated
CREATE_COMPLETE	2021-04-30 13:22:31 UTC+0900 vpcNestStack	CREATE_IN_PROGRESS -
	2021-04-30 13:22:31 UTC+0900 macroNestStack	CREATE_IN_PROGRESS -
	2021-04-30 13:22:31 UTC+0900 iamNestStack	CREATE_IN_PROGRESS -
	2021-04-30 13:22:24 UTC+0900 MIJIN-CATAPULT1	CREATE_IN_PROGRESS     User Initiated

日: 「CREATE_COMPLATE」の状態であれば、mijin の作成が完了しました。

### 2.2.3.15 Step.12

MIJIN-CATA	PULT1 Delete Update Sta	ick actions 🔻	Create stack 🔻
Stack info Ev	vents Resources Outputs Parameters Template Change sets		
Outputs (6)			C
<b>Q</b> Search outputs	2		٥
Key 🔺	Value $\bigtriangledown$	Description	<pre></pre>
chainInfo	http://MIJIN-CATAPULT1-nlb-rest-0a16468e0cca959b.elb.ap-northeast- 1.amazonaws.com:3000/chain/info	-	-
emptyAddress	https://ap-northeast-1.console.aws.amazon.com/systems-manager/parameters/MIJIN- CATAPULT1/shares/nemesis_addresses.json	-	-
harvestAddress	https://ap-northeast-1.console.aws.amazon.com/systems-manager/parameters/MIJIN-CATAPULT1/shares/nemesis_addresses_harvesting.json	-	-
mijinEndpoint	http://18.183.121.6:3000,http://54.95.9.192:3000	mijin Catapult Rest Endpoint	-
mijinLBENdpoint	http://MIJIN-CATAPULT1-nlb-rest-0a16468e0cca959b.elb.ap-northeast-1.amazonaws.com:3000		-
nodePeers	http://MIJIN-CATAPULT1-nlb-rest-0a16468e0cca959b.elb.ap-northeast- 1.amazonaws.com:3000/node/peers	-	-

日: 作成した Stack の「Outputs」を押すと、作成された mijin の設定情報を確認できます。 ※ 以下表は、新規ネットワーク版と同じのため省略

	mijinLBEndpoint
	ロードハランサーを通した mijin の API エンドホイントです。API ノードを 負荷分散しますが、ソース IP によるスティッキーセッションが有効です。 詳細はこちら
	mijinEndpoint
	API ノード(EC2 インスタンス)の直接アクセス用の API エンドポイントで す。ロードバランサーを介さずに接続できます。
"working", "0"           "working", "0"     <	chainInfo
	mijin の現在のブロック数を確認できます。ブロック数が「2」以上である ことを確認してください。
	<b>harvestAddress</b> AWS Systems Manager パラメータストアに登録された、通貨分配用アドレ スへのリンクです。
Millionations (International International I	
Nit Mat Oxto Hit Websening, attention at Termination at the second attention atten	
Visit         List and the set of	
Van, Jay, Y. The second and the s	
American Construction and American Constr	
A manufactor Value (CARLEND CARLEND CA	
A series of the	emptyAddress
	AWS Systems Manager パラメータストアに登録された、未使用アドレスの リンクです。
<ul> <li>Vessari's 0, "vessari's 0, "vessari's 0, "vessari's 0, "vessaritheren's "FAUPTION 1316137" 000002396/978308 (FAUSTOCK150080468/533361744000", "vessaritheren's 0, "vessaritheren's 0,</li></ul>	
- Transient Berley Transient Der Greine seiner Franz Transient Der Greine seiner Franz Transient Der Greine Seiner Franz Franz Berley Franz Berley Fra	
"     The second	
************************************	
"Value"         State	
Verwart B,     V	
"part": 7880.           "methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/methods/meth	
	nodePeers
	mijin API からノードの接続状態を確認できます。API ノード1台と設定さ れた PEER ノード数が表示されていれば正常です。

#### 表 9: mijin エンドポイントと確認項目

日: これで mijin Catapult を使用する準備が整いました。それでは次の項で操作を始めてみましょう!

# 2.2.4 トライアル版の mijin をデプロイする

トライアル版では、パラメーターを変更することで環境にあったネットワークを構築することが可能です。 以下にパターン例を示します。

日: 本ページは AWS マーケットプレイス内にある cloudformation から mijin Catapult トライアル版を起 動する手順となります。

2.2.4.1 AWS 使用サービス

- Amazon EC2
- Amazon EBS
- Amazon Route53
- Amazon VPC(Nat Gateway)
- パラメータストア

#### 2.2.4.2 View Network

#### 注釈:

トライアル版では、シングル AZ のみの構成となっています。 配置はシングルリージョンとなりますが、世界 20 リージョン毎にデプロイすることが可能です。



### 2.2.4.3 Step.1



日: mijin Catapult Free Trial の AMI を使用するためにサブスクライブする必要があります。赤枠のボタン を押してください。

#### 2.2.4.4 Step.2

mijin Cat	tapult (v.2) Free Trial	Continue to C You must first rev
< Product Detail <u>Subscribe</u> Subscribe to this s To create a subscription, review the	oftware pricing information and accept the terms for this software.	
Terms and Conditions Tech Bureau Holdings, Corp. Or By subscribing to this software, you as <u>License Agreement (EULA)</u> C ² . You also information about this transaction (in seller, reseller or underlying provider, <u>Notice</u> C ² . Your use of AWS services is	ffer gree to the pricing terms and the seller's <u>End User</u> o agree and acknowledge that AWS may share cluding your payment terms) with the respective as applicable, in accordance with the <u>AWS Privacy</u> subject to the <u>AWS Customer Agreement</u> ^C or other	
agreement with AWS governing your of the following table shows pricing info	use of such services.	
mijin Catapult (v.2) Free Trial	Additional taxes or fees may apply. mijin Catapult (v.2) Free Trial EC2 Instance Type Software/hr	

日: mijin Catapult AMI を使用するため、使用の承認をしてください。

### 2.2.4.5 Step.3



mijin Catapult Free Trial CFT 0.9.6.4

CloudFormation Template

This Cloudformation Stack makes it easy to build mijin.

Users create a new VPC and place a single EC2 instance in a public network.

After the Cloudformation execution is complete, the mijin address data is stored in the System Manager's Parameter Store. Using symbol-cli without having to remotely log in to the server, users are able to use the mijin immediately.

- View Template Components
- View Usage Instructions
- Close CloudFormation Template



日: 図のようなネットワークを AWS CloudFormation によって作成するため、赤枠の文字をクリックして ください。

### 2.2.4.6 Step.4

aws Services ~	Resource Groups 👻 🛠	۵	Tokyo 👻	Support 👻
🖬 🕆 🎓 🔂 🗹 Close		Choose template language: ○ JSON ● YAML @	±02	
Resource types • DirectoryService • DocDB • DynamoDB • EC2 • ECR • ECS • EFS	File: "template1"	mijinNest Stack		() +
			Messages	880
template1 1 AWSTemplateFormatVersion: 2 Description: mijin Catapul 3 - Parameters: 4 - ServiceName: 5 Description: Please in Components Template	2010-09-09 t with multiple subnets VPC put Service Name.	Choose template language: O JSON   YAML		

### 日: とくに編集せず、雲マークの CreateStack を押します。

### 2.2.4.7 Step.5

Services     I       CloudFormation     Stacks       Step 1     Specify template       Step 2     Specify stack details       Step 3     Configure stack options       Step 4     Review	Create stack	
Step 2 Specify stack details	Prerequisite - Prepare template	
CloudFormation > Stacks ; Step 1 Specify template Step 2 Specify stack details Step 3 Configure stack options Step 4 Review	Prepare template Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want t the stack.	o incluc
Step 4	Template is ready     Use a sample template     Create template in Design	er
Review		
	Specify template A template is a JSON or YAML file that describes your stack's resources and properties.	
	Template source Selecting a template generates an Amazon 53 URL where it will be stored.	
	Amazon S3 URL     Upload a template file	
	Amazon S3 URL	
<complex-block> </complex-block>		
	Amazon S3 template URL	

日:とくに編集せず、赤枠の「Next」を押します。

# 2.2.4.8 Step.6

	tacks ) Counts	etack			
CloudFormation > S	tacks > Create	stack			
Step 1 Specify template		Specify stack details			
Specify template					
Step 2 Specify stark details		Stack name			
Specify stack details		Stack name			
Step 3 Configure stack option	. 1	Enter a stack name			
configure stack option		Stack name can include letters (A-Z and a-z)	), numbers (0-9), and dashes (-).		
Step 4					
Neview		Parameters			
		Parameters are defined in your template and	d allow you to input custom values when you create or update	e a stack.	
		VPC Configuration			
		ServiceName			
	2	MIJIN-CATAPULT			
		AvailabilityZone3			
	3	Please input AvailabilityZone1			
	9	ap-northeast-1c			
		Security Group Configuration			
		PublicLocationIP Please input range of IP addresses that can a	access mijin rest		
	4	0.0.0.0/0			
		Node Configuration			
		DefaultUnixUser Please Input Default Unix User			
	9	ubuntu			
		KeyName			
	6	Name of an existing EC2 KeyPair to enable S	SSH access to the api and peer instances		
		API Node Configuration			
		DualinstanceType			
	(7)	t3.large			
	0	milin Configuration			
		CatapultVersion			
		Please select mijin Catapult Version			
		for a state of the			
		CatapultShareMode Please select mijin Catapult Share Mode			
		ssm			
		CatapultNetwork			
		Please select mijin Catapult Network Identif	ñer		
		- April 100			
		CatapultBlockGenerationTargetTime Please select mijin Catapult Block Generation	on Target Time		
	8	60s			
				9	
				Cancel Previous	5 Ne

日: パラメータを入力します。

No	Parameter	Describe
1	Stack Name	このスタックにおける名前を記載してください。
2	Service Name	
		フタックによって作成されスサービスタを記載してください。
		インシンによりで15%とれるシービス石を記載してくたとい。 全リソースの冠名として使用されます。
3	Availability Zone	お使いのリージョンのアベイラビリティゾーンを選択してください。
4	Public Location IP	
		mijin Catapult の API への接続許可 IP アドレスを指定してください。
		IP アドレスはレンジでも可能です(/24 など)
5	Default UnixUser	作成する EC2 インスタンスの標準の Unix ユーザーを記載してください。
6	KeyName	
		作成する EC2 インスタンスのリモート接続用の SSH 鍵を選択してください。
		表示されていない場合、事前に鍵を作成する必要があります。
		鍵の作成方法はこちらをご確認ください。
7	Dual InstanceType	作成する EC2 インスタンスのスペックを選択してください。
8	Catapult	mijin のブロック生成時間を選択してください。30 秒か 60 秒のみを選択できます。
	BlockGenerationTarget	ime

日:パラメータの入力完了後、⑨の「Next」を押します。

# 2.2.4.9 Step.7

aws Services ~	Resource Groups 👻 🕏	Ą	Tokyo 👻 Support 👻
CloudFormation > Stacks	> Create stack		
Step 1 Specify template	Configure stack optio	ns	
Step 2 Specify stack details	<b>Tags</b> You can specify tags (key-value pairs) to apply	to resources in your stack. You can add up to 50 unique tags i	for each stack. Learn more 🔀
Step 3 Configure stack options	Key	Value	Re
Step 4 Review			m o v e
	Add tag		
	Permissions Choose an IAM role to explicitly define how Cle uses permissions based on your user credential	udFormation can create, modify, or delete resources in the st s. Learn more 🕻	tack. If you don't choose a role, CloudFormation
	IAM role - optional Choose the IAM role for CloudFormation to use IAM role na ▼ Sample-role-na	for all operations performed on the stack. The	▼ Remove
	Advanced options		
	You can set additional options for your star Stack policy Defines the resources that you want to pro-	k, like notification options and a stack policy. Learn in the stack policy stack update.	more 🖸
	Rollback configuration Specify alarms for CloudFormation to mon back. Learn more	itor when creating and updating the stack. If the operation br	reaches an alarm threshold, CloudFormation rolls it
	Notification options		
	Stack creation options		
			Cancel Previous Next
Feedback 🔇 English (US)		© 2008 - 2020, Amazon Web Services, Inc. or its affiliate	s. All rights reserved. Privacy Policy Terms of

日:とくに編集せず、赤枠の「Next」を押します。

### 2.2.4.10 Step.8

aws services - Re	essunce Groups - 1	Q. Takyo - Support -						
CloseProvation > Status >	Chole dask							
910p 1	Provine MUN CATADULT TEST	TPIAL1						
Specify template	Review MIDIN-CATAPOLI-TEST	-IRIAL I						
Step 2 Specify stack details	step 1: speciny template							
	Template							
Configure stack options	Template URL							
2004	https://cl-ap-northeast-1.amazonaws.com/cf-templates	1u/reduced10pf-ap-northeast-1/20203131k70-tempiate14hjustaling						
	Stack description mijim Catapult with multiple subnets VPC							
	tistimate cen 🕑							
	Step 2: Specify stack details	Cristian Contract Con						
	Parameters (10)							
	Q, Search parameters							
	Ray	A Value T						
	Availability Done 1	ap-month-east-tic						
	CatapultificaleGenerationTargetTime	60n						
	CatapullNetwork	mijin-best						
	CatapultihareMade	428						
	Catapulitienion	v0964						
	DefaultUnitUser	abenta 15 Janua						
	Keyhane	test-oppine-ty						
	Publiclecitier/P	63.0.5/3						
	ServiceName	MUNI-CATAPULT-TEST-TRIAL1						
	Step 3: Configure stack options	Ed.						
	Tags (0)							
	Q, teach tops							
	1.17	Value						
		No taga						
	There an	no tags defined for this stack						
	Permissions							
	No permissions There is no bidd rate permitted with this starts							
	There is no I	AH role associated with this stack.						
	Stack policy							
		No stack palicy						
	The	t is ne stack policy defined						
	Rollback configuration							
	Hositoring time							
	CoudWatch alarm ARN							
	1							
	Notification options							
	Houristian optimu							
		io natification options						
	There are	no netification options defined						
	Stack creation options							
	Rollback on failure							
	Enabled							
	Timeout							
	Termination protection							
	Disabled							
	<ul> <li>Quick-create link</li> </ul>							
	Capabilities							
	③ The following resource)() require capabilities: [KW	i: ClaufFormation: Stack)	1					
	This template contains identity and Access Managem and that they have the minimum required permission	ent DAMD researces. Check that you want to create each of these resources is in addition, they have custom names. Check that the outcom names are	1					
	For this template, AHS Claudiformation might require	t an annotagenized capability: CAMABUTY, AUTO, 200440. Check the	I					
U	capabilities of these resources.							
	<ul> <li>I acknewledge that AWS CloudFormation might</li> <li>I acknewledge that AWS CloudFormation might</li> </ul>	create war desources with castom names. require the following capability:	1					
	CAPIBILITY_AUTO_EXPANS							
		Cancel Previous Create Previous Create stack						

日: ① の赤枠にて 2 つの項目にチェックを入れます。日: ② の赤枠の「Create Stack」を押します。エラー がなければ作成が始まります。

### 2.2.4.11 Step.9

aws Services	<ul> <li>Resource Groups</li> </ul>	× 🖈	Δ	Tokyo 👻 Support 👻
CloudFormation >	Stacks > MIJIN-CATAPULT-	TEST-TRI	L1	
Stacks (2) MIJIN-CATAPULT	C	×	MIJIN-CATAPULT-TEST-TRIAL1       Delete     Update       Stack actions       Create stack	
Active 🔻 🔍	View nested	>	Stack info Events Resources Outputs Parameters	Template Change sets
NESTED MIJIN-CATAPULT-TEST BNUHF1074P 2020-08-19 10:00:04 UTI CREATE_IN_PROGRES	T-TRIAL1-mijinNestStack-X C+0900 S	0	Events (1) Q. Search events	C
MIJIN-CATAPULT-TES 2020-08-19 09:59:57 UT (CREATE_IN_PROGRES	T-TRIAL1 C+0900 IS	•	Timestamp v Logical ID Status	⊗ Status reason
L			2020-08-19 09:59:57 MIJIN-CATAPULT- UTC+0900 TEST-TRIAL1 GRESS	Jser Initiated

日: Stack が始まり「CREATE_IN_PROGRESS」になっていることを確認してください。この状態はおよそ 15~20 分程度かかります。

#### 2.2.4.12 Step.10

Services v Resource Groups v	*	4		Tokyo 👻 Support 👻
CloudFormation > Stacks > MIJIN-CATAPULT-TEST-	TRIAL1			
⊡ Stacks (2) C	MIJIN-CATAPUI	T-TEST-TRIA	AL1	_
Q MIJIN-CATAPULT-TEST-TRIAL1 X	Delete Update	Stack actions 🔻	Create stack 🔻	
Active  View nested  View nested  View nested	Stack info Events	Resources Out	puts Parameters	Template Change sets
NESTED MIJIN-CATAPULT-TEST-TRIAL1-mijinNestStack-X BNUHF1074P 2020-08-19 10:00:04 UTC+0900 CREATE_IN_PROGRESS	Events (5)			
MIJIN-CATAPULT-TEST-TRIAL1         ●           2020-08-19 09:59:57 UTC+0900         ●           Ø CREATE_COMPLETE         ●	Timestamp 🔻	Logical ID	Status	(2) Status reason
	2020-08-19 10:10:44 UTC+0900	MIJIN-CATAPULT- TEST-TRIAL1	⊘ CREATE_COMPL ETE	
	2020-08-19 10:10:42 UTC+0900	mijinNestStack	O CREATE_COMPL ETE	
	2020-08-19 10:00:05 UTC+0900	mijinNestStack	CREATE_IN_PRO GRESS	Resource creation Initiated
	2020-08-19 10:00:03 UTC+0900	mijinNestStack	CREATE_IN_PRO GRESS	
	2020-08-19 09:59:57 UTC+0900	MIJIN-CATAPULT- TEST-TRIAL1	CREATE_IN_PRO	User Initiated
	CloudFormation > Stacks > MUIN-CATAPULT-TEST- Stacks (2) C MIJIN-CATAPULT-TEST-TRIAL1 X Active V View nested MIJIN-CATAPULT-TEST-TRIAL1-miJinNestStack-X BNUHF1074P 2020-08-19 100:59:57 UTC+0900 C CREATE_IN_PROGRESS MIJIN-CATAPULT-TEST-TRIAL1 2020-08-19 09:59:57 UTC+0900 C CREATE_COMPLETE	CloudFormation > Stacks > MUIN-CATAPULIT-TEST-TRAL1 Stacks (2) C MUIN-CATAPULIT-TEST-TRIAL1 Active  View nested C NESTED MUIN-CATAPULIT-TEST-TRIAL1-mijinNestStack-X BNUHF1074P 2020-08-19 10:00:04 UTC-0900 C CREATE_COMPLETE MUIN-CATAPULIT-TEST-TRIAL1 2020-08-19 10:10:44 UTC-0900 C CREATE_COMPLETE C CREATE_C CREATE_COMPLETE	CloudFormation > Stacks > MUIN-CATAPULT-TEST-TRIAL1 Stacks (2) MUIN-CATAPULT-TEST-TRIAL1 Active  View nested 1 ) NESTED MUIN-CATAPULT-TEST-TRIAL1-mijinNestStack-X BNUHF1074P 2020-08-19 00:59:57 UTC-0900 C CREATE_COMPLETE MUIN-CATAPULT-TEST-TRIAL1 2020-08-19 10:10:44 MUIN-CATAPULT-TEST-TRIAL1 2020-08-19 10:10:42 MUIN-CATAPULT-TEST-TRIAL1 2020-08-19 10:00:05 mijinNestStack 2020-08-19 10:00:05 mijinNestStack 2020-08-19 10:00:05 mijinNestStack 2020-08-19 10:00:05 mijinNestStack	CloudFormation > Stacks > MUIN-CATAPULT-TEST-TRIAL Stacks (2) MUIN-CATAPULT-TEST-TRIAL1 Active  View nested C 1 >> NESTED MUIN-CATAPULT-TEST-TRIAL1-mijinNestStack-X BNUHF1074P 2020-08-19 10:00:04 UTC+0900 C CREATE_NP, PROCRESS MUIN-CATAPULT-TEST-TRIAL1 2020-08-19 00:95957 UTC+0900 C CREATE_COMPLETE C CREATE_COMPLETE 2020-08-19 10:10:42 UTC+0900 MUIN-CATAPULT- TEST-TRIAL1 2020-08-19 10:00:05 mijinNestStack C CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_PRO CREATE_IN_

### 日: 「CREATE_COMPLATE」の状態であれば、mijin の作成が完了しました。

# 2.2.4.13 Step.11

	117 Hz F							
🗆 Stacks (2)								
Q MIJIN-CATAPULT-TEST-TRIAL1 X								
Active  View nested  Stack info Events Resources Outputs Parameters Template Change set								
NESTED MIJIN-CATAPULT-TEST-TRIAL1-mijinNestStack-X BNUHF1074P	Outputs (4)							
2020-08-19 10:00:04 UTC+0900 CREATE_IN_PROGRESS	Q Search outpu	ts						
MIJIN-CATAPULT-TEST-TRIAL1								
2020-08-19 09:59:57 UTC+0900  CREATE_COMPLETE	Key 🔺	Value 🗢	Description ⊽ Exp nam					
	chainHeight	http://54.168.116.90:3000/chain/height						
	emptyAddress	https://ap-northeast-1.console.aws.amazon.com/systems- manager/parameters/MUIN-CATAPULT-TEST- TRIAL1/shares/nemesis_addresses.json						
	harvestAddress	https://ap-northeast-1.console.aws.amazon.com/systems- manager/parameters/MIJIN-CATAPULT-TEST- TRIAL1/shares/nemesis_addresses_harvesting.json						
	mijinEndpoint	http://54.168.116.90:3000	mijin Catapult					

日: 作成した Stack の「Outputs」を押すと、作成された mijin の設定情報を確認できます。

	mijinEndpoint							
	miiin の ADI エンドポイントです							
	mijin の API エント小イント じゅ。							
("height" "56")								
	chainHeight							
	mijin の現在のフロック数の確認かできます。フロック数か「2」以上にな							
	っていることを確認してください。							
💯 Baratan - Pananan Sanga - S 👌 Talay - Jugani -								
ANS Server Revert > Reverse Start > Rev								
Overview Statery Tops								
Name Excorption Addukt/CEST/ENAL/phoreab_addenses_haveship.phor - Tex								
те Бакари Бака 1904 Lat nodific cor								
Stolg among WMI (1887) Second vice (VIII) TEST THE Legis.     MODULATION OF CONTRACT ON THE CONTRACT OF CONTRACT ON THE C								
"Small, ap", "Search and "States Control And "								
Jaugest, 260,0319490021041069440611400006     Jaugest, 260,031940021041069440611400006     Cannel Jackstrateger Jassertinger Statement Connel (2000)								
Ľ								
Freehen: O fogle (A) 6208-208, boson kin binner, to urbuildens, Mright sources. Priory Velay, Served Ser	narvestAddress							
	AWS Systems Manager パラメータストアに登録された、基軸通貨(トライ							
	アル版のため 2000cat currency) を分配したアドレスのリンクです							
2013 Bardam - Pasapas Gauga - \$								
Ath Spenis Tarage () Formatic Star () Allah Gold AS-100-5004 (Vanadamine) Athenadamine) Commer (MURIN III, CATADURI T, TEST, TORIAL 1 (charges (Ageneration and Ageneration and Agene								
THE REAL PROPERTY AND A DESCRIPTION OF THE REAL PROPERTY								
Overlage Million Control of Contr								
Many Trap								
The second secon								
Construction of the second secon								
The second								
Image: Second control of the second control								
Image: Second control of the second control								
Image: Section of the section of t								
Image:								
Image:								
	emptyAddress							
	emptyAddress							
	<b>emptyAddress</b> AWS Systems Manager パラメータストアに登録された、未使用のアドレス							
	<b>emptyAddress</b> AWS Systems Manager パラメータストアに登録された、未使用のアドレス のリンクです。							

表 10: mijin エンドポイント確認項目(トライアル)

日: これで mijin Catapult を使用する準備が整いました。それでは次の項で操作を始めてみましょう!

# 2.2.5 AWS MarketPlace テクニカル資料

AWS MarketPlace における様々な技術資料を追加します。

### 2.2.5.1 AWS Marketplace Cloudformation パラメーター比較表

本章では、AWS Marketplace の mijin Catapult(v.2) をデプロイする際のパラメーターの説明とデフォルト 値を説明します。 製品版2製品及びトライアル版が異なるデフォルト値であることを確認することができます。

### Cloudformation パラメーター比較表

	表 11: Trial / Product 設定一覧								
No	っカ テ ゴ リ ー	設定名	説明	設 定 値	制限值	Trial 設定 可否	Trial デフ ォルト値	Product NewVPC デフォルト 値	Product ExistsVPC デフォルト 値
1	VPC Configu	ServiceName Iration	リソースの冠名となるサ ービス名を指定します。	String	記 号 始 ま り NG、大文字小 文字英数字、ダ ッシュ (-) 使 用可能	0	MIJIN- CATAPULT	MIJIN- CATAPULT	MIJIN- CATAPULT
2		AvailabilityZone.	1 VPC で使用する AZ を指 定します。	List	リージョンの AZ	0	リージョン に依存	リージョン に依存	•
3		AvailabilityZone:	2 VPC で使用する AZ を指 定します(AZ1 とは別を 指定)	List	リージョンの AZ	•	リージョン に依存	リージョン に依存	•
4		VPC	すでに存在する VPC ld を指定します。	List	リージョンに ある VpcId	•	•	•	0
5		VpcCidrBlock	VPC の IP レンジを 指 定 (例:vpc-xxxx (xx.xx.xx.xx/16))	List	正規表現: (d{1,3}.){3}d{1,3}	/d{1,2} •	•	•	0
6		Public1	公開ネットワーク(AZ1) のサブネットを指定	List		•	•	•	0
7		Public2	公開ネットワーク(AZ2) の サ ブ ネ ッ ト を 指 定 (Public1 と異なるもの)	List		•	•	•	0
8		Private1	非 公 開 ネ ッ ト ワ ー ク (AZ1) のサブネットを指 定	List		•	•	•	0
9		Private2	非 公 開 ネ ッ ト ワ ー ク (AZ2) のサブネットを指 定(Private1 と異なるも の)	List		•	•	•	0
10		InternalDomainI	Na <b>肉</b> e部 DNS 名 (例: mijin.internal)を指定	String		•	•	mijin.interna	l mijin.interna
11	Securit Group Configu	y PublicLocationIF Iration	'mijin エンドポイントへ の接続許可 IP レンジ	String	正規表現: (d{1,3}.){3}d{1,3}	/d{1,2}			
12	Node Configu	DefaultUnixUser iration	EC2 リモートログインユ ーザー名	String	"catapult" 以 外	0	ubuntu	ubuntu	ubuntu
13		KeyName	EC2 で使用する SSH 鍵	List		0	リージョン に依存	リージョン に依存	リージョン に依存
14	API Node Configu	ApiPlacementNe iration	tw <b>AGH</b> kノード配置(Public または Private)	List	Public, Private	•	•	Public	Public
15		ApiInstanceType	API インスタンスのスペ ック	List	複数	Δ	t3.large	t3.large	t3.large
16		ApiRootVolumeS	iz <b>d</b> レートボリュームの容量 (GB)	List	30, 100	•	30	30	30
17		ApiBlockVolume	Si <b>延</b> ロックデータ用のディ スク容量(GB)	List	50, 300, 500, 800,1000	•	50	500	500

次のページに続く

				K 11 1631					
Nc	っカ テ ゴ リ ー	設定名	説明	設 定 値	制限值	Trial 設定 可否	Trial デフ ォルト値	Product NewVPC デフォルト 値	Product ExistsVPC デフォルト 値
18		ApiBlockVolume	lopぢロックデータ用 IOPS (gp3)	List	3000, 5000, 10000	•	100	3000	3000
19		ApiMongoVolum	e <b>Swizze</b> ngoDB 用のディスク 容量(GB)	List	50, 300, 500, 800,1000	•	50	300	300
20		ApiMongoVolum	el <b>めp</b> oongoDB用IOPS(gp3)	List	3000, 5000, 10000	•	300	3000	3000
21	PEER Node Configu	PeerNumberOfU ration	niBEERノード台数(固定)	Int	3–10	•	•	3	3
22		PeerInstanceTyp	e PEER ノードのインスタ ンスタイプ	List		•	•	t3.large	t3.large
23		PeerRootVolume	Si⊉EER ノードのルートデ ィスク容量	List	30, 100	•	•	30	30
24		PeerBlockVolum	e <b>Sh£e</b> R ノードのブロック データ用容量	List	50, 300, 500, 800,1000	•	•	500	500
25		PeerBlockVolum	elðþar ノードの IOPS (io1)	List	3000, 5000, 10000	•	•	3000	3000
26	mijin Configu	CatapultVersion ration	起動時の Catapult バー ジョン	List	v10037,v10038	Δ	v10038	v10038	v10038
27		CatapultShareM	po初期データ (アドレス等) の保存場所(SSM 推奨)	List	ssm	Δ	ssm	ssm	ssm
28		mijinDataDiretor	y <b>N^{iam}タ</b> マウント先のパス を指定	String	絶対パス	•	/mnt/mijin	/mnt/mijin	/mnt/mijin
29		CatapultNetworl	i mijin のネットワークタ イプ	List	mijin, mijin- test	Δ	mijin-test	mijin	mijin
30		CatapultBlockGe	neration加建成間隔e(目安)	List	5s, 15s, 30s, 60s	Δ	60s	15s	15s
31		CatapultEffective	eF <b>d</b> eランザクション手数料 有無	Boolea	n Yes, No	•	Yes	No	No
32		MaxCosignedAcc	い最大連署アカウント数	List	25, 50, 100, 1000	•	25	25	25
33		FinalizationType	ファイナライズ方式		Deterministic, Probabilistic	•	Deterministic	Deterministic	Deterministic
34		MaxTransaction	e <b>rびロck</b> ク内最大トランザ クション数	List	6'000, 10'000, 20'000, 50'000, 100'000	•	6'000	6'000	6'000
35		RestThrottring	API 接続数(バースト時 +100)	List	30tps, 100tps, 200tps, 500tps, NoLimit	•	30tps	30tps	30tps
36		UnconfirmCache	Si <b>表</b> 承認トランザクション のキャッシュサイズ	List	Small, Medium, Large	•	Small	Small	Small
37	LoadBa Configu	la <b>dse</b> ŁoadBalance ration	r NLB を使用するか	Boolea	n Yes, No	•	•	Yes	Yes
38		LoadBalancerTy	peNLB の配置場所	List	external, internal	•	•	external	external
39	Other	mijinStackAlread	y <b>®i</b> すタックが存在するか の指定	Boolea	n Yes, No	•	•	No	No

表 11 – 前のページからの続き

### 2.2.5.2 AWS MarketPlace Cloudformation 仕様

AWS MarketPlace にて提供している、mijin Catapult(v.2) は、オーケストラレーションツールである Cloudformation にてデプロイを行なっています。 本章では、Cloudformation を使って作成される AWS リソースの説明をします。

Cloudformation Template(CFT) の構成は複数のファイルから構成されており、親 Stack が子の Stack を それぞれ呼び出しています。親 Stack のパラメータによって、呼び出される子の Stack が変わります。



#### macroNestStack

macroNestStack では、Cloudformation Macro を作成します。パラメータ mijinStackAlreadyExist を Yes の場合、Macro を作成しないスタックが (empty-macro) を呼び出します。これは、Cloudformation Macro がユニークな名前で作成するため、複数同一名で作成することができないためです。

Cloudformation Macro は Lambda(Node.js) で作成されており、この後に出てくる mijinNestStack の CFT を指定パラメータによって変換します。

この作成した Lambda から Amazon CloudWatch Logs への読み書き権限を許可をするため、以下の IAM ロール及びポリシーを割り当てます。

```
Resources:
 PeerUnitsExecutionRole:
 Type: 'AWS::IAM::Role'
 Properties:
 AssumeRolePolicyDocument:
 Version: 2012-10-17
 Statement:
 - Effect: Allow
 Principal:
 Service:
 - lambda.amazonaws.com
 Action:
 - 'sts:AssumeRole'
 Path: /
 Policies:
 - PolicyName: root
 PolicyDocument:
 Version: 2012-10-17
 Statement:
 - Effect: Allow
 Action:
 - 'logs:CreateLogGroup'
 - 'logs:CreateLogStream'
 - 'logs:PutLogEvents'
 Resource:
 - Fn::Join:
 - 1:1
 - 'arn:aws:logs'
 - Ref: 'AWS::Region'
 - Ref: 'AWS::AccountId'
 - 'log-group:/aws/lambda/*:*:*'
```

Macro では、パラメータ PeerNumberOfUnits の数に応じて、EC2 インスタンスのスタックを複製し、起 動する EC 2インスタンスを動的に変更することができます。

### 警告:

同一リージョンに複数の mijin Catapult(v.2) を作成する場合は、mijinStackAlreadyExist を YES にす る必要があります。

#### vpcNestStack

vpcNestStack では、新たに VPC を作成します。マルチ AZ 環境を構築し、各 AZ にパブリックサブネット・ プライベートサブネットを配置します。また、プライベートネットワークのルーティングのデフォルトゲ ートウェイは同じ AZ にあるパブリックネットワークに配置した Nat Gateway を使用してインターネット に出ることができます。


注釈: 既存ネットワークに mijin を展開する場合は、このスタックは使いません。

### iamNestStack

iamNestStack では、EC2 インスタンスが使用する IAM ロールおよびそれに紐づく IAM ポリシーを作成します。

以下は、API ノード、PEER ノードそれぞれに割り当てるロールです。

```
AWSApiAccessRole:
Type: 'AWS::IAM::Role'
Properties:
AssumeRolePolicyDocument:
Version: 2012-10-17
Statement:
- Effect: Allow
Principal:
Service:
```

(continues on next page)

(continued from previous page)

```
- ec2.amazonaws.com
 Action:
 - 'sts:AssumeRole'
 Path: /
AWSPeerAccessRole:
 Type: 'AWS::IAM::Role'
 Properties:
 AssumeRolePolicyDocument:
 Version: 2012-10-17
 Statement:
 - Effect: Allow
 Principal:
 Service:
 - ec2.amazonaws.com
 Action:
 - 'sts:AssumeRole'
 Path: /
```

AWS Systems Manager Session Manager の権限を付与します。

AWS Systems Manager Session Manager では、EC2 インスタンスに IAM 権限でリモートログインできる ように権限を付与し、IAM ロールに紐づけます。

```
AWSSSMRolePolicies:
 Type: 'AWS::IAM::Policy'
 Properties:
 PolicyName: AWSSSMAccessPolicy
 PolicyDocument:
 Version: 2012-10-17
 Statement:
 - Effect: Allow
 Action:
 - 'ssm:DescribeAssociation'
 - 'ssm:GetDeployablePatchSnapshotForInstance'
 - 'ssm:GetDocument'
 - 'ssm:GetManifest'
 - 'ssm:GetParameters'
 - 'ssm:ListAssociations'
 - 'ssm:ListInstanceAssociations'
 - 'ssm:PutInventory'
 - 'ssm:PutComplianceItems'
 - 'ssm:PutConfigurePackageResult'
 - 'ssm:UpdateAssociationStatus'
 - 'ssm:UpdateInstanceAssociationStatus'
 - 'ssm:UpdateInstanceInformation'
 Resource: '*'
 - Effect: Allow
 Action:
 - 'ssmmessages:CreateControlChannel'
 - 'ssmmessages:CreateDataChannel'
 - 'ssmmessages:OpenControlChannel'
 - 'ssmmessages:OpenDataChannel'
 Resource: '*'
```

AWS Systems Manager パラメータストアへの読み書き権限を付与します。 AWS Systems Manager パラメータストアでは、mijin Catapult(v.2) の最初のノードの動的生成されたデ ータを保存し、その他の各ノードから参照できる権限を付与し、IAM ロールに紐づけます。

```
AWSPSAccessRolePolicies:
Type: 'AWS::IAM::Policy'
Properties:
PolicyName: AWSPSAccessRole
PolicyDocument:
Version: 2012-10-17
Statement:
- Effect: Allow
Action:
- 'ssm:PutParameter'
- 'ssm:GetParameter'
- 'ssm:GetParametersByPath'
```

同一アカウント同一サービス名の IAM ロールのリソースからの Security Token Service(STS) の付与し、 IAM ロールを紐づけます。

STS が付与されていることで、AWSApiAccessRole 及び AWSPeerAccessRole の IAM ロールはポリシー指 定した AWS サービスを操作できるようになります。

```
AWSAssumeAccessRolePolicies:
 Type: 'AWS::IAM::Policy'
 Properties:
 PolicyName: AWSAssumeAccessRole
 PolicyDocument:
 Version: 2012-10-17
 Statement:
 - Effect: Allow
 Action:
 - 'sts:AssumeRole'
 Resource:
 - Fn::Join:
 - ':'
 - 'arn:aws:iam:'
 - Ref: 'AWS::AccountId'
 - !Sub "role/${ServiceName}*"
 Roles:
 - !Ref AWSApiAccessRole
 - !Ref AWSPeerAccessRole
```

### s3NestStack

**警告:** この機能は現在無効化されており、パラメーターストアのみしか保存することができません。

s3NestStack では、mijin のデータを AWS Systems Manager パラメータストアに配置するか、S3 バケットに保存するかを選択し、S3 を選択した場合、S3 バケットを作成します。

### mijinNestStack

mijinNestStack では、Route53 によるプライベート DNS、セキュリティグループ、EBS、EC2 インスタン スを作成します。

mijin は各 EC2 インスタンス間を DNS 名で通信を行います。ドメインは mijin.internal で固定になり、各 インスタンス名はそれぞれ以下のように、A レコード設定されます。

- 1. api1.mijin.internal
- 2. api2.mijin.internal
- 3. peer1.mijin.internal
- 4. peer2.mijin.internal
- 5. peer3.mijin.internal
- 6. peer4.mijin.internal
- 7. peer5.mijin.internal
- 8. peer6.mijin.internal
- 9. peer7.mijin.internal
- 10. peer8.mijin.internal
- 11. peer9.mijin.internal

セキュリティグループは以下、画像のように最低限の通信に使用するセキュリティグループが適用されま す。

セキュリティグループ名	説明
attach-ssm_ssh-client	このセキュリティグループは踏み台など設定することで、SSH ログインが可能になります。新規 VPC では使用しません。(既 存 VPC 用で作成した場合、既存の踏み台に割り当てるなど)
attach-node-client	ノード間通信用です。
sv-api	API ノード用です。 3000 ポート / REST アクセス用です 7900 / mijin ノード通信用です。
sv-peer	PEER ノード用です。 7900 ポート / mijin ノード通信用です。



EBS は mijin のデータ領域を保存するため、API ノードでは、root パーティション以外に 2 つの EBS、 PEER ノードでは、一つの EBS をアタッチしています。

VolumeType は `GP3` 固定であり、ディスクの暗号化はしていません。

EC2 インスタンス初回構築時のみ、cloud-init を使って、EBS を XFS フォーマットし、ディスクマウント するように設計されています。

mijin データは `block` に保存されます。API ノードでは、block` と rest で使用する `mongo` のデータを 保存しますが、 `block にあるデータがあれば、mongo データを復元することができます。



EC2 インスタンスは、mijin パッケージがインストールされたカスタム AMI を使用し起動します。UserData にて、cloud-init を動作し、初期パッケージの設定を実行します。UserData の実行結果を受け取り、設定 に失敗するとロールバックするように動作します。

- 1. パラメータで指定した Unix ユーザーの設定
- 2. ホスト名の設定
- 3. OS パッケージの更新
- 4. pip のインストール
- 5. cloudformation helper script のインストール
- 6. mijin のセットアップ
  - 1. EBS フォーマット
  - 2. mijin セットアップ (api1 はパラメータストアにデータをアップ)

また、インスタンスを作成する順序としては、以下のようになります。

- 1. Apilnstance1
- 2. ApiInstance2 PeerInstanceX 並行実行

## Apilnstance1 にて、すべてのノードに使用する設定を作成し、AWS Systems Manager パラメータスト アにデータを保存します。その他インスタンスはこのパラメータストアからデータを取得し、mijin を作 成します。

パラメーターストアに保存される内容は、以下となります。

パラメータ名	説明
/デプロイ時に指定した冠名/shares/api_node.json	API ノードが使用する公開鍵
/デプロイ時に指定した冠名/shares/generation_hash.json	mijin Catapult(v.2) のブロックチェーンのジェネ
	シスハッシュ (GenerationHash)
/デプロイ時に指定した冠名/shares/harvest_fee_sink_public_key.json	Harvest を受け取るアドレス (mijin では不要)
/デプロイ時に指定した冠名/shares/init_host_count.json	デプロイ時に作成したノード数
/デプロイ時に指定した冠名/shares/mosaic_rental_fee_sink_public_key.jso	n Mosaic レンタル費用を受け取るアドレス
/デプロイ時に指定した冠名/shares/namespace_rental_fee_sink_public_ke	y.j <b>skam</b> espace レンタル費用を受け取るアドレス
/デプロイ時に指定した冠名/shares/nemesis_addresses.json	とくに使用されてない空のアドレス (使用可能)
/デプロイ時に指定した冠名/shares/nemesis_addresses_harvesting.json	harvest などを受け取るアドレス
/デプロイ時に指定した冠名/shares/nemesis_addresses_harvesting_voting.	jsōħァイナライズの権限に使用するアドレス
/デプロイ時に指定した冠名/shares/nemesis_addresses_harvesting_vrf.jsor	・セキュリティ強化用のアドレス (ブロック生成で
	きる状態をわからなくする)
/デプロイ時に指定した冠名/shares/peer_node.json	PEER ノードが使用する公開鍵
/デプロイ時に指定した冠名/shares/rest_gateway_private_key.json	API ノードが使用する REST 用のアドレス
/デプロイ時に指定した冠名/shares/signer_private_key.json	Nemesis(Genesis) ブロックを署名するアドレス
/デプロイ時に指定した冠名/shares/new-cert/各ノード/CA/[*].pem	ノード間の通信を暗号化する SSL 証明書

### 注釈:

このパラメータストアの値はブロックチェーンの1ブロック目に作成する値として保存され、デプロイ以 降、パラメータストアから呼び出すことはありません。

そのため、セキュリティとしてデータを削除しておきたいなどがある場合、このデータを削除しても問題 ありません。

また、障害や新規に増設したいなどがある場合などは、このデータから復旧することができます。

#### **loadbalanceNestStack**

loadbalanceNestStack では、ELB(ロードバランサー)を作成します。

パラメータ UseLoadBalancer が No の場合は ELB を作成しないスタック (empty-elb) を呼び出します。 ELB は、API ノードの REST アクセスポート `3000`ポートへ分散します。 ELB の Type は NLB(ネットワークロードバランサー) を使用し、API ノードへの接続はスティッキーセッ ションにより同一セッションは一定期間、同じノードを使用する設定となっています。

## 注釈:

ELB は Network Load Balancer(NLB) のみで作成します。 NLB について理解したい場合は、以下を参照してください。 https://docs.aws.amazon.com/ja_jp/elasticloadbalancing/latest/network/introduction.html

```
TargetGroupAttributes:
 - Key: stickiness.enabled
 Value: 'true'
```

NLB は、プライベートのみで使用する内向き用の配置、インターネット経由で接続する外向き用の配置を パラメータで指定することができます。



API ノードの REST アクセスポート `3000`ポートへは以下のような条件でヘルスチェックをおこなって います。 ヘルスチェックは http://API ノード:3000/chain/info を死活監視しています。

Properties: HealthCheckIntervalSeconds: 10 # 10 秒間隔でチェックする Check at 10-second intervals. UnhealthyThresholdCount: 3 # 異常とみなす回数 Number of times considered abnormal HealthyThresholdCount: 3 # 正常とみなす回数 Number of times considered normal HealthCheckPath: /chain/info # ヘルスチェックをする URL URL for health check HealthCheckProtocol: HTTP Port: 3000 # ヘルスチェックポート health check port

### 2.2.5.3 AWS MarketPlace mijin Catapult(v.2) アーキテクチャパターンによるリカバリ戦略

AWS MarketPlace で展開する mijin Catapult(v.2) は、デプロイ時のパラメーターによって様々なアーキテ クチャパターンがあります。

本章では、アーキテクチャパターンを学び、ディザスタリカバリなどの BCP 対策によるリカバリ戦略を説 明します。

### mijin Catapult(v.2)の高可用性と耐障害性

mijin Catapult(v.2) は、全ノードに同じブロックチェーンデータを保持するため、最低でも PEER ノード 1台あれば、ブロックチェーンの更新は継続します。(分散型フォールトトレランス)

ブロックチェーンは。RDB などレコード単位でデータ保存はせず、ブロック単位で纏めてデータ保存をす る特徴があります。

そのため、障害があった状況によって、最新のブロックチェーンデータが必ずしも最新かつ正しいデータ であるとは限りません。

理由としては、デプロイ時に設定したおよその秒数でブロックチェーンのブロックデータが生成され、ブ ロック生成するノード (ハーベスティングノード)、生成したブロックを受信するノードに分かれます。

ブロックチェーンデータを作成するノードはコンセンサスアルゴリズム (Proof-of-Stake Plus) によって、 ブロックを生成するノード (ハーベスティングノード) が剪定されます。

コンセンサスアルゴリズムについて詳しく知りたい場合は、Symbol の Document を参照してください。

https://docs.symbol.dev/ja/concepts/consensus-algorithm.html

mijin にもブロック生成の仕組みによる報酬は存在しますが、プライベートブロックチェーンでは必要な いため、機能的に報酬プロセスが動いているだけの状態となります。

以下の図では、PEER3 でブロックチェーンを生成し、各ノードにブロックチェーンデータを送信する図で す。



PEER3 から生成したブロックデータを受信していない状態で PEER2 のみが残った状況で障害になった場合、最低でも1ブロック分の差分が出る可能性があります。

その際、PEER2 は PEER1 と PEER3 を認識できないため、新たなブロックチェーンデータを生成し、独立 したノードに変わります。(フォーク)

ただし、PEER1 と PEER3 がすぐに復旧した場合、正しいブロックデータを持っている PEER3 を正として ロールバックし、正常のブロックチェーンデータへとリカバリします。

API ノードが全てダウンした場合は、プログラムなどからアクセスはできませんが、PEER ノードがいる限りはブロック生成は進みます。

## マルチ AZ によるリカバリ戦略

目標復旧時間 (RTO)	ほぼ 0
目標復旧時点 (RPO)	ほぼ0

Marketplace に展開する mijin Catapult(v.2) は、マルチ AZ 環境にノードをそれぞれ配置します。 片方の AZ 側に障害が発生したとしても、分散型フォールトトレランスである mijin Catapult(v.2) はサー ビス継続することが可能です。

API ノードへの接続を継続したい場合は、Elastic Load Balance を有効にすることで、可用性が向上します。

CLD	
Internet	
	<b>\</b>
Availability Zone	Availability Zone
PEER1 PEER2	PEER3

マルチリージョンにリカバリ戦略

リージョン間バックアップ

目標復旧時間 (RTO)	1日
目標復旧時点 (RPO)	2 時間

Marketplace に展開する mijin Catapult(v.2) は、いずれかのノード一つからブロックチェーンデータをバ ックアップすることで全復旧することができます。

AWS Backup を使用することで、ノードにあるブロックチェーンデータを別リージョンに容易にバックアップすることが可能です。

バックアップ方法は、mijin Catapult(v.2) の定期的なノードのバックアップを参照してください。 バックアップするまでの復元方法については、以下を参照してください。

https://docs.aws.amazon.com/ja_jp/aws-backup/latest/devguide/restore-resource.html

	) net	
Availability Zore	Availability Zone	Region
API1	API2	AWS Backup
PEER1 PEER2	PEER3	SNAPSHOT

アクティブ/アクティブによるコンソーシアムチェーン

目標復旧時間 (RTO)	ほぼ 0
目標復旧時点 (RPO)	ほぼ 0

# 警告:

手動で、別リージョンに mijin を構築し、コンソーシアムチェーンを実現することで **0 ダウンタイム**を 実現します。

以下図で例を示していますが、構築手順は環境によって異なるため、有償サポートによる支援も可能 ですので、以下よりお問合せください。

https://mijin.io/aws_contact/

Marketplace に展開する mijin Catapult(v.2) は、世界 21 カ国のリージョンに設置することができますが、 リージョン間でノードを構築する仕組みにはなっていません。

手動でメインリージョン以外に mijin ノードを構築することで、マルチリージョンによるディザスタリカ バリ戦略を構築することができます。

API ノードも併せて設置することでコンソーシアムチェーンとなり、リージョン間でもアクティブ/アクティブ構成となります。



# 2.2.6 mijin Catapult(v.2) デプロイ後の AWS 設定

本章では、AWS MarketPlace にて提供している mijin Catapult(v.2) をデプロイ後、設定可能な方法を記載 します。

2.2.6.1 mijin Catapult(v.2) EC2 インスタンスログイン方法

本章では、AWS 上の mijin Catapult(v.2) へのノードログイン方法を説明します。

mijin Catapult(v.2) は Linux サーバ上で動いているため、Linux ログイン方法の手順になりますが、AWS ではマネージメントコンソールから容易にリモートログインできる「Session Manager」があるため、デ プロイ時に SessionManager でリモートログインできるように設定しています。 Session Manager について詳しく知りたい方は以下を参照してください。 https://docs.aws.amazon.com/ja_jp/systems-manager/latest/userguide/session-manager.html

## 注釈:

本章は、AWS MarketPlace にある mijin Catapult(v.2) をデプロイした場合の EC2 へのログイン手順の一 例です。 接続方法は、SSH など従来のリモートログイン方法でもセキュリティグループなどを変更することで接続 できます。 AWS マネージメントコンソールにログイン

AWS マネージドコンソールにて、ログインします。 https://aws.amazon.com/jp/console/

### EC2 サービスに移動

- 1. 上部「サービス」をクリックします
- 2. 表示されたメニューから「コンピューティング」をクリックします
- 3. 「EC2」をクリックします。

aws	🗰 サービス 🛛 <b>Q</b> サービス、特徴、ブロク	、お <i>よびドキュメントなどを検索</i> [オプション+S]
🔞 Reso		
💽 Ne	最近アクセスしたサービス	「□『 コンピューティング ×
Tell	お気に入り	
EC2 S	すべてのサービス	AWS App Runner
ECZク		Build and run production web applications at scale
タグ	⊜ ∆WS コスト管理	Batch
制限	Customer Enablement	すべての規模に対応する完全マネージド型のバッチ処理
▼ インス	↔ IoT	★ EC2
インス	Machine Learning	クラウド内の仮想サーバー
インス	a Quantum Technologies	EC2 Image Builder
起動う	· · · · · · · · · · · · · ·	OS イメージの構築、カスタマイズ、デプロイを自動化するマネージド型サービス
スポッ	<b>頌 エンドユーザーコンピューティ</b>	Flastic Beanstalk
Savin	ング	ウェブアプリの実行と管理
リサ- New	979 ゲーム開発	Lambda
Dedic	🖮 コンテナ	サーバーのことを考えずにコードを実行する
キャノ	回 コンピューティング	
<b>▼</b> イメ-	🖹 ストレージ	LightSan と 仮想プライベートサーバーの起動および管理
AMI I	セキュリティ、ID、およびコン プライアンス	
AMI 7	目 データベース	AWS Outposts オンプレミスで Aws のサービスを実行
🔻 Elasti		
ボリュ	ツ配信	Serverless Application Repository チーム内または公開してサーパーレスアプリケーションを構築、デプロイ、および共
スナッ	🏦 ビジネスアプリケーション	有する
ライフ	⁹⁸⁶ ブロックチェーン	
Ne	▶ メディアサービス	
▼ ネット ィ	🛄 モバイル	
セキュ	🏯 ロボット工学	
Elasti	☑ 分析	
プレ-	☆ 拡張現実 (AR) とバーチャルリ アリティ (VR)	
+-/	<ul> <li>○ 移行と転送</li> </ul>	
ネット イス	高 管理とガバナンス	

## ログインしたいインスタンスを選択し、接続

- 1. 左側メニューから「インスタンス」をクリックします
- 2. インスタンス一覧からログインしたいノードのチェックをクリックします。
- 3. 「接続」を押します。

aws   # サービス Q サー	ビス、特徴、プログ、およびドキュメントなどを検索 [オプション+S]	2 🗘  🕐 東京 🔻
🔞 Resource Groups & Tag Editor 🛛 👩	EC2 🧔 Systems Manager 👩 CloudFormation 🛛 CloudWatch 👩 Elastic Container Service 📓 RDS 💐 Cognito	🗑 CodePipeline 🛛 🗑 S3 🛛 👩 Simple Queue Servio
New EC2 Experience	インスタンス (1/5) 情報     ご 接続     イン:	スタンスの状態 ▼ アクション ▼
EC2 ダッシュボード EC2 グローバルビュー	Q Find インスタンス by attribute or tag (case-sensitive)         インスタンスの状態 = running       X         MUJIN-CATAPULT-G       X	
イベント	■ Name マ インスタンス ID インスタンス	▽ │ インスタンス ▽ │ ステータフ
タグ	□ MIJIN-CATAPULT-G-api1 i-0dba039b7d99b7008 ⊘ 実行中	⊕ Q t3.large
制限	☑ MIJIN-CATAPULT-G-peer3 i-0a496d90f9d7afed0 ② 実行中	⊕     Q         t3.large
▼ インスタンス	□ MIJIN-CATAPULT-G-peer1 i-034b22389cba46ac7 ② 実行中	⊕     Q
インスタンス New	□ MIJIN-CATAPULT-G-api2 i-Off373a80e5646585 ② 実行中	⊕ Q t3.large
インスタンスタイプ	□ MIJIN-CATAPULT-G-peer2 i-0490c7e0c938c3fe8 ② 実行中	⊕ Q t3.large
起動テンプレート		
スポットリクエスト		
Savings Plans		
リザーブドインスタンス		
New		
Dedicated Hosts		
キャパシティーの予約		
▼ イメージ		
AMI New		
AMI カタログ		

## セッションマネージャーを選択し、接続

1. 「セッションマネージャー」であることを確認します。(選択されていなければクリック)

- 2. 「接続」を押します。
- 3. コンソール画面のウィンドウが別で開きます。



### コンソール画面の操作

- 1. コンソール画面のウィンドウで 「\$」が出ていることを確認します。
- 2. mijin Catapult(v.2) を動かしている UNIX ユーザー catapult にスイッチします。

sudo su – catapult

3. mijin Catapult(v.2) が動いているかを確認します。

```
PEER ノードに接続した時 (本章では PEER ノード接続時) When connected to a PEER
node (in this chapter, when connected to a PEER node)
cd mijin-catapult-package/package/peer/catapult/
docker-compose ps
API ノードに接続した時 When connected to an API node
cd mijin-catapult-package/package/api/catapult/
docker-compose ps
```



## 注釈:

mijin Catapult(v.2) は docker 上のコンテナの一つとして動いてます。 操作には docker の知識が必要になります。 Docker の知識は、以下のドキュメントを参照してください。 https://docs.docker.jp/ 公式 (英語) https://docs.docker.com/get-started/overview/

## 2.2.6.2 mijin Catapult(v.2) のノードストレージの暗号化

本章では、AWS 上の mijin Catapult(v.2) のノードで使用するストレージの暗号化を説明します。

デプロイ時点では、各ノードがマウントする EBS ボリュームは暗号化されていません。 より安全にするため、ブロックチェーンデータや Mongo のデータを暗号化することができます。 ここでは、PEER ノード 3 を例に、mijin Catapult(v.2) を格納している EBS ボリュームを暗号化する手順 を説明します。

### ノードのブロックチェーンデータを暗号化するまでの流れ

- 1. KMS Key を作成します
- 2. ノードの一台を停止する。(製品版では1台停止しても可用性は維持されます)
- 3. 停止したノードのスナップショットを取得する
- 4.3で作成したスナップショットをコピーして暗号化したスナップショットを作成する
- 5.4 で作成した暗号化済みスナップショットから Volume を作成する。
- 6. 2 で停止した PEER ノードのブロックチェーンデータの Volume をデタッチする
- 7.2で停止した PEER ノードに5で作成した Volume をアタッチする
- 8.2 で停止した PEER ノードを起動する。

**注釈:** API ノードでも同じ流れですが、API ノードの場合は、mongo データもマウントしているため、二つのボリュームを暗号化することができます。

### KMS Key を作成

ストレージを暗号化するために、KMS を使用し、暗号鍵を作成します。 KMS による鍵作成について知りたい方は、以下を参照してください。 https://docs.aws.amazon.com/ja_jp/kms/latest/developerguide/create-keys.html







## ノードの一台を停止する

PEER ノード又は API ノードを一台停止しても、mijin Catapult(v.2) のブロックチェーンネットワークは 停止しません。

ここでは、PEER ノード 3 を例に停止します。

PEER ノード 3 にログインします。	mijin Catapult(v.2) EC2 インスタンスログイン方法 を参照し、ノードにログ インしてください。
	v sudo su - catapult catapult@peer]:-\$ catapult@peer]:-/mijin-catapult-package/package/peer/catapult/ catapult@peer]:-/mijin-catapult-package/package/peer/catapult\$ catapult@peer]:-/mijin-catapult-package/peer/catapult\$ docker-compose ps Name Command State Ports
	<pre>catapult_peer-node_l bash -c /bin/bash /scripts Up 0.0.0.0.07900-&gt;7900/tcp catapult@peer1:/mijin-catapult=package/package/peer/catapult\$ docker-compose down Stopping catapult_peer-node_l done Removing catapult_peer-node_l done Removing network catapult_default catapult@peer1:/mijin-catapult_package/package/peer/catapult\$ docker-compose ps Name Command State Ports</pre>
	<pre>catapult@peer3:-/mijin-catapult-package/package/peer/catapult\$</pre>
	<pre>catapult@peer3:-/mijin-catapult-package/package/peer/catapult\$ exit logout </pre>
	\$ sudo shutdown -h now
mijin Catapult(v.2) を停止し、EC2 インスタンス を停止します。	
sudo su - catapult	
cd mijin-catapult-package/	
→package/peer/catapult/	
docker-compose ps	
exit	
sudo shutdown -h now	

### 停止したノードのスナップショットを取得する

暗号化ボリュームを作成するには、まずは対象のボリュームのスナップショットを作成する必要がありま す。

	MUIN-CATAPULT-G-peer3	i-0a496d90f9d7afed0	○停止済み ④○ t3.large	- 79-4 + ар-г
	MUIN-CATAPULT-G-peer1	I-034b22389cba46ac7	受実行中	② 2/2 のチェックに1 アラーム + ap-r     ③ 2/2 のチェックに1 アラーム + ap-r     ④ 2/2 のチェックに1 アラーム + ap-r
	MUIN-CATAPULT-G-peer2	i-0490c7e0c938c3fe8	受美行中 @Q t3.large	② 2/2 のチェックに: アラーム + ap-r
	インスタンス: i-0a496d90f9d7	afed0 (MIJIN-CATAPULT-G-peer3)	=	© ×
	詳細 セキュリティ ネット	フーキング ストレージ ステータスチ:	エック モニタリング タグ	
	▼ ルートデバイスの詳細			
	ルートデバイス名 〇 /dev/sda1	ルートデバイスタイプ EBS	EB 無	15 最適化 効
	▼ プロックデバイス			
	Q 70ックデバイスのフィルタ-			
	жу _л —д ю	バイス名 ポリュームサイズ ア	タッチメントの アタッチ時刻	暗号化済み KMS キー ID
	vol-01bc638ad7fc8e758	lev/sda1 30 ⊘	アタッチ済み Thu Sep 08 2022 アタッチ済み Thu Sep 08 2022 Thu Sep 08 2022	17:58:23 G いいえ -
停止した PEER ノード 3 を選択し、ストレージタブをク リックし、対象の Volumeld をクリックします。 ここで、Volumeld とデバイス名 (/dev/sdf) は控えてお きます。				
	ポリューム (1/1) Q 他家		ポリュームの変更	C アクション ▲ ポリュームの作成 < 1 > の
	ポリューム ID = vol-0251ee8aef9dedf0	0 × フィルターをクリア	スナップショットの	Yf R
	☑ Name ▽ ポリュール	ID マ タイプ マ サイズ	▼ IOPS ボリュームの削除	ライフサイクルボリシーを作成
	vol-0251c	e8aef9dedf00 gp3 500 GIB	3000 ポリュームのアタッ	チ /09/08 17:58 GMT+9
			ホリュームのデタッ ポリュームの強制デ	デタッチ
			自動有効化されたり	/0 の管理
			2.2.6.B.W	
対象ボリュームをチェックをクリックし、「スナップシ				
ョットの作成」をクリックします。				
	ポリューム ID: vol-0251ee8aef9dee	f00		
	詳細 ステータスチェック	モニタリング タグ		
	詳細			
	ポリューム ID	サイズ	タイプ	ポリュームのステータス
	vol-0251ee8aef9dedf00	🗇 500 GIB	gp3	⊘ок
	ポリュームの状態 ⊘ 使用可能	10PS 3000	スループット 125	暗号化 職号化なし
	кмs ‡— ID	KMS キーエイリアス	KMS +- ARN	スナップショット
	- アペイラビリティーゾーン	- 作成日	- マルチアタッチ有効	- アタッチされたインスタンス
	ap-northeast-1c	① Thu Sep 08 2022 17:58:23 GMT+0900 (日本標準時)	いいえ	•
	Outposts ARN			
ここで、このボリュームがどのアベイラビリティゾーン	·			
に所属しているかを確認しておきます。				
	EC2 > ボリューム > v	ol-0251ee8aef9dedf00 > スナッフ	プショットの作成	
	スナップショ	ットの作成 🚃		
	ポイントインタイムスナッ	プショットを作成して、Amazon EB	S ボリューム上のデータを An	nazon S3 にバックアップします。
	詳細			
	ボリューム ID	adf00		
			_	
	説明 スナップショットの説明を	自力口		
	peer3-block-2022090	9		
	最大 255 文字。			
	暗号化 情報			
	昭 写15 なし			
	<b>夕グ 情報</b> タグは、AWS リソースに割 ング、および AWS のコスト	り当てるラベルです。各タグはキーとオフ の追跡に使用できます。	プションの値で構成されています。	タグは、リソースの検索とフィルタリ
	リソースにタグが関連的	けられていません。		
	タグを追加	n der sein sein		
	さらに 50 個のタグを追加す	- C 3, 9 o		
			生わいわり	スナップショットの作成
検索する時にわかりやすい名前を説明に記載  「マー			キャンセル	
ッフンヨットのTFRX」をフリックしまり。				

作成したスナップショットをコピーして暗号化したスナップショットを作成する

暗号化したスナップショットから暗号化ボリュームを作成するため、スナップショットをコピーする際に 暗号化を行います。

メニューのスナップショットをクリックし、作成したスナップ	● Provide Streament         X           EX 25 > 25 × 10 × 10         Q as           Provide Streament         Q as
ショットをチェックし、スナップショットをコピーします。	
	<text><section-header></section-header></text>
送信先リージョンはスナップショットを取得したノードと同 じリージョンを選択します。 このスナップショットの暗号化をチェックし、作成した KMS を指定します。 「スナップショットをコピー」をクリックします。	
	○ スナップションコビージΞ RL やはぎ わました State Collection         >           スナップショントコビージΞ RL やはぎ わました State Collection         ⑦ ① ② かめ ② アション マ スナップショントコレ ○          スナップション ○            BE/MR ▼ Q, state         ○ ② かめ ② アン         スナップショント ◎             Part State Collection (X)         フィルテーモウリア         ○ ③ アメカ ン             Name ♡ スナップショント ◎ ♡ ヴィズ         ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○
スナップショットが完了済みになっていることを確認してく ださい。	

# 暗号化済みスナップショットから Volume を作成する

暗号化したスナップショットから暗号化ボリュームを作成します。 この時点では、まだマウントされていないボリュームが作成されます。

	スナップジョット (1/2)         ⑦ 区 204回         アクション 8         スナップショット 05-01/2
	アークイプからステップショット 電気 形式時期 安定 ストップショットの開始 クグモ製紙
暗号化したスナップショットをクリックし、「スナップショッ トからボリュームを作成」をクリックします。	
	<b>ボリュームの作成 met</b> 同じアベイラビリティーゾーン内の任意の EC2 インスタンスにアタッチする Amazon EBS ポリュームを作成します。
	ポリューム設定
	スナップショット ID ① snap-0b01089ac19dfcfGe
	ボリュームタイプ 博羅 汎用 55D (gp2) ▼
	サイズ (GIB) 情報 500 最小に1 GIB、 豊大 16384 GIB。信以際数である必要があります。
	IOPS 1500 / 3000
	1 GB あたり 3 GPG めペースライン、最小 100 GPFs、3000 KPFS にパースト司配。 スループット (MB/59) 情報 該当しません
	アペイラビリティーゾーン 傷懼 ap-northeast-ic ・
	高速スナップショット復元 賃報 ① 選択されたスナップショットでは有効になっていません
	競号化 情報 EC2 インズタンスに関連付けられた EBS リソースの母号化ンリューションとして、Amazon EBS 留号化を使用します。 同2 アのはリュームを担号ルマネ
	KMS 年一 憤慨 mijin/ebs_encrypt_key
	KM5キーの説明 - 
	N/5 + / 小谷 伯 (7) KM5 キー ID
	①         8115f4c-612f-4b37-ac8b-648e30fabc50           KMS = A - AN
	648e30fabc50  ③ 贈号化されたスナップショットから作成されたボリュームは、スナップショットと同じキーを使用する × か、指定した別のキーを使用して自動的に勝号化されます。最早化されていないスナップショットから作成されたポリ、小なにの書へけまであった。 スナップショットが選択されていない場合は、ボリュームを暗号化して独自のキーを指定できます。詳細 はこちら [2]
	<b>タグ - オブション 信暇</b> タグは、AWS リソースに削り当てるラベルです。各タグはキーとオブションの値で構成されています。タグは、リソースの検索とフィルタリ ング、および AWS のコストの追踪に使用できます。
	リソースにタグが関連付けられていません。
	<b>ダクを追加</b> さらに 50 個のタグを追加できます。
	キャンセル ポリュームの作成
アベイラビリティゾーンをスナップショットを取得したノー ドと同じアベイラビリティゾーンを選択します。 作成した KMS キーを選択します。 「ボリュームの作成」をクリックします。	

## 注釈:

ここで、ノードと違うアベイラビリティゾーンを選択すると、ノードにアタッチする際にボリュームが表 示されないため、注意してください。

ノードのアベイラビリティゾーン確認は、停止したノードのスナップショットを取得する を確認してくだ さい。

アベイラビリティゾーンについて理解したい場合は、以下を参照してください。

https://docs.aws.amazon.com/ja_jp/AWSEC2/latest/UserGuide/using-regions-availability-zones. html#concepts-availability-zones

停止した PEER ノードのブロックチェーンデータの Volume をデタッチする

ノードにアタッチされている暗号化されていないボリュームを外します。

	インスタンス (1/5) 情報	○ 接続	インスタンスの状態 ▼ アクション ▼	インスタンスを起動 🔻
	Q Find インスタンス by attribute or tag (case-sensitive)			< 1 > @
	MIJIN-CATAPULT-G X フィルターをクリア	7		
	Name V		171471 × 171471 ×	ステータスチェ アラームのお
	MIJIN-CATAPULT-G-api1	i-0dba039b7d99b7008	②実行中 ④ ○ t3.large	◎ 2/2 のチェックに: アラーム
	MIJIN-CATAPULT-G-peer3	i-0a496d90f9d7afed0	○ 停止済み ④ Q t3.large	- アラーム
	MIJIN-CATAPULT-G-peer1	i-034b22389cba46ac7	⊘実行中 ④ Q t3.large	◎ 2/2 のチェックに: アラーム
	MIJIN-CATAPULT-G-apl2	I-Off373a80e5646585	受実行中	⊙ 2/2 のチェックに: アラーム
	MIJIN-CATAPULT-G-peer2	i-0490c7e0c938c3fe8	受実行中	⊙ 2/2 のチェックに: アラーム
	インスタンス: i-0a496d90f9d7afed0 (MIJIN)           評量         セキュリティ           マルートデバイスの評量           ルートデバイスの評量           パートデバイスの評量           マートデバイスの           マノロックデバイス           Q         プロックデバイスの           マリコーム ID         デバイスを 体が/Aa1           vel-01553/f60-75         ダイバクス           vel-0253/f60-75         ダイバイム	CATAPULT-G-peer3) = ストレージ ステータスチョ ルートデバイスタイブ ES 30 ○ 500 ○		③ ● ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○
再度、PEER ノード 3 からボリュームを選択します。	▼ 曇近のルートポリュームの溜き換えタスク			
	ポリューム (1/1)		C 775	/ヨン ▲ ポリュームの作成
	Q. 按索		ポリュームの変更	< 1 > 💿
	ポリューム ID = vol-0251ee8aef9dedf00 ×	7ィルターをクリア	スナップショットの作成 スナップショットのライフサイクルポリシ	一を作成
	図 Name ▽ ポリューム ID ▽	タイプ マ サイズ マ	ポリュームの削除	作成日
	vol-0251ee8aef9dedf00	gp3 500 GIB	ポリュームのアタッチ	2022/09/08 17:58 G
			ポリュームのデタッチ	
			ボリュームの強制デタッチ	
			自動有効化された I/O の管理	
			タグを管理	
	ポリューム ID: vol-0251ee8aef9dedf00			
ボリュームから「ボリュームをデタッチ」をクリック し、ポップアップした画面で OK を押します。				

# PEER ノードに暗号化済み Volume をアタッチする

ノードに暗号化済みボリュームをアタッチします

	ポリューム (1/1) C アクション A ポリュームの作成
	Q、使用 ポリュームの変更 〈 1 〉 ④ フォップションの登録
	Insp-uou/ussak isanctie × Zイルターをクリア ストックス ひのか ストック マックイブ マ サイズ マ ストックショントのライフサイクルボリシーを作成     Mame マ ボリューム ID マ タイブ マ サイズ マ オリューム Math
	ポリュームのアタッチ ポリュームの実例デタッチ
	自動有効化された (/0 の管理 タグを管理
	7.7.6.898
暗号化済みボリュームを選択し、「ボリュームのアタッチ」をクリッ	
クします。	
	EC3 \ #211/ \ \ yel 0.545774040b554523 \ #11/のでないぶ
	EC2 > ボリューム > vol-05da77d9d0b65d623 > ボリュームのアダッナ
	n ノAァンスにホリユームをアタッナし、週帯の物理ハードナイスクドライフと同じように使用します。
	基本的な詳細
	① このボリュームは暗号化されており、EBS 暗号化をサポ
	ートするインスタンスにのみアタッチできます。詳細は こちら
	ポリューム ID
	ap-northeast-1c
	インスタンス 情報
	I-0a496d90f9d7afed0 💌 🖸
	選択したポリュームと同じアペイラビリティーゾーンにあるインスタンスのみが表示されます。
	デバイス名 情報
	/ Jdev/sdf Linux 用の推奨デバイス名: ルートボリュームの場合は /dev/sda1。データボリュー xの場合は /dev/sd[F-p]。
	<ul> <li>ここで入力された(および詳細情報に表示される)デバイス名が/dev/sdfから /dev/sdpであっても、新しいLinuxカーネルによっては内部でデバイスの名前</li> </ul>
	が /dev/xvdf から /dev/xvdp に変更されることがあります。
	キャンセル ポリュームのアタッチ
インスタンスを PEER ノード 3 を選択し、控えておいたデバイス名を	
暗号化前のボリュームと同じパスを指定します。	
「ボリュームのアタッチ」をクリックします。	
	インスタンス (1/5) 物種 C 単美 インスタンスの状態 マ アクション マ イシスタンスを放け マ
	Q, Find 1>X\$>X by attribute or tag (ass-smathing) < 1 > ∅
	MUNICATAPULT.G         マ         フィルターをクリア           ■ Name         マ         インスタンス ID         インスタンス マ         インスタンス マ         ステータスチェ         アラームの状態
	□ MUR-CATAPULT-G-spl1 I-Odbi0356759507008 ②東行中 砲砲 t3 Jarge ②2/2 のチェックた: アラーム +
	Contraction Process Provide Contraction Contra
	MURICATAPULT-G-api2         LOTIS733801556585         ②東行中         ●         ③         I3.large         ③2/2 のチェックに1 アラーム +           MURICATAPULT-G-pert2         I-0490C70C9383188         ③東行中         ●         I3.large         ④2/2 のチェックに1 アラーム +
	ア・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・
	▼ ルートデバイスの詳細
	ルートがバイス名 ルートがバイスタイプ EES 環境化 ¹ /dev/sda1 EBS 無効
	▼ プロックダバイス
	Q. プロックデバイスのフィルター ボリューム10 デバイス名 ポリュームサイズ アタッチメントの アタッチ扱制 最短の体法 viext 8
	vel-01bc538ad7t08r758 /dev/kda1 30 〇 アタッチ消み Thu Sep 08 2022 17:58:23 G いいえ -
	vxi-0251ee8kef9dedt00 /dev/bdf 500 ②アタッチ済み Thu Sep 08 2022 17:58:23 G [55.). [40d] ▼ 最近のルートボリュームの置き換えタスク
	Q、タスクをフィルタリング ルートボリュームを置き換える
	タスク10 タスクの状態 脱処時刻 先了時間 タグ ====================================
	第20のルートボウュームの面を換えダスクなし
暗号化済みボリュームがアタッチされていることを確認します。	

注釈: デバイス名は、停止したノードのスナップショットを取得する にて控えていたデバイス名を指定す

る必要があります。

## 停止した PEER ノードを起動する

停止していたノードを起動し、mijin Catapult(v.2) もあわせて起動します。

	インスタンス (1/5) 無難
	Q. Find インスタンス by attribute or tag (case-sensitive)         インスタンスを得止         く 1 > ③           インスタンスを開始         インスタンスを開始
	インスタンスを休止     MUIN-CATAPULT-G-apl1
	☑ MUIN-CATAPULT-G-peer3 H-0a496d90f9d7afed0 ビア押上済み 包包 TSJarge - アラーム +
	○         MUIN-CATAPULT-G-peer1         I-034b22389cba46ac7         ○ 実行中         ④ Q         15.large         ○ 2/2 のチェックに:         アラーム +           ○         MUIN-CATAPULT-G-api2         I-0ff373a80e5646585         ○ 実行中         ④ Q         15.large         ○ 2/2 のチェックに:         アラーム +
	□ MUJIN-CATAPULT-G-peer2 I-0490c7e0c938c3fe8 ②実行中 QQ t3.large ③2/2のチェックに1 アラーム +
	=
	インスタンス: i-0a496d90f9d7afed0 (MIJIN-CATAPULT-G-peer3)  ⑥ ×
	詳細 セキュリティ ネットワーキング ストレージ ステータスチェック モニタリング タグ
	▼ ルートデバイスの詳細 ルートデバイスタイプ FBS型油化
	⑦ /dev/sda1 EBS 無効
	▼ プロックデバイス
	Q 729971(12071N9-
	ポリュームID         デバイス名         ポリュームサイズ         アタッチドントの         アタッチ時刻         障号化済み         KMS キー           いりのからびかりだめである         パー         70         ワークマイルエズス         アクッチ時刻         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000         10.000 <td< th=""></td<>
	vol-010555583/168758 /dev/sda1 30 07タッチ病か Inu Sep 08 2022 17:58:23 G いいえ - vol-0251ee8aef9dedf00 /dev/sdf 500 ②アタッチ病み Thu Sep 08 2022 17:58:23 G いいえ -
	▼ 最近のルートボリュームの置き換えタスク
	Q タスクをフィルタリング ルートポリュームを置き換える
	タスク ID         タスクの状態         開始時刻         完了時間         タグ
	最近のルートポリュームの置き換えタスクなし
インスダンス一覧から、PEERノード 3 をナエック	
し、' 1 ンスダンスの開始」をクリックします。	
PEER ノード 3 にログインします。	mijin Catapult(v.2) EC2 インスタンスログイン方法 を参照し、ノード
	にログインしてください。
	S df -h Fileyysten Size Used Avail Use¥ Mounted on /dev/room 100 8.10 210 28 /
	dertarpts 3,30 0,336 0% /dev tarpts 3,30 0,336 0% /dev/stm tarpts 7278 8000 726 1% /dev
	tepfs 5.0N 0 5.0N 04 /run/lock tepfs 3.9G 0 3.9G 04 /sys/fs/cgroup /dev/locn0 26N 26N 0100/tsnn/asszn.ssen/5656
	/dev/loop1 29H 29H 0100V/snap/amazon-ssm-agent/2012 /dev/loop3 56H 56H 0100V/snap/core/18/2560 /dev/loop2 114H 114W 0100V/snap/core/18/255
	/dev/loop5 64N 64M 0100 /snap/core20/1623 /dev/loop4 56N 56N 0100W /snap/core18/1885 /dev/loop7 71M 71N 0100W /snap/cst222
	/dev/loop6 68% 68M 0 1004 /smap/lxd/22753 /dev/nvmeln1 5000 J.7G 497G 1% /mst/mijin/blocks 8
	§ sudo su = catapult catapult@peerl=5 catapult@peerl=5:-§ cd mijin-catapult-package/package/peer/catapult/
	catapultBpec17-rijin-catapul-package/package/package/packagelse catapultBpec17-rijin-catapult-package/package/packagelse Creating network "catapult_default" with the default driver
	Treating cataput_per=node_i done cataput@peerl:/mijis-cataputpeck/peer/cataputs docker=compose ps Name Command State Ports
	catapult[peer=node] bash -c /bin/bash /scripts Up 0.0.0.017900->7900/tcp cstapult[peer]:-mijin-cstapult-package/package/peer/cstapult@docker-compose logstai=10
	per-nois 1 / 2022-0-0-0 0716514.490243 0x0000763877fe700: <debug> (inportance::StorageImportanceCalculatorFactory.cpp89) writing importances to file data/importance/uj/00000000000140.dat for height 5280 mmm.mmfh 1 / 1002.00 0x0000ff200000.000007871870000.cdebugs (uslaw.ftsubtances.b23) mmhlan.genes. The Terrestane(0) uslaw.coma.dublec</debug>
	per-mode   2012-09-0 010114/2001 600007839724700. debug> (http://bankandoge.net/) per/solaring/tops/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/formage/tambos/technologies/fo
	<pre>peer-node_1   2022-09-09 07:06:14.520410 0x00007f83877fe700: <debug> (importance::StorageImportance/calculatorFactory.cpp#9) writing importances to file data/importance/wig/00000000000000046:dat for height 5320 peer-node 1   2022-09-09 70:6514.748499 0x0007f83897fe700: <debug> (cache::StorageImportanteCalculatorFactory.cpp#9) writing importances to file opeer-node 1   2022-09-09 70:6514.748499 0x0007f83897fe700: <debug> (cache::StorageImportanteCalculatorFactory.cpp#9) writing importances to file opeer-node 1   2022-09-09 70:6514.748499 0x0007f83897fe700: <debug> (cache::StorageImportanteCalculatorFactory.cpp#9) writing importances to file opeer-node 1   2022-09-09 70:6514.74899 0x0007f83897fe700: <debug> (cache::StorageImportanteCalculatorFactory.cpp#9) writing importances to file opeer-node 1   2022-09-09 70:6514.74899 0x0007f83897fe700: <debug> (cache::StorageImportanteCalculatorFactory.cpp#9) writing importances to file opeer-node 1   2022-09-09 70:6514.74899 0x0007f83897fe700: <debug> (cache::StorageImportanteCalculatorFactory.cpp#9) writing importances to file opeer-node 1   2022-09-09 70:6514.74899 0x0007f83897fe700: <debug> (cache::StorageImportanteCalculatorFactory.cpp#9) writing importances to file opeer-node 1   2022-09-09 70:6514.74899 0x0007f83897fe700: <debug> (cache::StorageImportanteFactory.cpp#9) writing importances to file opeer-node 1   2022-09-09 70:6514.74899 0x0007f83897fe700: <debug> (cache::StorageImportanteFactory.cpp#9) writing importances to file opeer-node 1   2022-09-09 70:6514.74899 0x0007f83897fe70019 opeer-node 1   2022-09-09 70:6514.7489 0x0007f83897fe700: <debug> (cache::StorageImportanteFactory.cpp#9) writing importances to file opeer-node 1   2022-09-09 70:6514.74899 0x0007f83897fe700: <debug> (cache::StorageImportanteFactory.cpp#9) writing importances to file opeer-node 1   2022-09-09 70:6514.74899 0x0007f83897fe700: <debug> (cache::StorageImportanteFactory.cpp#9) writing importanteFactory.cpp#900000000000000000000000000000000000</debug></debug></debug></debug></debug></debug></debug></debug></debug></debug></debug></debug></debug></pre>
	alized beight 5004 dynamic fem multiplier O total transactions 34 (score = (), 588521710748214720), beight = 5343) peer-node 1 (2022-05-09 07:06:15.293560 0x0000763)8776700: <debug9 (cache::supplementalbattastorage.cpp32)="" 5320="" beight="" f<br="" last="" recalculation="" vrote="">alized beight 5004 dynamic fem multiplier O total transactions 34 (score = (0, 5886019083)274391), beight = 5343)</debug9>
	per-mode 1   2022-09-00 07106115.304846 0x00007818476700 : disruptor:ConsumerDispatcher.cpp844) completing processing of element 1 (317 block (beights 5027 - 5343) [A472CC16] from Remote Juli with size 11653 7928], last consumer in 1 elements bahind per-mode 1   2022-09-0710513.313755 0x0000718184776705 : debugy Consumers:TavBADCACConsumer.cpp843 [forwarding a new block with height 5344
	<pre>per-node     2022-09-00 07106115.350909 0x000076188/F67000 <info> (disruptor::ConsumerDispatcher.cpp844) completing processing of element 2 (1 blocks sights 5344 - 5344) [FM05CEB] empty from Remote_Push with mixe 3768), last consumer is 0 elements behind cataguit@geer1://miji.cataguit.psckage/psck/actaguit\$</info></pre>
	cstapult@sex1:/ijin-cstapult_packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packsgs/packs
	signts 314 - 3146 [JavaA27] edgty from Memory with mits 3140 July [Ast Computer is V elsements behind peer-node 1 2022-0-0-0 9706:05.568080 0x0007#01a5152700: <debug> (disruptor:Disruptor:pg843) disruptor queuing element 5 [1 blocks (heights 5346 - 346) [0000000] from Remote Push with size 3760]</debug>
	<pre>pettimosi = vii/vivos viivo viis viivo viis viivo viis viivo viis viivo vii viika viivo viika vii</pre>
	per-mode_1   2022-09-09 07:06:52.387642 0x00007#335152700: <info> (extensions::Network0tils.cppf125) accept result to local node port 7900 from 10.0.3 06: Accepted personnel   2022-09-09 07:06:52.387642 0x00007#335152700: <info> (extensions::Network0tils.cppf125) accept result to local node port 7900 from 10.0.3</info></info>
	<pre>TBC/TIGAARD/TDIAI65820633CC187ACE18AAD8729AA0 per-mode   2022-09-09 07106156.123110 0x0000763a51527001 <info> (info&gt; (info&gt; (info&gt; get):PacketSocket.cpp4850) connected to peerl.mijin.internal [10.0.2.78:7900] (00 000000000071</info></pre>
	peer-node_1 2022-09-09 07:06:56:123912 0x00007f83a5152700: <debug> (ionet::PacketSocket.cppf513) socket close triggered by destruction (000000000000) ), 7ss elapsed peer-node 1 2022-09-09 07:06:56:127166 0x00007f83a5953700: <info> (ionet::PacketSocket.cppf850) connected to peer2-milim.internal [10.0-3.206:7900] (0</info></debug>
	<pre>peer-node_1   2022-09-09 07:06:56.127767 0x00007f83a5953700: <debug> (ionet::PacketSocket.cpp#513) socket close triggered by destruction (0000000000000) , line clapsed</debug></pre>
	catapult@peerl=-/mijin-catapult-package/peer/catapult\$
ティスクかマウントされていることを確認し、mijin	
Catapult(v.2) を起動します。	
df h	
sudo su - catapult	
cd mijin-catapult-package/package/	
→peer/catapult/	
docker-compose up -d	
dockor_compose up u	
docker-compose ps	

## 2.2.6.3 mijin Catapult(v.2) の定期的なノードのバックアップ

本章では、AWS 上の mijin Catapult(v.2) のノードのデータをバックアップする手順を説明します。 ノードをバックアップすることで、万が一リージョン障害などがあった場合など、mijin Catapult(v.2) の ブロックチェーンデータから復旧することができます。

## AWS Backup サービスについて

EC2 インスタンス上で動いてる mijin Catapult(v.2) を容易にバックアップする「AWS Backup」を使用し、 バックアップを定期的に実行します。 AWS Backup について知りたい方は、以下を参照してください。 https://docs.aws.amazon.com/ja_jp/aws-backup/latest/devguide/whatisbackup.html

## バックアッププランの作成





### 注釈:

バックアップするノード (EC2 インスタンス) を複数指定してバックアップすることはできますが、 mijin Catapult(v.2) は1つのノードにあるブロックチェーンデータから全てを復旧できるため、定期的な バックアップは一つのノードだけバックアップすれば問題なく、コストを抑えることができます。 ※1 コピー先のリージョンを同一リージョンにすると、バックアップしたスナップショットからリストア する にて復旧することができます。

別リージョンにすることで、ディザスタリカバリなどの対災害対策とすることもできます。

## 2.2.6.4 mijin Catapult(v.2) 手数料ありモード有効時の残高アカウントの残高移動

本章では、AWS 上の mijin Catapult(v.2) ノードの残高を持つアカウントから別のアカウントに残高を移動 する方法を説明します。

移動したアカウントを mijin Catapult(v.2) の操作を行うアカウントとして推奨します。

#### 注釈:

ブロックチェーンにおいて、基軸通貨は必ずあり、ブロックチェーンを操作する上でトランザクション手 数料を支払う必要があります。

mijin Catapult(v.2) でも手数料ありモードを有効にすると、トランザクション作成に手数料が必要となり、 トランザクション手数料、Mosaic レンタル手数料など全てに基軸通貨の残高を持っているアカウントで 操作する必要があります。

逆に、手数料なしモードでは、残高を持ってないアカウントを使ってトランザクションを作成することが できます。

### 残高あるアカウントの確認

残高あるアカウントは、AWS Systems Manager パラメータストアにある、 nemesis_addresses_harvesting.json に保存されます。 このファイルのアカウント一覧は、各ノードにすべて紐づいています。
```
/MIJIN-CATAPULT/shares/nemesis_addresses_harvesting.json
 概要
 履歴
 タグ
 名前
 /MIJIN-CATAPULT/shares/nemesis_addresses_harvesting.json
 利用枠
 Standard
 痡粨
 String
 最終変更日
 Tue, 17 Jan 2023 06:26:05 GMT
 値
 {
 "api": [
 {
 "address": "MDBFPEAECQTM5CTDXWGAMEQQ5GRQ5ORMXFNTA4A",
 "public_key": "7D7C86B3229CC1B6551A1526DB4ADF646CD0A3DAB1C7090DBE70798FA63E9BD5",
 "private_key": "562B913CADD35D2FA18CA26B9F357966AFF6908DEFCBC6DA335355C1791D949E4"
 }.
 "address": "MB6XYV4MNZ2BI7SYEVHQF2HWYUF3CBBB4SWAQ3I",
 "public_key": "06358B47BF9A0DCB481A451E048A4880831CAF3160336A2F7555378C6107B75D",
 "private_key": "7F54E66E6F83FF6CF75BA1F5444DF9904334FB58CEEF60D03A68334543F29A2C"
 3.
 "address": "MBWDZML7UN4TW3W3OM3HYR6MYMLBUK2IZMCGMLQ",
 "public_key": "A2C27604062DE3F2ECD554E6FF19292A85D52CB5E16193CB7FF86950A305E41F",
 "private_key": "2EBA3EC82A2C2DEF7C01DD28908CBD2E346908E41C6351C9D2B149C3866270DC"
 3
 1,
 "peer": [
 "address": "MCLF2ATQK244CXTW452GENXSUNFND3A77N5K4GA",
 "public_key": "600F61AB6CBAE4E205DF13933479E8F597CE2F6E44EDA05228335CF074BEC397",
 "private_key": "CFB9E3CE97CE0A09EA0800CBB7863C6C3C34ED6BCED8FBEEDCAAAD82F783E31D"
 }.
 {
 "address": "MBUOACOIVGCYE4HTGD22KCYRQDTHDIKJ4UQEZAQ",
 "public key": "42203C105C56097EBFA73AFCA210FE64F798877C150D1A163E91AC76C84E4D05".
 "private_key": "66F286751674BEBB002321AD3098D851A68A78F7434A8323CDE3F8EF349093D7"
 3
 "address": "MAKYTI6UEHQPXT6URA7O343QAWGPGVQW2ZI4GJI",
 "public_key": "233DBC5D4F40AAEF258F6E95A4F345A5884C2066B9DB815AA1D84D2F363E4AD2",
 "private_key": "A46027A097CB264229E248F0B9DC9473F0F3CE202A84B49934CD9B9F8C27E9EE"
 }.
 "address": "MBY5AE5Z33TZK3LJKIYVAR6TOSAMBY43SSPZNEQ",
 "public_key": "F7A7BF7C36CCC292C20C8B5EF7A9D166BD64CEBC69E7CD1D1C3E0D2A890B8C39",
 "private_key": "AB30316D06C5DC8880E347B30C0ECAD55F2865876E25D97EEBC10E801D952406"
 "address": "MDRHUCI4BUGBE2UPOHW5YAU2RW4OXGF74YB2AKY".
 "public_key": "35647B4814CDC693FB9CAB8E19680977EEB1901BD91320153D957CDA31D7A9CF".
 "private_key": "99BFCBD6492131C257FBD7528B23A92FC53324B0FBB74A5DFB2C37E13B73F044"
 }.
 "address": "MB5VDICRGZUNLRDMBBIYDWVOZEJRTS2DZILZWPQ",
 "public_key": "8E5E81270C7DC9ECEB4EEA96C38559C51D73E3B52348937306E6119E82233B95",
 "private_key": "B1CBE7C231509D4346DE44A67AC234B73AA7335CE077F53FBA566ABD6B413D0E"
 3.
 "address": "MAPVOVSBZ7BVV4K6JJ337BEEIMSTRJBCD64GYNY",
 "public_key": "B2D198630DF58AAAB3AFF8DEFD3BEA1D844C7C00FA2D26777E3A219D6240CF27",
 "private_key": "4EB84C95958A5EA7319E8D603CF648F5A80249F38FE3CBBA6BEA8592B60E3773"
 1
 3
```

すべてのアカウントに残高はありますが、ここでは一番上の以下のデータを mijin-catapult-cli を使用し て確認します。

```
"address": "MDBFPEAECQTM5CTDXWGAMEQQ5GRQ5ORMXFNTA4A",
 "public_key": "7D7C86B3229CC1B6551A1526DB4ADF646CD0A3DAB1C7090DBE70798FA63E9BD5",
 "private_key": "562B913CADD35D2FA18CA26B9F357966AFF6908DEFCBC6DA33535C1791D949E4"
},
```

#### 準備 nodejs 及び yarn をインストール

{

mijin-catapult-tools を使用するため nodejs をインストールします。nodejs は NodeSource を利用して インストールします。

```
$ curl -fsSL https://deb.nodesource.com/setup_14.x | sudo -E bash - && sudo apt-get
install -y nodejs
$ node --version
$ sudo npm install -g yarn
```

#### mijin-catapult-tools のインストール

yarn を使用し、mijin-catapult-tools をインストールします。

\$ yarn global add @tech-bureau/mijin-catapult-tools \$ echo 'export PATH="\$HOME/.yarn/bin:\$PATH"' >> ~/.bashrc && source ~/.bashrc

#### 残高アカウントのインポート

残高アカウントをインポートします。

-u では、CloudFormation の出力にある mijinLBENdpoint 又は mijinEndpoint の URL を指定します。 -p では残高アカウントの private_key を指定します。

```
$ mijin-catapult-tools account generate -w aws.json -u http://xxxxxxxxxxxxxxxxxxxxxxxxxxxxx
→elb.ap-northeast-1.amazonaws.com:3000 -s -p
562B913CADD35D2FA18CA26B9F357966AFF6908DEFCBC6DA33535C1791D949E4
2023-01-17T07:35:12.266Z [info] : mijin URL: http://xxxxxxxxxxxxxxxxxxx.elb.ap-
→northeast-1.amazonaws.com:3000
2023-01-17T07:35:12.268Z [info] : Network: 96
2023-01-17T07:35:12.268Z [info] : Mosaic Currency Id: 769E11974E2CAD76
2023-01-17T07:35:12.269Z [info] : Mosaic Harvest Id: 792022E7945425E4
2023-01-17T07:35:12.269Z [info] : Start Account Generate...
2023-01-17T07:35:12.343Z [info] : Write Config File: aws.json
2023-01-17T07:35:12.345Z [info] : New Account: {
 "url": "http://xxxxxxxxxxxxxxxxx.elb.ap-northeast-1.amazonaws.com:3000",
 "workAccount": {
 "publicKey": "E28BF2A27FE64DF392CBF6D6883BE4858CF26790B4EABC3BCA8E08854BC6A9BF",
 "privateKey": "C3BE65EB9055405ED8CCB7AD568D1368067174F969E9321C0AB4379A7565C9C2",
 "address": "MB2ZQXQQQHOVYU4GX2TKNNJK6XLZPIZV6LK62MY"
```

},
"balanceAccount": {
"publicKey": "7D7C86B3229CC1B6551A1526DB4ADF646CD0A3DAB1C7090DBE70798FA63E9BD5",
"privateKey": "562B913CADD35D2FA18CA26B9F357966AFF6908DEFCBC6DA33535C1791D949E4",
"address": "MDBFPEAECQTM5CTDXWGAMEQQ5GRQ5ORMXFNTA4A"
}.
"mainAccount": {
"publicKey": "DAF95081E2D816062108424CF2404B9C3B7C4C7CD1DF6E1446158CC2A2D9B29B",
"privateKey": "ACF704D53457DF418E1784BACB6D0977B2626BC2847506C7D91B39CDD0515F0F",
"address": "MD4J2ZVP2AW3BCH6UKZOJNNBGS35DBKTNS4KGVI"
}.
"kevlink": {
"vrf": {
"publicKey": "E3D05474D23B57EEFCF953EB7A1AF7A44F9BA338C83900AB0A72927D933CD56A",
"privateKev": "D0F48B1926ECA6C32C1D3A61AE25A7A483EF47B6DE96B9B4E4970773F904CB73".
"address": "MBR5N.IKKF66GVIOFOKC.IOAOTR5KWFE5AKWTVI.OO"
}.
"voting": {
"publicKey": "8A615FD6E66CBCB6361FAE0156CB6E22E0932F7157F04849551FCAE9CC6E494F".
"privateKev": "1615F3484a2123131a624FA065E4B17a0CE916FF6D5A772501FBC83876B3B912".
"address": "MDWPSE2HLW4HFFTLSECCIMWOPF6IKFPOAT76V3Y"
}
}.
"test1Account": {
"publicKey": "889E1705185A2138E4408D70C28A015536E05A69185392E8C683BB39A0BDB951".
"privateKev": "8636D903270275A3A6459B41E73E8E7365A6A55ED2437AB25E3321230ED64C35"
"address" "MBRU2UKYC5C7J6MNOU7F3KYXFVMSTUKUZTPWFHI"
},
"test2Account": {
"publicKey": "24323D2D3594BF0A1993E018571EBD1175BD4B461EF3D15FE4EB09FADAB95834".
"privateKey": "2A4F1B1E98BF4F91F6FE6D6F90844D9FCA78BCF198028AD8778CC85C409F6B5E".
"address": "MAWOT4JSX30YBMNGRM47ZODDUHCJ3LLOKKO6RKA"
}

アカウントの情報を確認します。

"currency": true となっている Mosaic ld **769E11974E2CAD76** が基軸通貨になります。 "harvest": true となっている通貨は、ブロック生成する権利があるノードでは所有が必要になります。

**警告:**基軸通貨の Mosaic ld はデプロイ時に作成されるため、同じ lD でないことに注意してください。

```
$ mijin-catapult-tools account info -r aws.json -t balance
2023-01-17T07:36:16.432Z [info] : mijin URL: http://xxxxxxxxxxxxxxxxx.elb.ap-
onortheast-1.amazonaws.com:3000
2023-01-17T07:36:16.433Z [info] : Network: 96
2023-01-17T07:36:16.433Z [info] : Mosaic Currency Id: 769E11974E2CAD76
2023-01-17T07:36:16.434Z [info] : Mosaic Harvest Id: 792022E7945425E4
2023-01-17T07:36:16.434Z [info] : Start Account Info
2023-01-17T07:36:16.603Z [info] : balance Account: {
 "publicKey": "7D7C86B3229CC1B6551A1526DB4ADF646CD0A3DAB1C7090DBE70798FA63E9BD5",
```

```
"address": "MDBFPEAECQTM5CTDXWGAMEQQ5GRQ5ORMXFNTA4A",
"mosaics": [
 {
 "id": "769E11974E2CAD76",
 "amount": "1799799999600000",
 "currency": true,
 "harvest": false
 },
 {
 "id": "792022E7945425E4",
 "amount": "3000000",
 "currency": false,
 "harvest": true
 }
],
"keylink": {
 "vrf": {
 "publicKey": "9CD207F9A6DE6D485D350C29B749590251924A29C0EFD8E38DDE24866D71F160"
 },
 "voting": {
 "publicKey": "E3822AA0720F610847E4BE2B740F8FFF9130BEC4E9140BA845150DC2D591D86D",
 "startEpoch": 1,
 "endEpoch": 26280
 }
}
```

#### 新規アカウントの作成

残高がないアカウントを新規作成します。 アドレス MDJMNVU47CWHTZBMX7B6M6WWT5NEEY4GTG66GLQ に後ほど残高を移動します。

```
$ mijin-catapult-tools account generate -r aws.json
2023-01-17T07:38:41.738Z [info] : mijin URL: http://xxxxxxxxxxxxxx.elb.ap-
onortheast-1.amazonaws.com:3000
2023-01-17T07:38:41.740Z [info] : Network: 96
2023-01-17T07:38:41.740Z [info] : Mosaic Currency Id: 769E11974E2CAD76
2023-01-17T07:38:41.740Z [info] : Mosaic Harvest Id: 792022E7945425E4
2023-01-17T07:38:41.740Z [info] : Start Account Generate...
2023-01-17T07:38:41.768Z [info] : New Account: {
 "publicKey": "7437EB45A39AF335F08CABD203503632115CA1793902F5106BC03963C96AEE4F",
 "privateKey": "708AB4973F37B89195340AEA7EBD733ED16AE51B99EB648E7A3885869CBAF3C9",
 "address": "MDJMNVU47CWHTZBMX7B6M6WWT5NEEY4GTG66GLQ"
```

アカウントの情報をノードに問い合わせます。

このアカウントは、ノードに残高の記録がないため、存在しないエラーになっていることを確認してくだ さい。 \$ mijin-catapult-tools account info -r aws.json -t other -a
MDJMNVU47CWHTZBMX7B6M6WWT5NEEY4GTG66GLQ
2023-01-17T07:39:44.832Z [info] : mijin URL: http://xxxxxxxxxxxxx.elb.aponortheast-1.amazonaws.com:3000
2023-01-17T07:39:44.834Z [info] : Network: 96
2023-01-17T07:39:44.834Z [info] : Mosaic Currency Id: 769E11974E2CAD76
2023-01-17T07:39:44.834Z [info] : Mosaic Harvest Id: 792022E7945425E4
2023-01-17T07:39:44.834Z [info] : Start Account Info
2023-01-17T07:39:45.061Z [error] : Address Not Found

残高の移動

残高がある balance アカウントから新規アカウント (MDJMNVU47CWHTZBMX7B6M6WWT5NEEY4GTG66GLQ) に残高を転送します。 ここでは 10 万 cat.currency 転送してみます。

残高アカウント (mijin-have-currency-account) から転送トランザクションをアナウンスします。

\$ mijin-catapult-tools transaction transfer -r aws.json -f balance -d						
MDJMNVU47CWHTZBMX7B6M6WWT5NEEY4GTG66GLQ -a 100000						
2023-01-17T07:41:40.559Z [info] : mijin URL: http://xxxxxxxxxxxxxxxxxxxx.elb.ap-						
⇔northeast-1.amazonaws.com:3000						
2023-01-17T07:41:40.562Z [info] : Network: 96						
2023-01-17T07:41:40.562Z [info] : Start Transfer Account						
2023-01-17T07:41:40.585Z [info] : From Account Address:						
MDBFPEAECQTM5CTDXWGAMEQQ5GRQ5ORMXFNTA4A						
2023-01-17T07:41:40.585Z [info] : Dest Account Address:						
MDJMNVU47CWHTZBMX7B6M6WWT5NEEY4GTG66GLQ						
2023-01-17T07:41:40.585Z [info] : Currecny Amount: 10000000000						
2023-01-17T07:41:40.594Z [info] : Start Transfer Transaction						
2023-01-17T07:41:55.775Z [info] : End Transfer Transaction						
2023-01-17T07:41:55.775Z [info] : http://xxxxxxxxxxxxxxxxx.elb.ap-northeast-1.						
⇔amazonaws.com:3000/transactionStatus/						
⇔EC5FE12DBEFD1DF7DDE2D49287EC4DA1649546BB1EC43DE75641D5D4A7BEE770						
2023-01-17T07:41:55.775Z [info] : http://xxxxxxxxxxxxxxxxx.elb.ap-northeast-1.						
⇔amazonaws.com:3000/transactions/confirmed/						
Get Set Set Set Set Set Set Set Set Set S						

新規アカウントに残高があることを確認します。 先ほどエラーになっていたアカウント情報が mijin Catapult(v.2) に認識され、残高を持っていることを確 認できます。

```
$ mijin-catapult-tools account info -r aws.json -t other -a
MDJMNVU47CWHTZBMX7B6M6WWT5NEEY4GTG66GLQ
2023-01-17T07:42:26.802Z [info] : mijin URL: http://xxxxxxxxxxxx.elb.ap-
-northeast-1.amazonaws.com:3000
2023-01-17T07:42:26.803Z [info] : Network: 96
2023-01-17T07:42:26.804Z [info] : Mosaic Currency Id: 769E11974E2CAD76
```

```
2023-01-17T07:42:26.804Z [info] : Mosaic Harvest Id: 792022E7945425E4
2023-01-17T07:42:26.804Z [info] : Start Account Info
2023-01-17T07:42:26.927Z [info] : get Account: {
 "address": "MDJMNVU47CWHTZBMX7B6M6WWT5NEEY4GTG66GLQ",
 "mosaics": [
 {
 "id": "769E11974E2CAD76",
 "amount": "10000000000",
 "currency": true,
 "harvest": false
 }
],
 "keylink": {
 "vrf": {
 "publicKey": ""
 },
 "voting": {
 "publicKey": "",
 "startEpoch": "",
 "endEpoch": ""
 }
 }
```

#### 2.2.6.5 mijin Catapult(v.2) ノードの投票権ファイルの更新方法

本章では、AWS 上の mijin Catapult(v.2) ノードにあるノード投票権の期限が切れた場合の対応方法を記載 します。

ノードの投票権の期限は、ブロック生成間隔によって変動しますが、約 547 日 ~3285 日となっています。

# 警告:

投票権ファイルの期限が切れると、ファイナライズブロックが停止します。 そのため、ファイナライズブロックを使用している場合は必ず更新する必要があります。

# 投票権ノードの有効期限計算方法

一つの投票権ファイルの期限は以下の計算式で求めることができます。

**警告:** blockGenerationTargetTime(ブロック生成間隔) は例えば 15s を設定しても一定時間でブロック 生成するわけではないため、目安として値となることに注意してください。

(VotingSetGroup * maxVotingKeyLifetime) / ( 60 / blockGenerationTargetTime * 60 * 24)

VotingSetGroup 及び maxVotingKeyLifetime は固定値となるため、blockGenerationTargetTime の値に よって 有効期限が変動します。

```
blockGenerationTargetTime 10s
(180 * 26280) / (60 /10 * 60 * 24) = 547 days
blockGenerationTargetTime 15s
(180 * 26280) / (60 /15 * 60 * 24) = 821 days
blockGenerationTargetTime 60s
(180 * 26280) / (60 /60 * 60 * 24) = 3285 days
```

# AWS マネージメントコンソールにログイン

AWS マネージドコンソールにて、ログインします。

https://aws.amazon.com/jp/console/

# EC2 サービスに移動

1. 上部「サービス」をクリックします
 2. 表示されたメニューから「コンピューティング」をクリックします
 3.「EC2」をクリックします。

aws	🗰 サービス 🛛 <b>Q</b> サービス、特徴、ブロク	サービス <b>Q</b> サービス、特徴、ブログ およびドキュメントなどを検索 [オプション+S]				
🔞 Reso						
💽 Ne	最近アクセスしたサービス	「□『 コンピューティング ×				
Tell	お気に入り					
EC2 5	すべてのサービス	AWS App Runner				
ECZ ジ イベン		Build and run production web applications at scale				
タグ	🗟 AWS コスト管理	Batch				
制限	Customer Enablement     Output     Description     Sector	すべての規模に対応する完全マネージド型のバッチ処理				
▼ インス		★ EC2				
インス	Machine Learning	クラウド内の仮想サーバー				
インス	Quantum Technologies	EC2 Image Builder				
起動う	🔄 アプリケーション統合	os イメージの構築、カスタマイズ、デプロイを自動化するマネージド型サービス				
スポッ	<b>甌 エンドユーザーコンピューティ</b>	Elastic Beanstalk				
Savin	ング	ウェブアプリの実行と管理				
New	デーム開発     ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	Lambda				
Dedic		サーバーのことを考えずにコードを実行する				
キャノ	□ コンピューティング	Lightsail 12				
<b>▼</b> イメ-		仮想プライベートサーバーの起動および管理				
AMI I	⋓ セキュリティ、ID、およひコン プライアンス	AW/S Outposts				
AMI 🤈	ᇦ データベース	オンプレミスで AWS のサービスを実行				
▼ Elasti	佘 ネットワーキングとコンテン	Convertees Application Depository				
ボリュ	ツ配信	Serveness Application Repository チーム内または公開してサーバーレスアプリケーションを構築、デプロイ、および共				
スナッ	🏦 ビジネスアプリケーション	नि <b>ग</b> रु				
ライフ — Ne	BEG ブロックチェーン					
	▷ メディアサービス					
▼ イツト イ	□ モバイル					
セキュ	ふ ロボット工学					
Elasti	☑ 分析					
プレー	⇔ 拡張現実 (AR) とバーチャルリ アリティ (VR)					
+/						
イス	画 管理とガバナンス					

# ログインしたいインスタンスを選択し、接続

1. 左側メニューから「インスタンス」をクリックします

- 2. インスタンス一覧からログインしたいノードのチェックをクリックします。
- 3.「接続」を押します。



# セッションマネージャーを選択し、接続

1. 「セッションマネージャー」であることを確認します。(選択されていなければクリック)

- 2.「接続」を押します。
- 3. コンソール画面のウィンドウが別で開きます。

aws	サービス	<b>Q</b> <i>サービス、</i>	特徴、ブログ、およ	びドキュメントなる	どを検索 [オフ	プション+S]		
🔞 Resou	urce Groups & Tag E	Editor 👩 EC2	🔉 Systems Manager	😰 CloudFormation	CloudWatch	📀 Elastic Container Serv	vice 🔯 RDS	🛃 Cog
=	EC2 〉 インス	<b>、</b> タンス 〉 i−0a4	96d90f9d7afed0	〉 インスタンスに	接続			
	<b>インスタ</b> これらのオプ	<b>ンスに接続</b> ションのいずれかを	<b>情報</b> 使用してインスタンス	i-0a496d90f9d7afed0	(MIJIN-CATAPULT-G	i-peer3) に接続する		
	EC2 Inst	ance Connect	セッションマネ	ージャー 551	+ クライアント	EC2 シリアルコンソ	<b>ー</b> ル	
	Session N	Manager の使	用:					
	• SSH +-	-、または踏み台フ	ホストなしでインス	タンスに接続します	0			
	<ul> <li>セッショ</li> </ul>	ンは AWS Key Ma	anagement Service	キーを使用してセキ	キュア化されている	ます。		
	<ul> <li>セッショ</li> </ul>	ンのコマンドと言	羊細は、Amazon S3	バケットまたは Cl	oudWatch Logs	ロググループに記録でき	ます。	
	Session	Manager の設定へ	ページでセッション	を設定します。				
						キャンセル	接続	

#### コンソール画面の操作

```
1. コンソール画面のウィンドウで 「$」が出ていることを確認します。
```

```
$
```

mijin-catapult-tools のインストール

1. nodejs がインストールされてない場合、インストールします。

\$ curl -fsSL https://deb.nodesource.com/setup_14.x | sudo -E bash - && sudo
apt-get install -y nodejs

2. catapult ユーザーにスイッチし、mijin-catapult-tools をインストールします。

#### 現在の投票権ファイルの状況確認

公開鍵を控えます。

ここでは、**402B6ECE0D1CF99A7F07B832477048C56F213A4F54ED4AEB35AE829507FBC4A6**です。 この投票権ファイルは1から26280までの有効期限となっていることが確認できます。

# 注釈:

```
finalizationEpoch が 26280 になる前に更新する必要があるということに注意してください。
$ curl -Ss http://localhost:3000/chain/info | jq -r
{
 "scoreHigh": "0",
 "scoreLow": "16875391960469924",
 "height": "310",
 "latestFinalizedBlock": {
 "finalizationEpoch": 3,
 "finalizationPoint": 7,
 "height": "296",
 "hash": "AC19CA6C89F87F70470BD84649A31E4FA0E0C5DD71A55E9ADAE25C1AED47882F"
 }
}
```

# 投票権ファイルの更新

上記で取得した公開鍵から秘密鍵を検索し、投票権ファイルを作成します。 連番ファイル private_key_tree2.dat が作成されます。

```
$ mijin-catapult-tools votingkey update \
 -u http://localhost:3000 \
 -d mijin-catapult-package/package/api/catapult/userconfig/resources/
 votingkey \
 -p $(cat /mnt/mijin/shares/nemesis_addresses_harvesting_voting.json | jq -
 ir '.[]|.[]|select(.public_key ==
 "402B6ECE0D1CF99A7F07B832477048C56F213A4F54ED4AEB35AE829507FBC4A6")|.private_
 ikey')
2023-01-26T06:06:16.472Z [info] : Start Voting Key Update...
2023-01-26T06:06:16.476Z [info] : votingSetGroup: 160
2023-01-26T06:06:16.476Z [info] : votingMaxEpoch: 26280
2023-01-26T06:06:16.477Z [info] : votingStartEpoch: 26281
```

```
2023-01-26T06:06:16.477Z [info] : votingEndEpoch: 52560
2023-01-26T06:06:16.477Z [info] : blockGenerationTargetTime: 15
2023-01-26T06:06:42.716Z [info] : Voting Key file Create: SUCCESS mijin-

-catapult-package/package/api/catapult/userconfig/resources/votingkey/

-private_key_tree2.dat
```

#### 再度、投票権ファイルの状況確認

同じ公開鍵で作成された投票権ファイル (private_key_tree2.dat) が確認できていれば更新ファイルの作成は完了です。

次回は finalizationEpoch が 52560 になる前に、更新する必要があります。

#### 2.2.6.6 [アーカイブ] mijin Catapult(v.2) 手数料ありモード有効時の残高アカウントの残高移動

本章では、AWS 上の mijin Catapult(v.2) ノードの残高を持つアカウントから別のアカウントに残高を移動 する方法を説明します。

移動したアカウントを mijin Catapult(v.2) の操作を行うアカウントとして推奨します。

#### 警告:

symbol-cli は アーカイブ化 されたため、symbol-cli は使用できない可能性があります。 1.0.3.4 以降は、mijin Catapult(v.2) 手数料ありモード有効時の残高アカウントの残高移動 を参照して ください。

#### 注釈:

ブロックチェーンにおいて、基軸通貨は必ずあり、ブロックチェーンを操作する上でトランザクション手 数料を支払う必要があります。

mijin Catapult(v.2) でも手数料ありモードを有効にすると、トランザクション作成に手数料が必要となり、 トランザクション手数料、Mosaic レンタル手数料など全てに基軸通貨の残高を持っているアカウントで 操作する必要があります。 逆に、手数料なしモードでは、残高を持ってないアカウントを使ってトランザクションを作成することが

できます。

### 残高あるアカウントの確認

残高あるアカウントは、AWS Systems Manager パラメータストアにある、 nemesis_addresses_harvesting.json に保存されます。 このファイルのアカウント一覧は、各ノードにすべて紐づいています。

```
/MIJIN-CATAPULT/shares/nemesis_addresses_harvesting.json
 概要
 履歴
 タグ
 名前
 /MIJIN-CATAPULT/shares/nemesis_addresses_harvesting.json
 利用枠
 Standard
 痡粨
 String
 最終変更日
 Tue, 17 Jan 2023 06:26:05 GMT
 値
 {
 "api": [
 {
 "address": "MDBFPEAECQTM5CTDXWGAMEQQ5GRQ5ORMXFNTA4A",
 "public_key": "7D7C86B3229CC1B6551A1526DB4ADF646CD0A3DAB1C7090DBE70798FA63E9BD5",
 "private_key": "562B913CADD35D2FA18CA26B9F357966AFF6908DEFCBC6DA335355C1791D949E4"
 }.
 "address": "MB6XYV4MNZ2BI7SYEVHQF2HWYUF3CBBB4SWAQ3I",
 "public_key": "06358B47BF9A0DCB481A451E048A4880831CAF3160336A2F7555378C6107B75D",
 "private_key": "7F54E66E6F83FF6CF75BA1F5444DF9904334FB58CEEF60D03A68334543F29A2C"
 3.
 "address": "MBWDZML7UN4TW3W3OM3HYR6MYMLBUK2IZMCGMLQ",
 "public_key": "A2C27604062DE3F2ECD554E6FF19292A85D52CB5E16193CB7FF86950A305E41F",
 "private_key": "2EBA3EC82A2C2DEF7C01DD28908CBD2E346908E41C6351C9D2B149C3866270DC"
 3
 1,
 "peer": [
 "address": "MCLF2ATQK244CXTW452GENXSUNFND3A77N5K4GA",
 "public_key": "600F61AB6CBAE4E205DF13933479E8F597CE2F6E44EDA05228335CF074BEC397",
 "private_key": "CFB9E3CE97CE0A09EA0800CBB7863C6C3C34ED6BCED8FBEEDCAAAD82F783E31D"
 }.
 {
 "address": "MBUOACOIVGCYE4HTGD22KCYRQDTHDIKJ4UQEZAQ",
 "public key": "42203C105C56097EBFA73AFCA210FE64F798877C150D1A163E91AC76C84E4D05".
 "private_key": "66F286751674BEBB002321AD3098D851A68A78F7434A8323CDE3F8EF349093D7"
 3
 "address": "MAKYTI6UEHQPXT6URA7O343QAWGPGVQW2ZI4GJI",
 "public_key": "233DBC5D4F40AAEF258F6E95A4F345A5884C2066B9DB815AA1D84D2F363E4AD2",
 "private_key": "A46027A097CB264229E248F0B9DC9473F0F3CE202A84B49934CD9B9F8C27E9EE"
 }.
 "address": "MBY5AE5Z33TZK3LJKIYVAR6TOSAMBY43SSPZNEQ",
 "public_key": "F7A7BF7C36CCC292C20C8B5EF7A9D166BD64CEBC69E7CD1D1C3E0D2A890B8C39",
 "private_key": "AB30316D06C5DC8880E347B30C0ECAD55F2865876E25D97EEBC10E801D952406"
 "address": "MDRHUCI4BUGBE2UPOHW5YAU2RW4OXGF74YB2AKY".
 "public_key": "35647B4814CDC693FB9CAB8E19680977EEB1901BD91320153D957CDA31D7A9CF".
 "private_key": "99BFCBD6492131C257FBD7528B23A92FC53324B0FBB74A5DFB2C37E13B73F044"
 }.
 "address": "MB5VDICRGZUNLRDMBBIYDWVOZEJRTS2DZILZWPQ",
 "public_key": "8E5E81270C7DC9ECEB4EEA96C38559C51D73E3B52348937306E6119E82233B95",
 "private_key": "B1CBE7C231509D4346DE44A67AC234B73AA7335CE077F53FBA566ABD6B413D0E"
 3.
 "address": "MAPVOVSBZ7BVV4K6JJ337BEEIMSTRJBCD64GYNY",
 "public_key": "B2D198630DF58AAAB3AFF8DEFD3BEA1D844C7C00FA2D26777E3A219D6240CF27",
 "private_key": "4EB84C95958A5EA7319E8D603CF648F5A80249F38FE3CBBA6BEA8592B60E3773"
 1
 3
```

すべてのアカウントに残高はありますが、ここでは一番上の以下のデータを symbol-cli を使用して確認し ます。

```
"address": "MAL4SPKWUI3WGSNOWSDA3KKIBJG7QHMCXD7GZVA",
"public_key": "FDA90ACB0B4DA564FBA3D9D3A3E67A7146A77D2F5C246BC67AC044AAD578E161",
"private_key": "F36139408F597D2F0DA0C5E3CB1162E3D80EFEF188E21089284F57723676CC5C"
},
```

symbol-cli のインストール

npm より symbol-cli をインストールします。

```
$ sudo npm i -g symbol-cli@1.0.0
/usr/local/bin/symbol-cli -> /usr/local/lib/node_modules/symbol-cli/bin/symbol-cli
+ symbol-cli@1.0.0
updated 1 package in 8.724s
```

#### 残高アカウントのインポート

残高アカウントをインポートします。

項目	説明	値
Select the network type	マットロークな地会しててさい	MIJIN/MIJIN_TEST
	ネットワークを指定して下さい。 構築時に指定した CatapultNetwork の 値	
Enter the Symbol node URL.	Cloudformation Stack 内 に あ る Outouts タ ブ の mijinLBENdpoint または mijinEndpoint の URL を指定し ます。	<http: xxxxxx:300=""></http:>
Enter a profile name	アカウントを呼び出すプロファイル名を 指定します。	任意
Enter your wallet password	アカウントのパスワードを指定します	任意
Do you want to set the account as the default profile?	このアカウントを Default で使うかどう かを指定します。	任意
Select an import type	再度保存するためのインポート方法を指 定します。	PrivateKey
Enter your account private key	<b>nemesis_addresses_harvesting.json</b> の private_key を指定します。	任意 ここでは、F3613 で始まる値

\$ symbol-cli profile import ✓ Select the network type: > MIJIN ✓ Enter the Symbol node URL. (Example: http://localhost:3000): … http:// →xxxxxxxxxxxxxxx.elb.ap-northeast-1.amazonaws.com:3000 ✓ Enter a profile name: … mijin-have-currency-account ✓ Enter your wallet password: … ******** ✓ Do you want to set the account as the default profile? … no ✓ Select an import type: > PrivateKey

✓ Enter your account private key: … ★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★						
Account						
Property	Property Value					
Address	MAL4SP-KWUI3W-GSNOWS-DA3KKI-BJG7QH-MCXD7G-ZVA					
Public Key   FDA90ACB0B4DA564FBA3D9D3A3E67A7146A77D2F5C246BC67AC044AAD578E161						
Private Key	Private Key   F36139408F597D2F0DA0C5E3CB1162E3D80EFEF188E21089284F57723676CC5C					
Password Test1234						
SUCCESS Stored mijin-have-currency-account profile						

#### アカウントの情報を確認します。

Balance Information にある Mosaic Id の Amount の数値が高い **1D8350FA8D4830FA** が基軸通貨になります。

# **警告:**基軸通貨の Mosaic ld はデプロイ時に作成されるため、同じ ID でないことに注意してください。

<pre>\$ symbol-cli account infoprofile mijin-have-currency-account Processing Account Information</pre>						
Property	Value					
Address	MAL4SP-KWUI3W-GS	NOWS-DA3KKI-BJG7QH-	-MCXD7G-ZVA			
Address Height	1					
Public Key	FDA90ACB0B4DA564FBA3D9D3A3E67A7146A77D2F5C246BC67AC044AAD578E161         iht         1         2850000         iht         2600					
Public Key Height						
Importance						
Importance Height						
Balance Information						
Mosaic Id	Relative Amount   Absolute Amount   Expiration Height					
1D8350FA8D4830FA	1,799,799,999.6	1799799999600000	Never			
01964E14621F06F6	3,000	300000	Never			

# 新規アカウントの作成

# 残高がないアカウントを新規作成します。 アドレス MADIFG-N27CKA-6DY42J-UMEFJA-7OKXLO-NXLAEQ-XII に後ほど残高を移動します。

項目	説明	值
Select the network type		MIJIN/MIJIN_TEST
	ネットワークを指定して下さい。 構築時に指定した CatapultNetwork の 値	
Do you want to save the account?	このアカウントを保存します。	yes
Select an import type	再度保存するためのインポート方法を指 定します。	PrivateKey
Enter the Symbol node URL.	Cloudformation Stack 内 に あ る Outouts タ ブ の mijinLBENdpoint または mijinEndpoint の URL を指定し ます。	<http: xxxxxx:300=""></http:>
Enter a profile name	アカウントを呼び出すプロファイル名を 指定します。	任意
Enter your wallet password	アカウントのパスワードを指定します	任意

\$ symbol-cli account generate

- ✓ Select the network type: → MIJIN
- $\checkmark$  Do you want to save the account?  $\cdots$  yes
- ✓ Select an import type: > PrivateKey
- ✓ Enter the Symbol node URL. (Example: http://localhost:3000): … http://
- ↔xxxxxxxxxxxxxxxxx.elb.ap-northeast-1.amazonaws.com:3000
- ✓ Enter a profile name: … mijin-no-currency-account
- ✓ Enter your wallet password: … *******
- $\checkmark$  Do you want to set the account as the default profile? … no

Account

Property	Value
Address MADIFG-N27CKA-6DY42J-UMEFJA-7OKXLO-NXLAEQ-XII	
Public Key	B86CDD63C3BA820C4659CF7FC3D53DA035CF8370AC3E0DBF025BEE691AED7DFA
Private Key	E911E779671BD33B26A9D424DB331A36BDD497BA62D91B27ADAA4B1350A52D43
Password	Test1234

SUCCESS Stored mijin-no-currency-account profile

#### アカウントの情報をノードに問い合わせます。

このアカウントは、ノードに残高の記録がないため、エラーになっていることを確認してください。

#### 残高の移動

残高アカウント (mijin-have-currency-account) から 新規アカウント (mijin-no-currency-account) に残 高を転送します。

ここでは 10万 cat.currency 転送してみます。

残高アカウント (mijin-have-currency-account) から転送トランザクションをアナウンスします。

項目	説明	値	
Enter your wallet password	設定したパスワードを指定してください	任意	
Mosaics to transfer in the format (mosaicId(hex) @aliasName)::absoluteA	基軸通貨 (cat.currency) を 10 万送りま motunt基軸通貨は可分性が 6 のため 6 桁小 数点があるため、0 を 6 足します。	3BF3AF8B22CB53D8::1	
Enter the recipient address or @alias	新規アカウントの転送先アドレスを指定 します	MADIFGN27CKA6DY42JUMEFJA7OKXLONXLA	4EQXI
Enter a message	転送トランザクションにメッセージを追   加できます	任意	
Enter the maximum fee (absolute amount)	トランザクション手数料を指定します。 これは手数料モードで変わります。 手数料あり 20000 程度 (0.2cat.currency) 手数料なし 0	0	
Select the transaction announce mode	トランザクションをアナウンスする方法 を指定します。	normal	

\$ symbol-cli transaction transfer --profile mijin-have-currency-account
</ Enter your wallet password: ... *******
</ Mosaics to transfer in the format (mosaicId(hex)|@aliasName)::absoluteAmount, (Ex:
sending 1 symbol.xym, @symbol.xym::1000000). Add multiple mosaics separated by commas:
... @cat.currency::10000000000
</ </pre>
Enter the recipient address or @alias: ... MADIFGN27CKA6DY42JUMEFJA7OKXLONXLAEQXII

Enter the maximum fee (absolute amount): ... 200000
Select the transaction announce mode: > normal
TRANSFER
Max fee: 200,000
Network type: MIJIN

MADIFG-N27CKA-6DY42J-UMEFJA-70KXLO-NXLAEQ-XII				
test				
100,000,000,000 cat.currency (85BBEA6CC462B244)				
Signature details				
B5000000000000496383B0C2AF6B3295D615336F48B2C299AAF38619399C40 85AC7F6CF58092EAE743D5C754DC3C149E4E5EFA8E6038519F8BAAFDDF3B05BC 41B355638528AE03FDA90ACB0B4DA564FBA3D9D3A3E67A7146A77D2F5C246BC6 7AC044AAD578E161000000001605441400D03000000000517F484B18000000 60068299BAF8940F0F1CD268C21520FB9575B9B758090BA1050001000000000 44B262C46CEABB8500E87648170000000074657374				
24FACA961CB1DF4D3F76DFFBE302D2CAA512F7D1BD424CC4E0D14ACFA7221FA4				
FDA90ACB0B4DA564FBA3D9D3A3E67A7146A77D2F5C246BC67AC044AAD578E161				
✓ Do you want to announce this transaction? … yes				

TIP To check if the network confirms or rejects the transaction, run the command  ${\,\hookrightarrow\,}' \, \text{symbol-cli}$  transaction status'

#### 新規アカウントに残高があることを確認します。

先ほどエラーになっていたアカウント情報が mijin Catapult(v.2) に認識され、残高を持っていることを確認できます。

\$ <b>:</b> A	<pre>\$ symbol-cli account infoprofile mijin-no-currency-account Processing Account Information</pre>						
	Property	Value	Value				
	Address	MADIFG-N27CKA-6D	Y42J-UMEFJA-7OKXLC	D-NXLAEQ-XII			
	Address Height	2706	2706				
	Public Key	00000000000000000000000000000000000000					
	Public Key Height						
	Importance						
	Importance Height						
в	Balance Information						
	Mosaic Id	Relative Amount	Absolute Amount	Expiration Height			

1D8350FA8D4830FA 100,000 1000000000 Never	L				
10000000000 Nevel	ł	1002505700402057	100 000	1 10000000000	Netter
		1D8350FA8D4830FA	100,000		Never

# 2.2.7 AWS のトラブルシューティング

AWS Marketplace で起動した mijin Catapult(v.2) のトラブルシューティングを纏めます。

# 2.2.7.1 バックアップしたスナップショットからリストアする

本章では、AWS 上の mijin Catapult(v.2) のノードのデータを AWS Backup で取得した Snapshot からバックアップからリストアする手順を説明します。

#### 注釈:

Snaoshot からではなく、ブロックチェーンデータを消して、他のノードからリストアしたい場合は、 mijin Catapult(v.2) のノード再同期 を参照してください。

#### ノードを完全に停止する

ブロックチェーンデータを ここでは、PEER ノード 3 を例に停止します。

ポリューム ID: vel deal db Mad al (7999)           IIII         スアータスチェック         モニタ           パリューム D: グリューム D: グリューム D: グリューム D: グリューム D: グリリューム D: クリリューレ D: クリリューム D: クリリューム D: フマペア D: ジョー C: クレリショー D: フマペア D: ジョー C: D: マーク: D: ジョー C: D: D: Collapset: Allin	942 942 942 950 950 950 950 950 950 950 950 950 950	977 90 XA-791 125 565 F-ABN	1193-4029-92 <b>Ο</b> Ω
新日           ポリュームの           ダームの           ダームの           ダームの           ダームの           ダームの           ダーク           アペイタンフィーノーン ネータのTheses 1g           Outpoint ARM	サイズ ⑦ 300 GB 10% 3000 10% キーズイリアス ・ の用 ⑦ W 6 9 14 2022 1226 16 GHT=0500 (日本	タイフ 1973 スループット 125 125 125	#194027-92 ⊘ок
Outposts ARN	標準時)	- マルチアタッチ有効 いいえ	編号化なし スナップショット ・ プタッチされたインスタンス にConstructionEd 23c6 PHVIN-CATAPULT-G-peerTic AdeyAd (PhVIn-CATAPULT-G-peerTic AdeyAd (PhVIn-CATAPULT-G-peerTic
<pre>\$ sudo su - cstspult cstspult@peer1-\$ cd cstspult@peer1-\$ cd cstspult@peer1-/mij catapult@peer1-/mij stopping catapult_pee Removing catapult_pee catapult@peer3:-/miji logout \$ \$ sudo shutdown -h no</pre>	mijin-catapult-packago/packag in-catapult-packago/packag in-catapult-packago/packag command bash -c /bin/bash /scr in-catapult-packago/packag er.nodg] done er.nodg] done er.nodg] done er.nodg] done in-catapult-packago/packag in-catapult-packago/packag in-catapult-packago/packag in-catapult-packago/packag	ackage/peer/catapult5 ge/peer/catapult5 ge/peer/catapult5 to be the state tpts Up 0. ge/peer/catapult5 doo ge/peer/catapult5 doo ge/peer/catapult5 exi	cker-compose ps Ports 0.0.0.17900->7900/tcp kker-compose down cker-compose ps
	<pre>% sudo su - catapult \$ catapult@peerl:-\$ catapult@peerl:-\$ catapult@peerl:-/mij catapult@peer-node.l Stopping catapult_peer-node.l Removing network cat catapult@peerl:-/mij Name Command Sta catapult@peerl:-/mij logout \$ \$ sudo shutdown -h n</pre>	<pre>y sudo su - catapult catapult@cerl:-3 catapult@cerl:-%iin-catapult-packags/packa catapult@cerl:-/miin-catapult-packags/packa catapult@cerl:-/miin-catapult-packags/packa Stopping catapult_peer-node] done Removing natapult_peer-node] done Removing natapult_peer-node] done Removing natapult_peer-node] done Removing natapult_peer-node] done Removing natapult_peer-node] done Removing natapult@cerl:-/mijin-catapult-packags/packag Name Command State Socta ratapult@peerl:-/mijin-catapult-packags/packag satapult@peerl:-/mijin-catapult-packags/packag s sudo shutdown -h now</pre>	<pre>y sudo su - catapult catapult@peer3:-3 catapult@peer3:-3 catapult@peer3:-/Ain-catapult-package/package/peer/catapult% catapult@peer-node_lanne Name Command State catapult@peer-node_ldone Removing natapult_peer-node_ldone Removing natapult_peer-node_ldone Removing natapult_peer-node_ldone Removing natapult_peer-node_ldone Removing natapult_peer-node_ldone Removing natapult_peer-node_ldone Removing natapult_peer-node_ldone Removing natapult_peer-node_ldone Removing natapult_peer-node_ldone Removing natapult@peer1:-/mijin-catapult_package/package/peer/catapult% doc stoppil@peer3:-/mijin-catapult-package/package/peer/catapult% catapult@peer3:-/mijin-catapult-package/package/peer/catapult% exi logout % % sudo shutdown -h now</pre>

停止した PEER ノードの Volume をデタッチする

ノードにアタッチされているボリュームを外します。

	A. 7 A. 7 (4 (4)	<b>a</b>				
	インスタンス (1/5) 領職	C Heat	1929922	の状態 ♥ 20	7ション ▼ インスタンスを起動	
	Q Find インスタンス by attribute or tag (case-sensiti	ve)			< 1	> ©
	MUIN-CATAPULT-G × フィルターをクリ	7				
	Name	▽ インスタンス ID	インスタンス	▽ インスタン	ス マ ステータスチェ アラ・	テームの状
	MI IIN-CATAPUI T-G-api 1	L0dba039b7d99b7008	@ 北行中	⊕⊖ tšlarne	@2/2.05+++2/2E1 25-	7-6
	MUIN-CATAPULT-G-peer3	I-0a496d90f9d7afed0	回停止済み	⊕⊖ t3.large	- 75	2-4
	MUIN-CATAPULT-G-peer1	1-034b22389rba46ac7	@ 書行中	⊕⊖ tšlarne	@2/2.05++	7-6
	MUIN-CATAPUILT-G-ani2	L0#373a80e5646585	0 #674	⊕⊖ t3 larne	@2/2.00f+++/2/211 75-	5-6
	MUN-CATABULT-G-page	L0490r7e0r938r3fe8	O \$174	@ ctalarge	Q200519761 77	5=6
		104300700053005100	O KITT	of of the stande	0.12072772177	·
	117 817, Lo-406 Jonfo J7-6 10 (11)	IN CATABILIT C	=			@ ~
	1 2X9 2X: 1-08496d90f9d7afed0 (MI)	IN-CATAPULI-G-peers)				• ~
	詳細 セキュリティ ネットワーキング	ストレージ ステータスチ	チェック モニタ	リング タグ		
	▼ ルートデバイスの詳細					
	ルートデバイス名	ルートデバイスタイプ		EBS 最適	HE	
	🗇 /dev/sda1	EBS		無効		
	▼ ブロックデバイス					
	Q 70997N7X077N9-					
	ポリューム ID デバイス名	ボリュームサイズ フ	アタッチメントの	アタッチ時刻	暗号化済み KMS	45 年一1
	vol-01bc638ad7fc8e758 /dev/sda1	30 6	Эアタッチ済み	Thu Sep 08 2022	2 17:58:23 G いいえ -	
			O 199 Physics of 199 Phys.	Thu Sep 08 2022	2 17:58:23 G いいえ -	
	vol-0251ee8aef9dedf00 /dev/sdf	500 6	0199720	110 300 00 2022		
	vol-0251ee8aef9dedf00 /dev/sdf ▼ 番近のルートポリュームの置き換えタスク	500 6	0199794	110 349 00 2022		
	vol-0251ee8aef9dedf00 /dev/sdf ▼ 番近のルートポリュームの置き絶えタスク	500 G	97777344	110 349 00 1011		
再度 PFFR ノード3からポリュームを選択します	vol-0251ceBacf9dcdt00 /dev/sdf	500 G	97997##	110 349 60 1011		
再度、PEER ノード 3 からボリュームを選択します。	vol-025 tee8aef9ded100 ▼ 番浜のルートボリュームの置き換えクスク	500 6	9799790			
再度、PEER ノード3からボリュームを選択します。		500 6	979979A			
再度、PEER ノード 3 からボリュームを選択します。	vel-0231e8aef56cf00 /der/ydf ▼ 値がのルートポリュームの資産発達キスク ポリューム (1/1)	500 G	97997 <b>9</b> 0	C	アクション ▲   ポリョーんか	Dtta
再度、PEER ノード 3 からボリュームを選択します。	vie.231e8aef56ef500 Mer/yet v 部分のホートポリュームの部界第2 9 2 9 ポリューム (1/1) 0. mtm	500 G	#Ua-	C Lost	ريمــــدل# مد_ـــدل#	D作成
再度、PEER ノード 3 からボリュームを選択します。	vie.231e8aef56ef500 Mer/yef ▼ 御祈のたートポリュームの資産権者 タスク ポリューム (1/1) Q 供加 1000000000000000000000000000000000000	500 6	ポリュー スナップ		<i>₹79532</i> ▲ <del>₹93</del> -60	01¢.at
再度、PEER ノード 3 からボリュームを選択します。 	vie.231eetaerfoedm0 Mervjed     vie.231eetaerfoedm0 Mervjed     vie.231eetaerfoedm0 Mervjed     vie.231eetaerfoedm0 ×	500 C	ポリュー スナップ スナップ	で ムの変更 ショットの作成 ショットの存成	】 アクション ▲ ポリュームの く 1 → クルボリシーを作成	0作成 〉 ③
再度、PEER ノード 3 からボリュームを選択します。	(vii-231e8aef56ef00) Mer/yet ▼ 副がのたートポリュームの営産施品をスタ 第パリューム (1/1) (へ 泉市 パリューム (0 × vii) (231e88aef56ef00) ×) 『 Name マ ポリューム 0 ×	500 C フィルターをクリア マ タイブ マ サイズ	ポリュー スナップ マン・パリュー スナップ		アクション▲ ポリュームの イ 1 > クルポリシーを作成 作成日	D作成 〉 ③
再度、PEER ノード 3 からボリュームを選択します。	viii-231setaarf56cff00 (Mer/ydf     viii-231setaarf56cff00 (Mer/ydf     マ 御祈のたートポリュームの資産施まタスク     ボリューム (1/1)     Q 然恋     ボリューム 10 = vol 0211setaarf56cff00 (X)     Name マ ポリューム 10 (X)     マ パリューム 10 (X)	500 C フィルターをクリア マ クイブ マ サイズ gp3 500 GB	xy xy xy xy xy	ムの変更 ショットの作成 ショットの方式フサイ ムの別除 ムのブタッチ	アクション ▲     ボリュームの       クルボリシーを作成     ・       クルボリシーを作成     ・       クルズリシーを作成     ・	0 tf at > ③ 8 17:58 G
再度、PEER ノード 3 からボリュームを選択します。 	vol.231estastfisterf00     der/pdf     will 23-4 (1/1)     の 前部     (1/1)     の 前部     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (1/1)     (	500         C           フィルターをクリア         ア           タイプ マ         サイズ           gp1         500 GB		C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C	】 アクション▲ ポリュームの く 1 ) りんポリシーを作成 2022/09/08	0†xt → ④ 8 17:58 G
再度、PEER ノード 3 からボリュームを選択します。	vid-231e88aff56df00 (Aer/ydf     vid-231e88aff56df00 (ス)     マ 御がのたートポリュームの資産権者をスク     ボリューム (1/1)     Q. 泉市     ボリューム 0 = vst/0231e88aff56df00 (ス)     マ 前 ジューム 0 マ     マ ボリューム 0 マ	500         C           フィルターをクリア         マ           マ クイブ マ         サイズ           gp3         500 GB	xyu xyu xyu xyu xyu xyu xyu xyu		アクション ▲         ポリュームの           クルポリレーを作成         1           702/00/08         1	0†xt → ④ 8 17:58 G
再度、PEER ノード 3 からボリュームを選択します。	vid.0231estastfibiettion         Mer/yet           ▼ BROR - ト.#() A - Δ.DBFRB2939           #U ユーム (1/1)           Q. ##           #U/a - Δ.0           MU a - Δ.0231estastfibestf00           X           #U a - Δ.0           Y	500 C フィルターモクリア マ タイブ マ サイズ gp3 500 GB	ボリュー スナップ ボリュー ボリュー ボリュー ボリュー	で ムの変更 ショットの作成 ショットの作成 ショットの有力サイト ムの対象 ムのデタッチ ムの強要 ムの数 が、の下タッチ ムの数 ムの数 の 大のから た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の の た の た の の た の た の で の た の た の た の た の た の た の た の た の た の た の た の た の た の で の た の た の た の た の で た い た の た の た の た の た の で た の た の で た の た の で た の た の た の で た た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た の た つ た の た ろ た の た ろ た ろ た ろ た ろ た ろ た ろ た ろ た ろ た ろ た ろ つ た ろ ろ ろ ろ ろ た ろ ろ ろ た ろ た ろ た ろ ろ ろ た ろ た ろ た ろ ろ た ろ ろ の た ろ た ろ た ろ た ろ た ろ た ろ た ろ た ろ た ろ た ろ た ろ た ろ ろ た ろ た ろ た ろ た ろ た ろ た う た ろ た ろ ろ ろ た ろ た ろ ろ ろ ろ ろ ろ ろ ろ た ろ た ろ ろ の ろ ろ ろ ろ ろ ろ ろ ろ ろ ろ ろ ろ ろ	アクション ▲         ボリューム6           クルボリレーを作成         竹城日           2022/99/98         2022/99/98	0 <b>†#</b> # → ④ 8 17:58 G
再度、PEER ノード 3 からボリュームを選択します。		500 0 フィルターモタリア ) ク タイプ マ サイズ 男3 500 GB	ボリュー スナップ ボリュー ボリュー オリュー オリュー		】 アクション▲】 ポリュームの ・ イ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・	D ft at > ③ 8 17:58 G
再度、PEER ノード 3 からボリュームを選択します。		500 0 フィルター€クリア フ タイブ マ サイズ gp3 500 08	ボリュー         スクップ           スクップ         スクップ           スクップ         スクップ           ボリュー         ボリュー           ボリュー         ボリュー           ボリュー         ボリュー           クタを置         クタを置	C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C     C	アクション ▲         ポリユームC           メート         イト           クルポリシーを作成         作成日           3022/09/08         3022/09/08	8 17:58 G
再度、PEER ノード 3 からボリュームを選択します。	vid.231e8aef56eff00 Mer/yet      With 2 → Δ (1/1)      Q. 点点ボ     (1/1)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/2)      (1/	500 0 フィルター€クリア ) ク ダイブ マ サイズ gp3 500 GB	ボリュー スナップ ボリュー スナップ ボリュー ボリュー 高新教 タグを音	で しの変更 とコットの作成 とコットの作成 しのアタッチ しの見除 とのアタッチ しの気感アタッチ この気感がタッチ の気感がないのの登録 確	】 フクション▲ ポリューム6 ( 1 ) クルポリシーそ句成 ・ 作成日 2022/99/98	8 17-58 G
再度、PEER ノード 3 からボリュームを選択します。	vid.231e8aef56df00 Mer/ydd     vid.231e8aef56df00 (ソームの資産施品を分え)      ボリュームの資産施品を分え)      ボリューム(1/1)     、供加     ベリカーム(1/2)     ペリカーム(1/2)     ベリカーム(1/2)      ベリカーム(1/2)      (1/2	500 00 フィルターモタリア フ タイブ マ サイズ gp3 500 GB	ボリュー スナップ スナップ ボリュー ポリュー ポリュー 和助式 タグを要	で ひょうらの作成 ひょうらの作成 ひょうらの作成 ひょうらの作成 ひょうらの作成 しのアクッチ ムの発展 しのアクッチ よのアクッチ よの見たいのの管理 種	アクション ▲ ポリュームの: く 1 > クルポリレーを作成 すね日 2022/09/04	91f.82 > © 8 17:58 G
再度、PEER ノード 3 からボリュームを選択します。	vel.223 set&effection Mer/yet     vel.223 set&effection ()     マ 部分DR→ トポリュームの部界を見まえる      ボリューム(1//1)     Q 前が     ボリューム(1//1)     Q 前が     ボリューム(1//1)     Q 前が     ボリューム(1//1)     Q 前が     マ コーム(1//1)     ローム(1//1)     ロ コーム(1//1)     ローム(1//1)     ローム(1//1)     ローム(1//1)	500 0 フィルターモタリア マ タイブ マ サイズ gp3 500 GB	ボリュー スナップ ボリュー ボリュー ボリュー ポリュー キャプ キャプ キャプ キャプ キャプ キャプ キャプ	で           ムの変更         で           ショットの方式         ショットのうれます           ムの方方ック         レートのクラック           ムの方方ック         ショットのライフライン           ムの方方ック         レートのクラック           ムの方方ック         ショットのうれたいのの管理           確         1	アクション ▲     ポリュームの       メルボリレーを作成     1 >       クルボリレーを作成     竹城田       3022/09/08	0 <b>↑# 20</b> > ⊗ 8 17-58 G
再度、PEER ノード 3 からボリュームを選択します。		500 0 フィルターモタリア 7 タイブ ロ タイズ 9p3 500 0日	■ ボリュー スナップ スナップ ボリュー ボリュー ボリュー 和 500 ボリュー ボリュー	で で なの変更 20 x からの成 ショットのライフライ、 30.079 x チ 40.079 x チ 40.0	アクション ▲         ポリュームの           クルポリレーを作成         1           クルポリレーを作成         作成目           2022/00/08         2022/00/08	0 <b>†:</b> .₫ > ⊕ 8 17-58 G
再度、PEER ノード 3 からボリュームを選択します。	vid.0231estastfielder00         Mer/yet           ▼ BROR- h.ft?) Δ.DEFRE2923           #U ユーム (1/1)           Q. ##           #U/ユーム 10 - vid.0231estastfielder00           X           If2/J ユーム 0 - vid.0231estastfielder00           X           If2/J ユーム 10 - vid.0231estastfielder00           Y	500 0 フィルター€クリア ク ダイブ マ サイズ gp3 500 08	ポリュー スナップ マ ポリュ ポリュ ポリュ ポリュ ネケップ キ 単 ポリュー	C     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G     G	アクション ▲         ポリユームC           ノレボリシーを作成         作成日           3022/09/08         3022/09/08	015.58 G
再度、PEER ノード 3 からボリュームを選択します。	vol.233 estatification Meriyat     vol.231 estatification X (1/1)     (     の、前ボ     (         (パリュームの日本見をなる         (パリュームの日本見をなる         (パリュームの日本見をなる         (パリュームの日本見をなる         (パリュームの日本見をなる         (パリュームの日本見をなる         (パリコームの日本見をなる         (パリコームの日本日本日本日本日本日本日本日本日本日本日本日本日本日本日本日本日本日本日本	500         C           フィルターをタリア         -           ターイブ マーサイズ         -           gp3         500 GB	ボリュー スナップ スナップ ボリュー ボリュー ボリュー 高新教 クグを書	で よの変更 と 1 からのれば と 1 からのれば しの方参すが よの方参すが よの方参すが ため方参すが ため方参すが ため方参すが ため方参すが たの方参すが たの方参すが たの方参すが たの方参すが たの方参すが たの方参すが たの方参すが たの方参すが たの方が たの方が たの方が たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうかが たのうか たのうか たのうか たのうか たのうか たのうか たのうか たのうか たのうか たのうか たのうか たのうか たのうか たのうか たのうか たのうか たのうか たのうか たのうか たのう たの たのう たの たの たの たの たの たの たの たの たの たの	】 アクション▲ ポリューム6 イ ) 2022/59/98 2022/59/98	0作成 > ② 8 17:58 G
再度、PEER ノード 3 からボリュームを選択します。		500 C フィルターモタリア マ タイブ マ サイズ gp3 500 GB	ポリュー ストップ マ ポリュー ポリュー ポリュー 第93- 第93- 第95- 第97を著	()         ()           ()         ()         ()           ()         ()         ()         ()           ()         ()         ()         ()           ()         ()         ()         ()           ()         ()         ()         ()           ()         ()         ()         ()           ()         ()         ()         ()           ()         ()         ()         ()           ()         ()         ()         ()           ()         ()         ()         ()           ()         ()         ()         ()           ()         ()         ()         ()           ()         ()         ()         ()           ()         ()         ()         ()           ()         ()         ()         ()           ()         ()         ()         ()           ()         ()         ()         ()           ()         ()         ()         ()           ()         ()         ()         ()           ()         ()         ()         ()           ()	アクション ▲         ポリュームの           く)         く)           クルポリレーを作成         1           2022/09/08         2022/09/08	8 1758 G
再度、PEER ノード 3 からボリュームを選択します。	vel.223 set&effection Mer/pdf     vel.223 set&effection ()	500 0 フィルターモクリア マ タイブ マ サイズ gp3 500 0間	ボリュー スナップ スナップ ボリュー ボリュー ボリュー ボリュー	()         ()           ムの変更         ()         ()           ムの方面         ()         ()           本の方面	アクション ▲     ポリューム6       クルポリシーを作成     竹城日       3022/99/08	8 17-58 G
再度、PEER ノード 3 からボリュームを選択します。	viii.0231e8taeff56eff00         Mer/yet           ▼ 助がDR→ ト.ポリュームの日本目をタスク           ポリューム         (1/1)           Q. ##         (1/1)           【パリューム         (1/2)           【パリューム         (1/2)<	500         C           フィルターセクリア         -           ア         タイプ マ ワイズ           gp3         500 GB	ボリュー スナップ ボリュー ボリュー ボリュー ボリュー 和りュー	(の)         (の)           ムの支支         の方点           ムの支支         カラトの方式           ムの方型シットのライフタン         ムの方型シット           ムの方型シット         ムの方型シット           ムの方型シット         ショントのライン           ムの方型シット         ショントのライン           ムの方型シット         ショントのライン           ムの方型シット         ショントのライン           ムの方型シット         ショントのライン           ムの方型シット         ショントのシーク           ムの方型シット         ショントのシーク           ムの方型シット         ショントのシーク           ムの方型シーク         ション           ムのガーク         ション           ムのガーク         ション           ムのガーク         ション           ムのガーク         ション           ムのガーク         ション	アクション ▲ ポリコームの イ ↓ ) クルポリレーを作成 7022/05/00	8 1758 G
再度、PEER ノード 3 からボリュームを選択します。	viiii 0.231 setsattistettiitii         Markydd           ▼ BROR - ト.RTV Δ.DEFRE2 9.3 2           ポリューム (1/1)           Q. REF           パレューム 0.1000 (RE2 9.3 2)           アビューム 0.1000 (RE2 9.3 2)           マ RTV Δ.DEFRE2 9.3 2)	500 C フィルターをクリア フ タイブ マ サイズ gp3 500 cm	ボリュー スナップ スナップ マ ↓ ボリュー ボリュー 高級新約 タグを著	で           ムの変更           ショットの方式をジェットの方式           ショットの方式をジェットの           ムの変更           ムの変更           ムの変更           ショットの方式を           ムの変更           ムの変更           ムの変更           ショットの方式を           ムの変更           ムの変更           スクラッジ           ムの変更           スクラッジ           ムの変更           マッジ           スの変更           マッジ           スのラッジ           ムの変更           マッジ           スの変更           マッジ           スの変更           マッジ           スの変更	アクション ▲         ポリシームビス           クルボリシーを作成         作成日           3022/09/08	90111 > ③ 817586
再度、PEER ノード 3 からボリュームを選択します。	vill.231estastfordf00         Mer/pdf           マ 田谷DR → K/Y ユームの日本見 タスク           ポリユーム (1/1)           マ 田谷DR → K/Y ユームの日本見 タスク           パリ ユーム (1/1)           マ 田谷DR → vol (221 testastfordf00 X)           「パリ ユーム 10 - vol (221 testastfordf00 X)           マ 田谷DR → vol (231 testastfordf00 X)           ● - vol (231 testastfordf00 X)	500         0           フィルターをクリア         )           タイブ マ サズ         993           993         500 GB	ボリュー スナップ マー ボリュー ボリュー ボリュー ボリュー	していたい しの支援 ビュットの方式ですく しの方法 しの方法 しの方法 しの方法 しの方法 しの方法 しの方法 しつうため しの方法 しの方法 しつうため しの方法 しつうため しの方法 しつうため しの方法 しつうため しつうため しつうため しつうため しつうため しつうため しつうため しつうため しつうため しつうため しつうため しつうため しつうため しのうため しつうため しのうため しのうため しのうため しのうため しつうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのうため しのう しのうため しのう しのうため しのう しのうため しのう しのうため しのう しのう しのう しのう しのう しのう しのう しのう	アクション▲     ポリューム6       クルポリレーを作成     1       アメリレーを作成     1       2022/09/08	8 17:58 G
再度、PEER ノード 3 からボリュームを選択します。	vid.233eteatrification         Mar/yet           ▼ BROR- 5.473 6.000 FB B2 93 3           # U コーム (1/1)           Q. ##           #U コーム (1/1)           Q. ##           #U コーム 0           #U コーム 0           Y B B2 - 6.473 - 4.00 Y           Q. ##           #U コーム 0           Y B P2 - 6.00 Y           Y B P3 - 7.00 Y	500     C       フ<応ター€クリンア     -       ターダイブ マ サイズ     -       gp3     500 GB	ボリュー スナップ スナップ マ ポリュー ポリュー 高新約3 クグを表	(ク)         (ク)           (A)の変更         (A)の変更           (A)の変更	アクション ▲         ポリュームC           く 1            クルポリシーを作成         作成日           2022/09/08         2022/09/08	9 <b>018</b>
再度、PEER ノード 3 からボリュームを選択します。	vill.233.estastfiction         Mer/pdf           ▼ BRDR - h.ft'J = - Δ.DBFRB2.9.2           #U _ 2 - Δ.0           Q. dt#           [ft'J = - Δ.0]           Mame           # U _ 2 - Δ.0           Y = work_21setastfishedton           X           If'J = - Δ.0           Y           Y           If'J = - Δ.0           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y           Y	500 €	ボリュー スナップ ボリュー ボリュー ボリュー ボリュー ホリュー ロ の イ	□ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○	アクション ▲     ポリューム6       クルポリレーを作成     竹城日       3022/99/08	<b>8</b> 17:58 G
再度、PEER ノード 3 からボリュームを選択します。	(vii:0231ee8aef56eff00)     (vii:0231ee8aef56eff00)     (マ 部形のルートボリュームの日本目を923     (リューム (1/1))     (マ 部形     (パリューム 0)     (マ 部形     (パリューム 0)     (マ 部     (ワ コーム 0)     (マ 部)     (マ のののののののののののののののののののののののののののののののの	500 C フィルター€クリア ク タイブ マ サイズ 留93 500 08	ボリュー スナップ スナップ ボリュー ボリュー ボリュー 第四 オップ キャー 第四 オップ オップ	(ワ)           ムの変更           ショットの約729×5           ムの方型ッチ           ムの方型ッチ           ムの方型ッチ           ムの方型ッチ           ムの方型ッチ           ムの方型・ショントの方(な)           東京	アクション ▲         イリュームの           ノトボリレーを作成         ・           フルボリレーを作成         ・           マンコンの/04         ・	D D D D D D D D D D D D D D D D D D D
再度、PEER ノード 3 からボリュームを選択します。	vel.223 set&effection     // 小しの日本を見ままえ      ボリューム(1/1)     ・ 新りのト・ボリュームの日本を見ままえ      ボリューム(1/1)     ・ ペークンの日本を見ままえ      ボリューム(1/1)     ・ ペークンの日本を見まれ      ・ ペークンの      ・ ペークンの日本を見まれ      ・ ペークンの      ・ ペークンの日本を見まれ      ・ ペークンの      ・ ペークンの日本を見まれ      ・ ペークンの      ・ ペークン	500 C	ボリュー スナップ スナップ ボリュー ボリュー ボリュー ポリュー	□ ↓ 0 支援 ショントの行業 ショントの行業 ショントの行業 ショントの行業 ショントの行業 ショントの行業 ショントの行業 ショントの行業 単 ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・	アクション ▲     ポリシームでの       クルポリシーを作成     中成日       フロンパッジー     中成日       フロンパッジー     フロンパッジー	8 1758 G
再度、PEER ノード 3 からボリュームを選択します。	vel.231ee8aef56ef00     ////     vel.231ee8aef56ef00     ////     vel.231ee8aef56ef00     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ////     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     ///     /////     /////     /////     /////     //////	500 C	ポリュー スナップ マ スナップ ポリュー オー パリュー ポリュー ポリュー ポリュー オー オー マ オー オー マ オー オー オー オー	()         ()           ムの麦夏         3 × 5 0 5 7 4 7 4 7 4 7 4 7 7 1 7 1 7 1 7 1 7 1 7	アクション・ パリュームの イ・ クルポリレーを作成 ・ クルポリレーを作成 ・ フロンパク(v)の ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・	0778 → ⊕ 8 1758 6 8 1758 6
再度、PEER ノード 3 からボリュームを選択します。	(vii) ム (1/1)     (スタック・トボリュームの日本目生 タスク      ボリューム (1/1)     (スタック・レボリュームの日本目生 タスク      ボリューム (1/1)     (スタック・レボリュームの      マック・マック・マック・マック・マック・マック・マック・マック・マック・マック・	500 C	ボリュー スナップ マ ↓ ボリュー ボリュー ボリュー 高級新約 タグを表	で           小の変更           小の変更           ショットの方式です。           小の変更のため方式です。           小の変更のため方式です。           小の変更のため方式です。           小の変更のため方式です。           小の変更のため方式です。           小の変更のため方式です。           小の変更のため方式です。           小の変更の方式です。           小の変更のの方式です。           小の変更ののの方式です。           小の変更ののの方式です。           小の変更のののの方式です。           小の変更のののののののののの方式です。           小の変更ののののののののののののののののののののののののののののののののののの	アクション ▲         ポリユームに           () 1)         () 1)           クルボリシーを作成         作成日           302/09/00         302/09/00	9 <b>114</b> 9 8 17:38 G
再度、PEER ノード 3 からボリュームを選択します。	vel.231ee8aef5dedf00     オリューム。     ボリューム。     ロア・レダリューム。     ロア・レダリューム     ロア・レダリコア・レダリ		ボリュー スナップ ボリュー ボリュー ボリュー ボリュー ボリュー	()         ()           ムの支援         ()           ムの支援         ()           ムの支援         ()           ムの支援         ()           ムの大学	アクション ▲         ギリュームの           シルボリレーを作成         1           アンション ▲         1           アンション ●         1 <t< th=""><th>© <b>004</b> ⇒ © 8 1754 G</th></t<>	© <b>004</b> ⇒ © 8 1754 G
再度、PEER ノード 3 からボリュームを選択します。	vol. 023 testarfisientitio	300     0       フィルターをクリア     ア       ターダイブ マーワイズ     ワイズ       gp3     500 08	ボリュー スナップ スナップ ボリュー ボリュー ボリュー ボリュー ボリュー	(ク)           ムの変更           ショットの方式を知ったのであった。           ムのブタッチ           ムのブタッチ           ムのブタッチ           ムのブタッチ           ムのブタッチ           ムのブタッチ           ムの変更になった。           ムのブタッチ           ムの変更デッチ           ためのブタッチ           ムの変更デッチ           ムの変更デッチ           ムの変更ティーション           ムの変更ティーション           ムの変更ティーション           ムの変更ティーション           ション           ション </th <th>アクション (イリュームの)       ノトボリレーを作成       ウルボリレーを作成       ウルボリレーを作成</th> <th><b>000 a</b> <b>0</b> <b>0</b> <b>1758 G</b> <b>1758 G</b> <b>1</b></th>	アクション (イリュームの)       ノトボリレーを作成       ウルボリレーを作成       ウルボリレーを作成	<b>000 a</b> <b>0</b> <b>0</b> <b>1758 G</b> <b>1758 G</b> <b>1</b>
再度、PEER ノード 3 からボリュームを選択します。 ボリュームから「ボリュームをデタッチ」をクリックし、ポッ	vill 223 settler/bidd/00         Mer/ydt           vill 223 settler/bidd/00         Adv/ydt           vill 223 settler/bidd/00         X	500 C	ボリュー スナップ スナップ ボリュー ボリュー ボリュー ボリュー	□ □ □ □ □ □ □ □ □ □ □ □ □ □	アクション ▲     ポリシームでの       クルポリシーを作成     中成日       3022/09/00	<b>● ○</b> ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ●
再度、PEER ノード 3 からボリュームを選択します。 ボリュームから「ボリュームをデタッチ」をクリックし、ポッ プアップした画面で OK を押します。	Vel.0231ee8acf56ecf00     Vel.0231ee8acf56ecf00     Vel.0231ee8acf56ecf00     Vel.0231ee8acf56ecf00     Vel.0231ee8acf56ecf00     Vel.0231ee8acf56ecf00     Vel.0231ee8acf56ecf00	500 C	xy3 xy3 xy3- xy3- xy3- xy3- xy3- a∎§53 y7€8	〇         〇           ムの変更         23 × h 0 h 7 2 × h           ムの方向の         25 × h 0 h 7 2 × h           ムのブラッチ         ムの変換アシッチ           ムの変換アシッチ         ムの変換アシッチ           ムの変換アシッチ         レークシッチ           ムの変換アシッチ         レークショー           ムの変換アシッチ         レークショー           ムの変換アシッチ         レークショー           ムの変換アシューショー         シークショー           ムの変換アシューショー         レークショー           ムの変換アシューショー         レークショー           ムの変換アシューショー         レークショー           ショー         レークショー           レークショー         レークショー           レーク         レーク           レーク         レーク           レーク         レーク           レーク         レーク	アウション ▲     イリュームの       ノルボリレーを作成     ・       フルボリレーを作成     ・       マンコンのがほ     ・	2074 → ③ # 1758 G
再度、PEER ノード 3 からボリュームを選択します。 ボリュームから「ボリュームをデタッチ」をクリックし、ポッ プアップした画面で OKを押します。	vel.0231ee8aef56ef00     /// / / / / / / / / / / / / / / /	300 C	ガリュー スナップ スナップ マ↓ ポリュー ポリュー ポリュー ネー 新新 クグを著	で           小の変更           ショットの方式をジェットの方式           ショットの方式をジェットの           ムの変更           ムの変更           ショットの方式をジェットの           ムの変更           ムの変更           ロージョン           ムの変更           ロージョン	アクション ▲         ポリユームに           () 1)         () 1)           クルボリンーを作成         作成日           3022/09/00         3022/09/00	© PP04

# AWS Backup した Snapshot からボリュームを作成する

	C Model         X         Analon 75/25/X-12 (MM6 (1/1)) tel         C         IS 28/8         C Million per baller         795/25.4         Model/25/25/25/25/25           Model         Million Per 2         C million per baller         (1.1.1.0)         (1.1.1.0)
	R2 P + 5 = 8 − τ         Merc VMS coll V & erg X         2 + 2 + 5 − 5 − 7 − 5 − 5 − 5 − 5 − 5 − 5 − 5 −
	У/У         В         Мак случат сумит         ин суптаточных         вывыду в селемовляющих разоков или что что.         воласница у в селемовляющих разоков или что что.         воласница у в селемовляющих разоков или что
	102892 == 102892847 
	ner 2010-1 Xafa 1912XA Seng Reg
	TO THE STATUS
	• (1) (1) (1) (1) (1) (1) (1) (1) (1) (1)
	He DSDY     Test Soft State And Its ami-On/SS2/SdMaba1 (MUN-CATAPULT-G-geer5)     Back State And Stat
	Table # Mail Eng Art-Fy タア     Set Table # Art Fyeld # Art
	- Б-1971(28, Б-1974) • Фонулар (Д. Анарала) • У 2009/9742 • У 2009/9742
	ホーペア - 2000分割 Viol Viol
	V E=7/1959597         De= 6/05000114888102         January 100         B00         gpd         Vx0.1         B1/v         -         -         -           D=1/05371-         Jenoved         Model         Model         Model         Model         Model         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         - <t< th=""></t<>
AWS Backup の場合、AMI から mijin のデータの Snapshot を選択します。	
手動で取得した Snapshot を選択する場合は、この手順はスキップします。	
	• Instructions         X         x2 > 37 x77 x5 >> > yuu susannoon           • Instructions         X         x2 > 37 x77 x5 >> > yuu susannoon           • Instructions         X         x2 > 37 x77 x5 >> > > > > > > > > > > > > > > > > >
	18270-1825- イベット スナップショットの目前 ップ スナップショットの目前
	NIE ステレジャントロ やくズ 近の次乱 ステレ マークンステンス ヴ Nop-Generativalization ヴ Nop See の Philippe Locory の アメーンス アメンス相互定定 マークンステンス ヴ Nop-Generativalization ヴ Nop See の Philippe Locory の アメーンス 用意用の大変音響
	スポットリクエスト         取代         的ダイー0         100 ダイーズ (17)ス         100 ダイーズ (17)ス           Sanguran         取代なし         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・         ・
	Image: Control of Co
	• 4/4=9 M
	Prot 0.00
	スナップショント
	Stat/Y1977
	4~77 8+97-97/29-74 7λ
「フナップショットからギリュームを佐成」をクリック」ます	
「ステップショットからホリユームをTF成」をクリックします。	
	EC2 〉 スナップショット 〉 snap-0a96649614a8f231d 〉 ポリュームの作成
	ホリュームの作成 🝿 同じアベイラビリティーゾーン肉の任意の EC2 インスタンスにアタッチする Amazon EBS ポリュームを作成します。
	ホリューム設定
	スナップショットID
	ポリュームタイプ 博興
	汎用 SSD (gp3) 🔻
	サイズ(GIB) 雪幅
	500 単小:1 GBA 単大: 16384 GBA、個は整数である必要があります。
	IOPS
	3000 最小: 5000 IOPS、最大: 16000 IOPS、個は整致である必要があります。
	スループット(MB/砂) 情報
	125 最小: 125 MIR、 最大: 1000 MIR。 ペースライン: TB ごとに 125 MIR/砂。
	アペイラビリティーゾーン 情報
	ap-normeast-1c
	○○巡ステップショット復元 情報 □ 選択されたステップショットでは有効になっていません
	翰哥化 雪輕
	EC-インスクンスに開始付けられたEBSリゲースの開始だりリューションとして、Amazon EBS 開始にを説明します。 こ このポリュームを開発化する
	ダヴ・オブション 第 タブ: Ant リーンスに使いてきシントデモ、各タブはキーとオブションの値で構成されています、タブは、リソースの検索とフィルタリ ング あば 2000 のこれでの通知できす。
	リソースにタグが限進付けられていません。
	タグを追加
	さらに 50 数の9グを追加できます。
	キャンセル ボリュームの作成
│ │ボリュームタイプ「gp3」を選択	
アベイラビリティーゾーンを選択 / ードを空今に停止する を参照	
ノントノビッティーノーノを送バノードで元王に庁エックで参照	

# PEER ノードにリストアした Volume をアタッチする

ノードに暗号化済みボリュームをアタッチします

	ポリューム (1/1) C アクション * ポリュームの作品
	Q、使家 ポリュームの使更 < 1 > ◎ スケップショットの作成 マーチル・ホール マー スティップショットの作成
	- vol-0563776903665623 gp2 500 GB ポリュームのアタッチ     2022/09/09 1556 GP
	ボジュームログラッチア ポリュームの強制デタッチ
	劇明的化された UO の管理 タグを管理
暗号化済みホリュームを選択し、「ホリュームのアタッチ」をクリックしま	
す。	
	EC2 〉 ポリューム 〉 vol-05da77d9d0b65d623 〉 ポリュームのアタッチ
	ボリュームのアタッチー
	ハンユームのテラック 備報 インスタンスにポリュームをアタッチし、通常の物理ハードディスクドライブと同じように使用します。
	基本的な詳細
	<ul> <li></li></ul>
	845 B
	gi 4- 2 VM
	🖸 vol-05da77d9d0b65d623
	アペイラビリティーゾーン
	ap-northeast-1c
	ト-0a496d90f9d7afed0 ▼ 【ご 選択したポリュームと同じアペイラビリティーゾーンにあるインスタンスのみが表示されます。
	デバイス名 情報
	/dev/sdf
	Linux 用の理算デバイス名: ルートボリュームの場合は /dev/sda1。データボリュー
	① ここで入力された(および詳細情報に表示される)デバイス名が/dev/sdfから
	/dev/sdp であっても、新しい Linux カーネルによっては内部でナバイスの名前 が /dev/xvdf から /dev/xvdp に変更されることがあります。
	キャンセル ポリュームのアタッチ
インスタンスを PEER ノード 3 を選択し、テタッナしたホリュームと同じバ	
スを指定します。	
「ボリュームのアタッチ」をクリックします。	
	<b>インスタンス</b> (1/5) *** ⑦ ジェン ▼ <b>インスタンスの状態 ▼</b> アクション ▼ <b>インスタンスを起意 ▼</b>
	Q. Prof √>X.9 × 24 protections or tag (core-sensitive)         < 1 > ∅           MUN-CATAPAULTS   X         7 < A+9 − € 21 7
	■ Name マ インスタンス ID インスタンス マ インスタンス マ ステータスチェ アラームの状態
	●         MUN-CATAPULT-G-spil         I=Otha0356759907008         ● 沢打中         ● QQ         ロStarge         ● Q22 のチェックに: アラーム +           図         MUN-CATAPULT-G-peer3         I=Ox896457078700         ● 原止消み         ● QQ         tStarge         -         アラーム +
	MUNI-CATAFULT-G-peer1 1-03622239cba46ac7 ② 東行中 砲包 はStarge ② 2/2 のチェックに アラーム +     MUNI-CATAFULT-G-spt2 1-095373a00x5645353 ② 東行中 砲包 はStarge ② 2/2 のチェックに アラーム +
	□ MUN-CATAPULT-G-peer2 I-0430C700C938CHe8 ②東行中 砲Q ISLinge ②2/2のチェックに)アラーム +
	ー インスダンス: i-03496d90f9d7afed0 (MUIN-CATAPULT-G-peer3) = ③ ×
	詳羅 セキュリティ ネットワーキング <mark>ストレージ</mark> ステータスチェック モニタリング タグ
	マルートデバイスの評論           ルートデバイス名           ルートデバイスタイプ
	び //exi/sta1 E85 数法 マブロックデバイス
	Q 7117999112071k9-
	ポリュームID デバイス名 ポリュームサイズ アクッチメントの アクッチ発明 暗号化済み IOMSキー: vsk-01xd558d/f/dxf755 /dxx/bdg1 50 (0.799ッチ系3A Thu-two.0631072.174871-0 1.11.*3
	vol.025 tee8aet96ed100 /bev/xdf 500 ② 79 ッチボル Thu 5ep 06 2022 175823 6 [157] [40d
	▼最近のルートボリュームの豊き換えタスク Q タスクホフィルの少しング Lートボリューイキ部本地ネット
	タスクロシ         タスクロシ         タスクロシ         タズクロシ         タブ
	最近のルートボリュームの間含絶えタスクなし
リストアしたボリュームがアタッチされていることを確認します。	

### 停止した PEER ノードを起動する

停止していたノードを起動し、mijin Catapult(v.2) もあわせて起動します。



#### 2.2.7.2 アベイラビリティゾーン (AZ) 障害時の対応方法

本章では、AWS 上の mijin Catapult(v.2) の VPC のアベイラビリティゾーン (AZ) に障害あった場合、どのように対応していくかを説明します。

#### mijin Catapult(v.2) 製品版の構成

mijin Catapult(v.2) 製品版では、1つのリージョンに VPC を配置し、二つのアベイラビリティゾーン (AZ) にノードを分散しています。

ELB 設置をデプロイ時のパラメーターで有効にすることで、片方の AZ に障害があっても、サービスを継続することができます。

mijin Catapult(v.2) は一台の PEER ノードがあれば、ブロックチェーンデータを更新することができます。 プログラムなどからアクセスしたい場合は、API ノードが一台あれば、アクセスを継続することができま す。

例えば、以下の図にあるように左側 AZ に囲まれたサービス全体が停止したとしても、mijin Catapult(v.2) は停止することはありません。



#### AZ 復旧後の対応方法

AZ 復旧後の対応は簡単です。 mijin Catapult(v.2) は、復旧後 (EC2 インスタンス起動後)、自動で動いているブロックチェーンデータが あるノードに接続し、データの同期を開始します。 そのため、とくに復旧に特別な作業を必要としません。 ノードが復旧しているかを確認する場合は、以下のようにコマンド又はブラウザーから確認することで確 認することができます。

http://mijin エンドポイント:3000/node/peers

```
→ "B319300B02B12264B7DF867F0EFD583CC3C6E65ED2732E3FD77BBC1DE8E00E85",
 "roles": 70,
 "port": 7900,
 "networkIdentifier": 96,
 "host": "api1.mijin.internal",
 "friendlyName": "api1.mijin.internal"
 },
 {
 "version": 0,
 "publicKey": "82DA8AE358AC7DF7BC97103A6ABE0F791A1655E20633CC387ACE198A0B7E9AA0",
 "networkGenerationHashSeed":
→ "B319300B02B12264B7DF867F0EFD583CC3C6E65ED2732E3FD77BBC1DE8E00E85",
 "roles": 69,
 "port": 7900,
 "networkIdentifier": 96.
 "host": "peer2.mijin.internal",
 "friendlyName": "peer2.mijin.internal"
 },
 "version": 0,
 "publicKey": "4EE257A9DD6D3F19331A467C6C76BA86B50B1297181E32C7A83C1184B666996C",
 "networkGenerationHashSeed":
↔ "B319300B02B12264B7DF867F0EFD583CC3C6E65ED2732E3FD77BBC1DE8E00E85",
 "roles": 69,
 "port": 7900,
 "networkIdentifier": 96,
 "host": "peer1.mijin.internal",
 "friendlyName": "peer1.mijin.internal"
 },
 "version": 0,
 "publicKey": "C158D513266B2C04216CDC03AD99036757A41AD2AFDF59D2A67F6D2D4F8CC84F",
 "networkGenerationHashSeed":
→ "B319300B02B12264B7DF867F0EFD583CC3C6E65ED2732E3FD77BBC1DE8E00E85",
 "roles": 69,
 "port": 7900,
 "networkIdentifier": 96,
 "host": "peer3.mijin.internal",
 "friendlyName": "peer3.mijin.internal"
 }
```

上記でノードが確認できない場合、急なノードによって、ブロックチェーンデータが壊れてしまう可能性 があります。 その場合は、mijin Catapult(v.2)のノード再同期 やバックアップしたスナップショットからリストアする を参照し、ノードを復旧してください。

# 2.2.8 AWS MarketPlace mijin Catapult(v.2) FAQ 一覧

AWS MarketPlace で展開する mijin Catapult(v.2) における FAQ 一覧です。

#### 2.2.8.1 製品版 FAQ 一覧

----

Q.Free Trial 版からデータを引き継ぐことは可能ですか? A. データを引き継ぐことはできません。

---

Q. 商用での利用は可能でしょうか? A. 可能です

---

Q.mijin ライセンス費用はどこに含まれているでしょうか? A.mijin のライセンス費用は、AWS より請求されます。 EC2 インスタンスの費用同様に1時間単位の利用料金として請求が追加されます。

----

Q. 起動するのに初期費用はかかりますか?

A. 初期費用はかかりません。ただし、従量課金で mijin ライセンス費用と AWS 使用料が発生します。

---

Q. 製品版のフリートライアル期間はありますか?

A. 製品版のフリートライアル期間はございません。

フリートライアル版を別で用意していますので、AWS Marketplace を使ったデプロイ準備 のトライ アル板を参照してください。

----

Q. サポートを受けたいのですが、どこに問い合わせればよいでしょうか? A. ケースによって問い合わせ先が異なりますので、以下をご参照下さい。

■ 構築時の問題または mijin に関する技術問い合わせ (有償サポート)

※ mijin に関する技術問い合わせは、有償サポートチケットの購入が必要となりますので以下からお 問い合わせください。

https://mijin.io/en/aws_contact

■ AWS 上のトラブル

https://aws.amazon.com/jp/premiumsupport/tech-support-guidelines/

----

Q.mijin Catapult (v.2) 製品版のバージョンを教えてください A.

catapult-server: 1.0.3.8 catapult-rest: 2.5.0

(2025/6/10 時点)

---

Q. バージョンアップは可能でしょうか?

A. バージョンアップすることは可能です。

ただし、バージョン差が大きい場合、バージョンアップができない可能性もあります。 有償サポートを購入することで、バージョンアップのサポートやアナウンスを受け取ることができ ます。

https://mijin.io/en/aws_contact

---

Q. 同じリージョンに二つ目の mijin Catapult(v.2) を立ち上げようとして失敗しましたが、なぜでしょうか。

A. 二つ目以降の mijin Catapult(v.2) を立ち上げる場合はデプロイ時のパラメーター 「**mijinStackAlreadyExist** 」の値を **YES** に指定してください。

---

Q. 災害対策としてディザスタリカバリ環境を構築したいです。

A. 製品版は標準でマルチ AZ 環境にノードを分散して配置します。

詳しくは、AWS MarketPlace mijin Catapult(v.2) アーキテクチャパターンによるリカバリ戦略を参照してください。

----

Q.AWS Systems Manager Parameter Store にあるデータをローテーションしたいです。 A.AWS Systems Manager Parameter Store にあるデータは、初期デプロイ時にのみ使用され、 **バックアップ**として保存されています。 Parameter Store にはノードの暗号化通信に使用する証明書データがあるため、更新したい場合は、 ノード間の暗号化通信の更新 を参照してください。

---

Q.mijin は arm(Graviton) のインスタンスでも動きますか? A.AWS Marketplace には x86_64、arm それぞれのバージョンにて提供しています。 AWS MarketPlace Enterprise x86_64 Version AWS MarketPlace Enterprise arm64 Version

Q.arm(Graviton) 版と x86_64 版の差はありますか?

A.mijin としては、それぞれの CPU アーキテクチャに最適化されていますが速度は同等であり、ライセン ス料金も同一です。

arm で立ち上げた方が、AWS 利用料ランニングコストをを抑えることができます。 価格差については、それぞれの最低要件のインスタンスにおけるシミュレーションデータを参考に してください。

(※ シミュレーションデータに mijin のライセンス費用は含まれません)

x86_64 構成 https://calculator.aws/#/estimate?id=3df2b4611ffde3cc598ffc6fec9aff49b8a986b2 arm64 構成

https://calculator.aws/#/estimate?id=c3bdc61df9a07f9760fdb790680cec8d3807b0dc

---

2.2.8.2 フリートライアル版 FAQ 一覧

---

Q. 機能制限はありますか?

A.mijin としての機能制限はございませんが、以下の制約を行っております。
 1. 基軸通貨を「2,000cat.currency」と少額に制限しております。
 2.Mosaic、Namespace、トランザクション発行手数料が必要になります。

---

Q.利用期間の制限はありますか?

A. システム上での利用期間の設定はございませんが、あくまでテストライセンスのため、商用利用不可、 サポート問い合わせできないなどございます。 ----

Q. 商用利用可能な製品版はありますか?

A. 製品版は商用利用可能です。

詳しくは、AWS Marketplace を使ったデプロイ準備 を確認してください。

---

Q.mijin の機能についてのサポートはどこまでしていただけますか? A. フリートライアル版は、テストライセンスのため、mijin についての機能や開発のサポートは行ってお りません。

サポートが必要な場合は、製品版をお使いいただき、サポート契約していただくことで可能です。

---

Q.mijin Catapult (v.2) フリートライアル版のバージョンを教えてください

Α.

catapult-server: 1.0.3.6 catapult-rest: 2.4.3

---

# 2.2.9 AWS MarketPlace mijin Catapult(v.2) 利用料金比較表

AWS MarketPlace で展開する mijin Catapult(v.2) では、AWS の EC2 インスタンスを自由に選択すること ができます。

ライセンス費用以外で発生するサーバ利用料金の比較表です。

プロセ ッサ	EC2	EBS	VPC	リージョン	イ ンス 支 払 方法	12 か月 (\$)	月額 (\$)	3 ヶ月概 算 (\$)	月額 (円)	3 ヶ月概 算 (円)	見積
x86	t3.large * 5	GP2 30GB	新規	us-east- 1	オ ン デ マンド	4914.00	409.50	1227	¥57,330	¥171,780	見積
				us-east- 1	リザー ブド 1 年	3399.84	283.32	849	¥39,665	¥118,860	見積
				ap- north- east1	オ ン デ マンド	6385.32	532.11	1596	¥74,495	¥223,440	見積
				ap- north- east1	リザー ブド 1 年	4419.88	368.32	1104	¥51,565	¥154,560	見積
			既存	us-east- 1	オ ン デ マンド	4124.64	343.72	1029	¥48,121	¥144,060	見積

次のページに続く

				1		213 2000	,0				
プロセ ッサ	EC2	EBS	VPC	リ ー ジ ョン	インス タンス 支払い 方法	12 か月 (\$)	月額 (\$)	3 ヶ月概   算 (\$)	月額 (円)	3 ヶ月概 算 (円)	見積
				us-east- 1	リザー ブド 1 年	2610.48	217.54	651	¥30,456	¥91,140	見積
				ap- north- east1	オ ン デ マンド	5297.64	441.47	1323	¥61,806	¥185,220	見積
				ap- north- east1	リザー ブド 1 年	3332.20	277.68	831	¥38,876	¥116,340	見積
	API c5n.2xlage * 2 PEER c5n.xlarge * 3	IO1 IOPS100 130GB * 2 IO1 IOPS100 80GB * 3	新規	us-east- 1	オンデマンド	15474.96	1289.58	3867	¥180,541	¥541,380	見積
				us-east- 1	リザー ブド 1 年	10018.84	834.90	2502	¥116,886	¥350,280	見積
				ap- north- east1	オ ン デ マンド	19378.92	1614.91	4842	¥226,087	¥677,880	見積
				ap- north- east1	リザー ブド 1 年	12506.88	1042.24	3126	¥145,914	¥437,640	見積
			既存	us-east- 1	オ ン デ マンド	14685.60	1223.80	3669	¥171,332	¥513,660	見積
				us-east- 1	リザー ブド 1 年	9229.48	769.12	2307	¥107,677	¥322,980	見積
				ap- north- east1	オ ン デ マンド	18291.24	1524.27	4572	¥213,398	¥640,080	見積
				ap- north- east1	リザー ブド 1 年	11419.20	951.60	2853	¥133,224	¥399,420	見積
Arm	t4g.large	GP2 30GB	新規	us-east- 1	オ ン デ マンド	4213.20	351.10	1053	¥49,154	¥147,420	見積
				us-east- 1	リザー ブド 1 年	2994.84	249.57	747	¥34,940	¥104,580	見積
				ap- north- east1	オ ン デ マンド	5404.20	450.35	1350	¥63,049	¥189,000	見積
				ap- north- east1	リザー ブド 1 年	3844.88	320.41	960	¥44,857	¥134,400	見積
			既存	us-east- 1	オ ン デ マンド	3423.84	285.32	855	¥39,945	¥119,700	見積
				us-east- 1	リザー ブド 1 年	2205.48	183.79	549	¥25,731	¥76,860	見積
				ap- north- east1	オンデマンド	4316.52	359.71	1077	¥50,359	¥150,780	見積
				ap- north- east1	リザー ブド 1 年	2987.20	248.93	744	¥34,851	¥104,160	見積

表 12 - 前のページからの続き

次のページに続く

プロセ	EC 2	EBC	VPC	11 - 33	117	12 か日	日 ⁄ (¢)	2ヶ日畑	日菊(田)	2ヶ日畑	日括
ッサ	EUZ	EBO	VFC	リークョン	+ ンス タンス 支払い 方法	(\$)	万瘕 (⊅)	30月碱 算(\$)	月観(口)	37月碱 算(円)	尤惧
	API c6g.2xlage * 2 PEER c6g.xlarge * 3	IO1 IOPS100 130GB* 2 IO1 IOPS100 80GB* 3	新規	us-east- 1	オンデ マンド	10569.36	880.78	2640	¥123,309	¥369,600	見積
				us-east- 1	リザー ブド 1 年	7134.84	594.57	1782	¥83,240	¥249,480	見積
				ap- north- east1	オ ン デ マンド	13197.86	1099.82	3297	¥153,975	¥461,580	見積
				ap- north- east1	リザー ブド 1 年	8873.88	739.49	2217	¥103,529	¥310,380	見積
			既存	us-east- 1	オ ン デ マンド	9780.00	815.00	2445	¥114,100	¥342,300	見積
				us-east- 1	リザー ブド 1 年	6345.48	528.79	1584	¥74,031	¥221,760	見積
				ap- north- east1	オ ン デ マンド	12110.18	1009.18	3027	¥141,285	¥423,780	見積
				ap- north- east1	リザー ブド 1 年	7786.20	648.85	1944	¥90, <mark>839</mark>	¥272,160	見積

表 12 - 前のページからの続き

# Technical

# 3.1 mijin Catapult(v.2) の基礎知識

# 3.1.1 mijin Catapult(v.2) のアクセス方法

mijin Catapult(v.2) のアクセスは API ノードの API Rest エンドポイントにアクセスすることでブロックチェーンの操作が可能です。

以下では、コマンドでブロックチェーンの現在のブロック数を確認することができます。

```
$ curl -Ss http://mijin エンドポイント:3000/chain/info | jq -r
{
 "scoreHigh": "8",
 "scoreLow": "16778237146341708801",
 "height": "2693365",
 "latestFinalizedBlock": {
 "finalizationEpoch": 8833,
 "finalizationPoint": 5,
 "height": "1413092",
 "hash": "C6D03C81BC98C5152E429CA4D9B03BEC9C1229DCE5B79B267EA250A4D43BAAD6"
 }
}
```

項目	説明
height	ブロックチェーンの現在のブロック高
latestFinalizedBlock.height	確定したブロックチェーンのブロック高

Chrome などのブラウザでも確認することができます。



その他、sdk や cli でもアクセスすることができます。 cli のアクセス方法はmijin Catapult(v.2) を操作する を参照してください。

# 3.1.2 mijin Catapult(v.2) ステータス確認方法

mijin ノードの各状態は、REST から取得できます。 ブラウザや curl で直接確認することができます。 また、sdk でも同様に確認することができます。

REST のレスポンスに関しては、Symbol from Nem と共通となります。

https://symbol.github.io/symbol-openapi/v1.0.0/

# 3.1.2.1 ブロック高を確認する

# /chain/info

```
$ curl -Ss http://mijin-catapult-1-nlb-rest-eef0ebffe49c4aa3.elb.ap-northeast-1.
...amazonaws.com:3000/chain/info | jq -r
{
 "scoreHigh": "0",
 "scoreLow": "1400415221207545868",
 "height": "12318",
 "latestFinalizedBlock": {
 "finalizationEpoch": 78,
 "finalizationPoint": 8,
 "height": "12300",
 "hash": "7DE8B8052D35E29D0020EE7DB65BE075F0B6CEC69F17018447470E205B68175D"
 }
}
```
項目	説明
height	現在のブロック高
latestFinalizedBlock	ファイナライズブロック
finalizationEpoch	•
finalizationPoint	•
height	•

# 3.1.2.2 REST のバージョンを確認する

#### /node/server

```
$ curl -Ss http://mijin-catapult-1-nlb-rest-eef0ebffe49c4aa3.elb.ap-northeast-1.

amazonaws.com:3000/node/server | jq -r
{
 "serverInfo": {
 "restVersion": "2.3.5",
 "sdkVersion": "2.3.5",
 }
}
```

тан	∃凶田 1
「「「」」	D1.97
restVersion	rest の version
sdkVersion	rest が使用する catapult-sdk の version

## 3.1.2.3 ノード情報を確認する

/node/info

```
$ curl -Ss http://mijin-catapult-1-nlb-rest-eef0ebffe49c4aa3.elb.ap-northeast-1.

→amazonaws.com:3000/node/info | jq -r

{

 "version": 16777216,

 "publicKey": "03ECD9C1929E26ED53BEBCCCF17E6F32F37ED9C6474397F592C883F771AB6A05",

 "networkGenerationHashSeed":

 "2DE20B93EBE048A3BA132CC9874BCABBC21C87E18FE9836B8D5D002E57640D4B",

 "roles": 70,

 "port": 7900,

 "networkIdentifier": 96,

 "host": "api2.mijin.internal",

 "friendlyName": "api2.mijin.internal",

 "nodePublicKey": "5958AE940208CF8FD0D7FF2A584F8B234A3814AFC4D93F304A5CEA926EF6A747"
```

項目	説明
publicKey	ハーベストで使われる公開鍵
networkGenerationHashSeed	作成したブロックチェーン固有の GenerationHash 設定
roles	ノードのロール (api/peer/dual/voting)
port	ノード間の通信ポート
networkIdentifier	ネットワークタイプ
host	実行しているノードのホスト名
friendlyName	ノードのフレンドリー名
nodePublicKey	ノード用の公開鍵

# 3.1.2.4 接続しているノードを確認する

#### /node/peers

```
$ curl -Ss http://mijin-catapult-1-nlb-rest-eef0ebffe49c4aa3.elb.ap-northeast-1.
→amazonaws.com:3000/node/peers | jq -r
[
 "version": 0,
 "publicKey": "DB8D9DD59D78AE62E157824305DE31B9D415AA217EFE1DF14A7361E9D20E7456",
 "networkGenerationHashSeed":
→ "2DE20B93EBE048A3BA132CC9874BCABBC21C87E18FE9836B8D5D002E57640D4B",
 "roles": 69,
 "port": 7900,
 "networkIdentifier": 96,
 "host": "peer1.mijin.internal",
 "friendlyName": "peer1.mijin.internal"
 },
 {
 "version": 0,
 "publicKey": "22722F1534AE77DA44A065C0E2ACB125CB66FB45E80403A183EFEBE222BF3D90",
 "networkGenerationHashSeed":
→ "2DE20B93EBE048A3BA132CC9874BCABBC21C87E18FE9836B8D5D002E57640D4B",
 "roles": 69,
 "port": 7900,
 "networkIdentifier": 96,
 "host": "peer2.mijin.internal",
 "friendlyName": "peer2.mijin.internal"
 },
 {
 "version": 0,
 "publicKey": "239CC13A2B3D112C4146415EE532146D5338614BBBAD1A1E2E8E4690638F07D9",
 "networkGenerationHashSeed":
→ "2DE20B93EBE048A3BA132CC9874BCABBC21C87E18FE9836B8D5D002E57640D4B",
 "roles": 69,
 "port": 7900,
 "networkIdentifier": 96,
 "host": "peer3.mijin.internal",
 "friendlyName": "peer3.mijin.internal"
 },
 "version": 0,
 "publicKey": "E4BF3706483B4D42243F3DCB2625021C3E3AE7C253CC466154EEDF9775012C20",
 "networkGenerationHashSeed":
→ "2DE20B93EBE048A3BA132CC9874BCABBC21C87E18FE9836B8D5D002E57640D4B",
```

```
"roles": 70,
"port": 7900,
"networkIdentifier": 96,
"host": "api1.mijin.internal",
"friendlyName": "api1.mijin.internal"
}
```

項目	説明
publicKey	ハーベストで使われる公開鍵
networkGenerationHashSeed	作成したブロックチェーン固有の GenerationHash 設定
roles	ノードのロール (api/peer/dual/voting)
port	ノード間の通信ポート
networkIdentifier	ネットワークタイプ
host	実行しているノードのホスト名
friendlyName	ノードのフレンドリー名
nodePublicKey	ノード用の公開鍵

# 3.1.2.5 総トランザクション数、総アカウント数を確認する

## /node/storage

```
$ curl -Ss http://mijin-catapult-1-nlb-rest-eef0ebffe49c4aa3.elb.ap-northeast-1.

amazonaws.com:3000/node/storage | jq -r
{
 "numBlocks": 12322,
 "numTransactions": 34,
 "numAccounts": 14
}
```

項目	説明
numBlocks	現在のブロック高
numTransactions	過去発行されたトランザクション数の総数
numAccounts	過去使用されたアカウントの総数

# 3.1.2.6 ネットワークタイプを確認する

` /network `

```
$ curl -Ss http://mijin-catapult-1-nlb-rest-eef0ebffe49c4aa3.elb.ap-northeast-1.

amazonaws.com:3000/network | jq -r
{
 "name": "mijin",
 "description": "mijin network"
}
```

項目	説明
name	使われているネットワーク名 mijin or mijin-test
description	ネットワークの説明

# 3.1.2.7 ノードのコンテナの状況を確認する

## /node/health

項目説明apiNodeapi-node コンテナのステータス up or downdbdb コンテナのステータス up or down

## 3.1.2.8 ブロックチェーン全体の設定を確認する

## /network/properties

```
$ curl -Ss http://mijin-catapult-1-nlb-rest-eef0ebffe49c4aa3.elb.ap-northeast-1.
→amazonaws.com:3000/network/properties | jq -r
{
 "network": {
 "identifier": "mijin",
 "nemesisSignerPublicKey":
→ "12086D4CB80CB6461887427BD49ED22D3914117526F573CC6F9937FC19DB2F73",
 "nodeEqualityStrategy": "host",
 "generationHashSeed":
→ "2DE20B93EBE048A3BA132CC9874BCABBC21C87E18FE9836B8D5D002E57640D4B",
 "epochAdjustment": "1560294000s"
 },
 "chain": {
 "enableVerifiableState": true,
 "enableVerifiableReceipts": true,
 "currencyMosaicId": "0x61D0'A72B'3C62'5448",
 "harvestingMosaicId": "0x1248'680A'CB99'E205",
 "blockGenerationTargetTime": "15s",
 "blockTimeSmoothingFactor": "3000",
 "importanceGrouping": "40",
 "importanceActivityPercentage": "5",
 "maxRollbackBlocks": "0",
 "maxDifficultyBlocks": "60",
 "defaultDynamicFeeMultiplier": "0",
 "maxTransactionLifetime": "24h",
 "maxBlockFutureTime": "500ms",
 "initialCurrencyAtomicUnits": "8'998'999'998'000'000",
 "maxMosaicAtomicUnits": "9'000'000'000'000'000",
 "totalChainImportance": "15'000'000",
 "minHarvesterBalance": "1'000'000",
 "maxHarvesterBalance": "15'000'000",
 "minVoterBalance": "1'000'000",
 "votingSetGrouping": "160",
 "maxVotingKeysPerAccount": "3",
```

```
"minVotingKeyLifetime": "72",
 "maxVotingKeyLifetime": "26280",
 "harvestBeneficiaryPercentage": "10",
 "harvestNetworkPercentage": "5",
 "harvestNetworkFeeSinkAddress": "MBVF6QLFNKAXDBZLJYBPBT2YYKMJW7UE7GH7RTY",
 "maxTransactionsPerBlock": "6'000"
 },
 "plugins": {
 "accountlink": {
 "dummy": "to trigger plugin load"
 },
 "aggregate": {
 "maxTransactionsPerAggregate": "1'000",
 "maxCosignaturesPerAggregate": "25",
 "enableStrictCosignatureCheck": false,
 "enableBondedAggregateSupport": true,
 "maxBondedTransactionLifetime": "48h"
 },
 "lockhash": {
 "lockedFundsPerAggregate": "0",
 "maxHashLockDuration": "2d"
 },
 "locksecret": {
 "maxSecretLockDuration": "30d",
 "minProofSize": "1",
 "maxProofSize": "1000"
 },
 "metadata": {
 "maxValueSize": "1024"
 },
 "mosaic": {
 "maxMosaicsPerAccount": "1'000",
 "maxMosaicDuration": "3650d",
 "maxMosaicDivisibility": "6",
 "mosaicRentalFeeSinkAddress": "MBKRTIOKHE34GF7J5WZDW6VLXEDYFRFFURN2EZA",
 "mosaicRentalFee": "0"
 },
 "multisig": {
 "maxMultisigDepth": "3",
 "maxCosignatoriesPerAccount": "25",
 "maxCosignedAccountsPerAccount": "25"
 },
 "namespace": {
 "maxNameSize": "64",
 "maxChildNamespaces": "256",
 "maxNamespaceDepth": "3",
 "minNamespaceDuration": "1m",
 "maxNamespaceDuration": "3650d",
 "namespaceGracePeriodDuration": "30d",
 "reservedRootNamespaceNames": "xem, nem, user, account, org, com, biz, net, edu,
mil, gov, info",
 "namespaceRentalFeeSinkAddress": "MBWRFMKEJRDUZC5WEW2PFYG374AI444HL2WQX6A",
 "rootNamespaceRentalFeePerBlock": "1",
 "childNamespaceRentalFee": "0"
 },
 "restrictionaccount": {
 "maxAccountRestrictionValues": "512"
```

```
},
 "restrictionmosaic": {
 "maxMosaicRestrictionValues": "20"
 },
 "transfer": {
 "maxMessageSize": "1024"
 }
}
```

この設定値については、

## 3.1.2.9 トランザクション手数料を確認する

/network/fees/transaction

```
$ curl -Ss http://mijin-catapult-1-nlb-rest-eef0ebffe49c4aa3.elb.ap-northeast-1.

→amazonaws.com:3000/network/fees/transaction | jq -r

{

"averageFeeMultiplier": 0,

"medianFeeMultiplier": 0,

"highestFeeMultiplier": 0,

"lowestFeeMultiplier": 0,

"minFeeMultiplier": 0

}
```

項目	説明
averageFeeMultiplier	平均の乗数値 (自動)
medianFeeMultiplier	中央値の乗数値 (自動)
highestFeeMultiplier	使われた最大の乗数値
lowestFeeMultiplier	使われた最小の乗数値
minFeeMultiplier	ノードに設定した最小で必要な乗数値。0 であれば、手数料なしモード

# 3.2 mijin Catapult(v.2) を操作する

Symbol という公開ブロックチェーンと mijin は元々同じものであり、デュアルライセンスのもと OSS 化しています。そのため、Symbol のツールを mijin でも使用することが可能です。ここでは、mijin Catapult(v.2) にて送金するテストを操作します。

## 警告:

symbol-cli は Public Archive となっており、正常に動作しない可能性があります。 symbol-sdk は、mijin 用にカスタムした sdk を fork しています。

# 3.2.1 mijin アカウント作成

本章では、mijin を操作するために、必ず必要となるアカウント操作について説明します。 操作は Linux(Ubuntu20.04) にて行いますので、最低限の Linux 操作を理解している前提になります。

**注釈:** mijin Catapult(v.2) を操作するにはアカウントを作成し、そのアカウントを使ってトランザク ションを送信します。また、デプロイ時の指定で手数料ありモードの場合は、アカウントに基軸通貨 (cat.currency) の残高を持つ必要があることに注意してください。

## 3.2.1.1 nodejs 及び yarn をインストール

mijin-catapult-tools を使用するため nodejs をインストールします。nodejs は NodeSource を利用して インストールします。

```
$ curl -fsSL https://deb.nodesource.com/setup_14.x | sudo -E bash - && sudo apt-get
install -y nodejs
$ node --version
$ sudo npm install -g yarn
```

#### 3.2.1.2 mijin-catapult-tools のインストール

yarn を使用し、mijin-catapult-tools をインストールします。

```
$ yarn global add @tech-bureau/mijin-catapult-tools
$ echo 'export PATH="$HOME/.yarn/bin:$PATH"' >> ~/.bashrc && source ~/.bashrc
```

mijin-catapult-tools が使えることを確認します。

```
$ mijin-catapult-tools
```

#### 3.2.1.3 アカウントの作成

まずは、テストで使用するアカウントを作成してみましょう。ここでは、**test1Account test2Account** を 使用するため mijin.json に保存します。

#### アカウント作成

```
$ mijin-catapult-tools account generate -u http://localhost:3000 -w mijin.json -s
2023-01-17T06:34:59.412Z [info] : mijin URL: http://localhost:3000
2023-01-17T06:34:59.413Z [info] : Network: 96
2023-01-17T06:34:59.413Z [info] : Mosaic Currency Id: 268CF9B2D33FBD22
2023-01-17T06:34:59.413Z [info] : Mosaic Harvest Id: 4C39D26C386E3182
2023-01-17T06:34:59.413Z [info] : Start Account Generate...
2023-01-17T06:34:59.475Z [info] : Write Config File: mijin.json
2023-01-17T06:34:59.476Z [info] : New Account: {
 "url": "http://localhost:3000",
 "workAccount": {
```

```
"publicKey": "425F06A8870381A00BD83E2D1083BB690F9FCB815F0919DFFA1A53A68E144D06",
 "privateKey": "9E9C660164AB344030DF5E77539952D9F5B380311C978369BA08923F577D8DAB",
 "address": "MC5VTRZ07IGJUG2VP0XBD4GD04A74YLFUCVZZGI"
},
"balanceAccount": {
 "publicKey": "",
 "privateKey": "",
 "address": ""
},
"mainAccount": {
 "publicKey": "F2985009341A526B17E954EB6EA3EC77E1A0B33AE31EB29F86A69D9BB283AF67",
 "privateKey": "FF9A552ED35D930378F1E6B349A677771F19A06BCA0C4D5DCA621F354F5C6956",
 "address": "MCV66SABR6MBWT2I56YI3ULCPYWREJFX5DHHGDQ"
},
"keylink": {
 "vrf": {
 "publicKey": "C074A57A0EDF633414DDD764C6771926E91596481E0C2CAD4D9C3EFAAA432BCE",
 "privateKey": "C07D9A63B8DD8B0E639AAAE6059D7FA6C554082677926B7CACF0514E3940692C",
 "address": "MBLYBGODVGLT3HED6EXAQAXJE4CHHVRWGIAS45Y"
 },
 "voting": {
 "publicKey": "F6571CBC420A4EBA09E027AA53E5DD9486642911CDC11E0D3A6D1B2E4BC228D1".
 "privateKey": "FD5744779348F77F6EA6288232D0C8944E2F3C9E24D2E0FB1E93A8C4F9DD9BA5",
 "address": "MD2M6SDIE6406ZWF2IJ7R4R5RDKOT2FNJDUIY0Q"
 }
},
"test1Account": {
 "publicKey": "CB51613497A40D9A256B17932579BC64D5037A04B29737B944965C1ADADD6E04",
 "privateKey": "8D5969EF1796F5F90256C92B5017396E40786ED87995169D4E26C9E5E01D7F8C",
 "address": "MAS36UGDCOGG6GYCBDPX3ROISABSNPZ6JQXMJSA"
},
"test2Account": {
 "publicKey": "5CC14799D5B2643914C8E574C8D073A7EE9AE2A405F1339A53612B566498AB1D",
 "privateKey": "FDE625C1D53AF04533FEB06A3556679FC02C4C6246952D3A534EB3E19CF83C56",
 "address": "MC5AH4UGBPPHNCMPTVNSY6LRDPIMEDODS20373A"
}
```

## この時点で test1Account、test2Account を確認しても、存在しないことになっています。 ブロックチェーン上で、このアドレスのやり取りがないため、存在を確認できないからです。

```
$ catapult@catapult:~$ mijin-catapult-tools account info -r mijin.json -t test1
2023-01-17T06:36:26.600Z [info] : mijin URL: http://localhost:3000
2023-01-17T06:36:26.601Z [info] : Network: 96
2023-01-17T06:36:26.601Z [info] : Mosaic Currency Id: 268CF9B2D33FBD22
2023-01-17T06:36:26.601Z [info] : Mosaic Harvest Id: 4C39D26C386E3182
2023-01-17T06:36:26.601Z [info] : Start Account Info
2023-01-17T06:36:26.620Z [error] : Address Not Found
```

```
$ mijin-catapult-tools account info -r mijin.json -t test2
2023-01-17T06:36:43.387Z [info] : mijin URL: http://localhost:3000
2023-01-17T06:36:43.387Z [info] : Network: 96
2023-01-17T06:36:43.387Z [info] : Mosaic Currency Id: 268CF9B2D33FBD22
```

```
2023-01-17T06:36:43.387Z [info] : Mosaic Harvest Id: 4C39D26C386E3182
2023-01-17T06:36:43.387Z [info] : Start Account Info
2023-01-17T06:36:43.405Z [error] : Address Not Found
```

## 手数料について

手数料とは、以下の項目で発行するトランザクションによって手数料が発生し、基軸通貨 (cat.currency) にて支払う必要があります。

- トランザクション手数料
- Mosaic レンタル手数料
- Namespace レンタル手数料
- ・ ロック保証金

マーケットプレイスで提供する mijin では手数料モードは標準で **なし**に設定されています。 手数料なしでは、アカウントに基軸通貨の残高を持つ必要がありませんので特に意識することはありませ んが、sdk などで手数料設定を都度 0 を設定する必要があります。 手数料ありで mijin を作成した場合は、(番外) 手数料ありモード時の、基軸通貨の移動方法 の項目を参照 し、**test1Account** のアカウントに基軸通貨を送信してください。

#### 3.2.1.4 Mosaic を作成し転送

まずは基本となる Mosaic を作成し、転送してみましょう。

#### Mosaic の作成

test1 アカウントをオーナーとした1つの発行数の Mosaic(Token) を発行します。

```
$ mijin-catapult-tools transaction mosaic create -r mijin.json -o test1 -s 1 -d 0
2023-01-17T06:42:51.942Z [info] : mijin URL: http://localhost:3000
2023-01-17T06:42:51.942Z [info] : Network: 96
2023-01-17T06:42:51.942Z [info] : Create Mosaic...
2023-01-17T06:42:51.955Z [info] : Mosaic Owener Account:
MAS36UGDCOGG6GYCBDPX3ROISABSNPZ6JQXMJSA
2023-01-17T06:42:51.955Z [info] : MosaicId: 3D86C9FE5D52DE6F
2023-01-17T06:42:51.955Z [info] : Mosaic Flags: supplymutable:true, transferable:true,
restrictable:true, revokable:false
2023-01-17T06:42:51.955Z [info] : Mosaic Supply: divisibility:0, supply:1
2023-01-17T06:42:51.964Z [info] : Transaction Fee: 0
2023-01-17T06:42:51.964Z [info] : Mosaic Rental Fee: 0
2023-01-17T06:42:51.964Z [info] : Start Aggregate Transaction...
2023-01-17T06:43:00.967Z [info] : End Aggregate Transaction
2023-01-17T06:43:00.967Z [info] : http://localhost:3000/transactionStatus/
→ 39BBE7E083460C3B85EB7D5FA2FB486F9786CA3FF1E0F976214F5753E527383A
2023-01-17T06:43:00.967Z [info] : http://localhost:3000/transactions/confirmed/
↔ 39BBE7E083460C3B85EB7D5FA2FB486F9786CA3FF1E0F976214F5753E527383A
```

A アカウントの情報を確認すると、Mosaicld 3D86C9FE5D52DE6F を 1 だけ持っていることが確認できます。

```
$ mijin-catapult-tools account info -r mijin.json -t test1
2023-01-17T06:43:43.349Z [info] : mijin URL: http://localhost:3000
2023-01-17T06:43:43.350Z [info] : Network: 96
2023-01-17T06:43:43.350Z [info] : Mosaic Currency Id: 268CF9B2D33FBD22
2023-01-17T06:43:43.350Z [info] : Mosaic Harvest Id: 4C39D26C386E3182
2023-01-17T06:43:43.350Z [info] : Start Account Info
2023-01-17T06:43:43.365Z [info] : test1 Account: {
 "publicKey": "CB51613497A40D9A256B17932579BC64D5037A04B29737B944965C1ADADD6E04",
 "address": "MAS36UGDCOGG6GYCBDPX3ROISABSNPZ6JQXMJSA",
 "mosaics": [
 {
 "id": "3D86C9FE5D52DE6F",
 "amount": "1",
 "currency": false,
 "harvest": false
],
 "keylink": {
 "vrf": {
 "publicKey": ""
 },
 "voting": {
 "publicKey": "",
 "startEpoch": "",
 "endEpoch": ""
 }
 }
```

## Mosaic を転送する

#### 先程作成した Mosaic 3D86C9FE5D52DE6F を test1 から test2 に転送します。

```
$ mijin-catapult-tools transaction transfer -r mijin.json -f test1 -d test2 -m
3D86C9FE5D52DE6F -a 1
2023-01-17T06:44:46.983Z [info] : mijin URL: http://localhost:3000
2023-01-17T06:44:46.984Z [info] : Network: 96
2023-01-17T06:44:46.984Z [info] : Start Transfer Account...
2023-01-17T06:44:46.996Z [info] : From Account Address:
MAS36UGDCOGG6GYCBDPX3ROISABSNPZ6JQXMJSA
2023-01-17T06:44:46.996Z [info] : Dest Account Address:
MC5AH4UGBPPHNCMPTVNSY6LRDPIMEDODS20373A
2023-01-17T06:44:47.005Z [info] : Start Transfer Transaction...
2023-01-17T06:44:52.197Z [info] : End Transfer Transaction
2023-01-17T06:44:52.197Z [info] : http://localhost:3000/transactionStatus/
-454DFDC48F00852A3DE355D7D2AD4B581D718D999BA177E1BB392AECD1656C6A
2023-01-17T06:44:52.197Z [info] : http://localhost:3000/transactions/confirmed/
-454DFDC48F00852A3DE355D7D2AD4B581D718D999BA177E1BB392AECD1656C6A
```

test1 アカウントの状態を確認します。 Mosaic **3D86C9FE5D52DE6F** を持っていないことを確認できます。

```
$ mijin-catapult-tools account info -r mijin.json -t test1
2023-01-17T06:45:51.931Z [info] : mijin URL: http://localhost:3000
2023-01-17T06:45:51.931Z [info] : Network: 96
2023-01-17T06:45:51.931Z [info] : Mosaic Currency Id: 268CF9B2D33FBD22
2023-01-17T06:45:51.931Z [info] : Mosaic Harvest Id: 4C39D26C386E3182
2023-01-17T06:45:51.931Z [info] : Start Account Info
2023-01-17T06:45:51.946Z [info] : test1 Account: {
 "publicKey": "CB51613497A40D9A256B17932579BC64D5037A04B29737B944965C1ADADD6E04",
 "address": "MAS36UGDCOGG6GYCBDPX3ROISABSNPZ6JQXMJSA",
 "mosaics": [],
 "keylink": {
 "vrf": {
 "publicKey": ""
 },
 "voting": {
 "publicKey": "",
 "startEpoch": "",
 "endEpoch": ""
 }
 }
```

## 続いて、**test2 アカウント**の状態を確認します。 Mosaic **3D86C9FE5D52DE6F** を所有していることが確認できます。

```
$ mijin-catapult-tools account info -r mijin.json -t test2
2023-01-17T06:46:41.737Z [info] : mijin URL: http://localhost:3000
2023-01-17T06:46:41.738Z [info] : Network: 96
2023-01-17T06:46:41.738Z [info] : Mosaic Currency Id: 268CF9B2D33FBD22
2023-01-17T06:46:41.738Z [info] : Mosaic Harvest Id: 4C39D26C386E3182
2023-01-17T06:46:41.738Z [info] : Start Account Info
2023-01-17T06:46:41.754Z [info] : test1 Account: {
 "address": "MC5AH4UGBPPHNCMPTVNSY6LRDPIMEDODS20373A",
 "mosaics": [
 {
 "id": "3D86C9FE5D52DE6F",
 "amount": "1",
 "currency": false,
 "harvest": false
 }
],
 "keylink": {
 "vrf": {
 "publicKey": ""
 },
 "voting": {
 "publicKey": "",
 "startEpoch": "",
 "endEpoch": ""
 }
 }
```

#### 3.2.1.5 (番外)手数料ありモード時の、基軸通貨の移動方法

手数料ありモードで mijin を作成した場合、トランザクション送信などに全て基軸通貨から手数料が取ら れるようになります。

そのため、アカウント作成時に基軸通貨の残高を送信する手順を説明します。

## 基軸通貨の保持アカウント登録

## 注釈:

AWS Marketplace の | mijin | を使用した場合、AWS Cloudformation Stack 内にある Outouts タブを選択 します。

Key「mijinLBEndpint」又は「mijinbEndpoint」のURLを控えてください。

次に、Key「HarvestAddress」の Value の URL をクリックしてください。

api の中にある一番はじめの Vaule にある private_key の値を控えてください。(ここでは 055E で始まる 値)

詳しくは、既存 VPC 上に、mijin をデプロイする 又は新規 VPC を作成し、mijin をデプロイする を参照し てください。

mijin-catapult-cli にて上記の private_key を import します。すでに mijin.json ファイルがある場合は、 balanceAccount のみ追記することができます。

```
$ mijin-catapult-tools account generate -r mijin.json -w mijin.json -s -p
90EEBCB77A767F8F5CCCE9D0F89A60CB2D7FCD5FD8F469E2F8BDFC0CDD8B8A2F
2023-01-17T06:55:52.167Z [info] : mijin URL: http://localhost:3000
2023-01-17T06:55:52.167Z [info] : Network: 96
2023-01-17T06:55:52.167Z [info] : Mosaic Currency Id: 268CF9B2D33FBD22
2023-01-17T06:55:52.167Z [info] : Mosaic Harvest Id: 4C39D26C386E3182
2023-01-17T06:55:52.167Z [info] : Start Account Generate...
2023-01-17T06:55:52.229Z [info] : Write Config File: mijin.json
2023-01-17T06:55:52.233Z [info] : New Account: {
 "url": "http://localhost:3000",
 "workAccount": {
 "publicKey": "425F06A8870381A00BD83E2D1083BB690F9FCB815F0919DFFA1A53A68E144D06",
 "privateKey": "9E9C660164AB344030DF5E77539952D9F5B380311C978369BA08923F577D8DAB",
 "address": "MC5VTRZ07IGJUG2VPQXBD4GD04A74YLFUCVZZGI"
 },
 "balanceAccount": {
 "publicKey": "4FAAC9BF9881893CB31BC2065E8A8D0B12364423E2A08DAF4F77E1FEF5D5B2E8",
 "privateKey": "90EEBCB77A767F8F5CCCE9D0F89A60CB2D7FCD5FD8F469E2F8BDFC0CDD8B8A2F",
 "address": "MCPYNELDE5QS63QBRZ2L7OLNZ63YPQJSOCEWXQI"
 },
 "mainAccount": {
 "publicKey": "F2985009341A526B17E954EB6EA3EC77E1A0B33AE31EB29F86A69D9BB283AF67",
 "privateKey": "FF9A552ED35D930378F1E6B349A677771F19A06BCA0C4D5DCA621F354F5C6956",
 "address": "MCV66SABR6MBWT2I56YI3ULCPYWREJFX5DHHGDQ"
 },
 "keylink": {
 "vrf": {
 "publicKey": "C074A57A0EDF633414DDD764C6771926E91596481E0C2CAD4D9C3EFAAA432BCE",
 "privateKey": "9E9C660164AB344030DF5E77539952D9F5B380311C978369BA08923F577D8DAB",
 "address": "MBLYBGODVGLT3HED6EXAQAXJE4CHHVRWGIAS45Y"
 },
```

```
"voting": {
 "publicKey": "F6571CBC420A4EBA09E027AA53E5DD9486642911CDC11E0D3A6D1B2E4BC228D1",
 "privateKey": "FD5744779348F77F6EA6288232D0C8944E2F3C9E24D2E0FB1E93A8C4F9DD9BA5",
 "address": "MD2M6SDIE6406ZWF2IJ7R4R5RDKOT2FNJDUIY0Q"
 }
},
"test1Account": {
 "publicKey": "CB51613497A40D9A256B17932579BC64D5037A04B29737B944965C1ADADD6E04",
 "privateKey": "8D5969EF1796F5F90256C92B5017396E40786ED87995169D4E26C9E5E01D7F8C",
 "address": "MAS36UGDCOGG6GYCBDPX3ROISABSNPZ6JOXMJSA"
},
"test2Account": {
 "publicKey": "5CC14799D5B2643914C8E574C8D073A7EE9AE2A405F1339A53612B566498AB1D",
 "privateKey": "FDE625C1D53AF04533FEB06A3556679FC02C4C6246952D3A534EB3E19CF83C56",
 "address": "MC5AH4UGBPPHNCMPTVNSY6LRDPIMEDODS20373A"
1
```

これで balanceAccount にインポートすることができました。 アカウントの情報を確認します。 このアカウントでは、初期時 2 つの Mosaic を持っています。 Balance Information の 268CF9B2D33FBD22 が基軸通貨 (cat.currency) となり、4C39D26C386E3182 は Harvest 有効用の Mosaic です。

**警告:** この ID は mijin を作成毎に違う値になり、基軸通貨は"currency": true と表示された ID となり ます。

```
$ mijin-catapult-tools account info -r mijin.json -t balance
2023-01-17T06:56:51.299Z [info] : mijin URL: http://localhost:3000
2023-01-17T06:56:51.299Z [info] : Network: 96
2023-01-17T06:56:51.299Z [info] : Mosaic Currency Id: 268CF9B2D33FBD22
2023-01-17T06:56:51.299Z [info] : Mosaic Harvest Id: 4C39D26C386E3182
2023-01-17T06:56:51.299Z [info] : Start Account Info
2023-01-17T06:56:51.315Z [info] : balance Account: {
 "publicKey": "4FAAC9BF9881893CB31BC2065E8A8D0B12364423E2A08DAF4F77E1FEF5D5B2E8",
 "address": "MCPYNELDE5QS63QBRZ2L7OLNZ63YPQJSOCEWXQI",
 "mosaics": [
 {
 "id": "268CF9B2D33FBD22",
 "amount": "8998977498000000",
 "currency": true,
 "harvest": false
 },
 "id": "4C39D26C386E3182",
 "amount": "15000000",
 "currency": false,
 "harvest": true
 }
],
```

```
"keylink": {
 "vrf": {
 "publicKey": "2A88BA2689D584B03A3D4B829347F0A8B63AF55A8E9F176F1D2327F9E87E22D8"
 },
 "voting": {
 "publicKey": "22BB9DCA05D483E4D4DDE764E8742E741ADA676F461D5F3E6663840C5290320F",
 "startEpoch": 1,
 "endEpoch": 26280
 }
}
```

# 3.2.2 mijin Catapult(v.2) のバージョンアップ

本章では、mijin Catapult(v.2) ノードのマイナーバージョンアップ方法を説明します。 mijin Catapult(v.2) のプログラムは、docker コンテナのバージョンを更新、必要であれば Config ファイ ルを更新することでバージョンアップすることが可能です。 バージョンアップのアナウンスは、有償サポートのお客様向けに発信しています。

## 警告:

メジャーバージョンアップや、バージョンの差異がある場合は、同期に失敗する可能性があります。 2022/9/20 時点バージョンアップはございません。

#### 3.2.2.1 Step.1

ノードにリモートログインします。 AWS MarketPlace の mijin Catapult(v.2) にログインしたい場合は、mijin Catapult(v.2) EC2 インスタンス ログイン方法 を参照してください。

## 3.2.2.2 Step.2

mijin を起動している「catapult」ユーザーにスイッチします。

```
$ sudo su - catapult
catapult@api1:~$
```

#### 3.2.2.3 Step.3

mijin の起動ファイルがあるディレクトリに移動します。 API,PEER ノードでディレクトリが違うことに注意してください。

ノード	ディレクトリ
API/Dual	mijin-catapult-package/package/ <b>api</b> /catapult/
PEER	mijin-catapult-package/package/peer/catapult/

#### API ノードの場合は、以下のように移動します。

catapult@api1:~\$ cd mijin-catapult-package/package/api/catapult/ catapult@api1:~/mijin-catapult-package/package/api/catapult\$

#### 3.2.2.4 Step.4

mijin では docker による複数コンテナを起動しています。 複数のコンテナを確認します。

#### API ノードの場合

4つのコンテナが起動しています。全ての State が Up であることを確認してください。何かしら異常が ある場合は、Up`ではなく`Exit になり、ダウンした状態になります。

Name	Command	State	Ports
catapult_api-node-broker_1	<pre>bash -c /bin/bash /scripts bash -a park /acripts/wait</pre>	Up	0 0 0 0.7800->
↔7900/tcp	bash -c peri /scripts/wait	бЪ	0.0.0.0.7900->
catapult_db_1	docker-entrypoint.sh bash	Up	27017/tcp
catapult_rest-gateway_1 ⇔3000/tcp	docker-entrypoint.sh ash	Up	0.0.0.3000->

#### PEER ノードの場合

1つのコンテナが起動しています。 全ての State が Up であることを確認してください。 何かしら異常がある場合は、Up`ではなく`Exit になり、ダウンした状態になります。

#### 3.2.2.5 Step.5

docker-compose を使って、mijin コンテナすべてを停止します。

## コンテナ全停止

```
catapult@api1:~/mijin-catapult-package/package/api/catapult$ docker-compose down
Stopping catapult_rest-gateway_1 ... done
Stopping catapult_db_1 ... done
Removing catapult_rest-gateway_1 ... done
Removing catapult_rest-gateway_1 ... done
Removing catapult_api-node_1 ... done
Removing catapult_db_1 ... done
Removing catapult_db_1 ... done
Removing catapult_db_1 ... done
Removing catapult_db_1 ... done
```

## 停止後、ロックファイルを確認し、あれば削除する

# 上記のように停止後にロックファイルが存在する場合、異常停止したと考えられます。 そのため、ロックファイルを削除します。

\$ rm -rf /mnt/mijin/blocks/data/broker.lock /mnt/mijin/blocks/data/server.lock

#### 3.2.2.6 Step.6

docker-compose ファイルを修正します。 ここでは例として、**1.0.0.0** から **1.0.0.1** へ置き換えます。

```
catapult@api1:~/mijin-catapult-package/package/api/catapult$ sed -i -e s/gcc-1.0.0.0/
```

## 注釈:

Config ファイルの修正が必要な場合があります。 バージョンアップのアナウンス時に手順を公開します。(2022/9/20 時点バージョンなし)

#### 3.2.2.7 Step.7

```
catapult@api1:~/mijin-catapult-package/package/api/catapult$ docker-compose up -d
Creating network "catapult_default" with the default driver
Creating catapult_db_1 ... done
Creating catapult_rest-gateway_1 ... done
Creating catapult_api-node-broker_1 ... done
Creating catapult_api-node_1 ... done
```

起動後、Step.4 の項目を実行し、すべてのコンテナが Up になっていることを確認してください。

# 3.2.3 [アーカイブ] mijin アカウント作成 (>=1.0.0.0)

本章では、mijin を操作するために、必ず必要となるアカウント操作について説明します。

## 警告:

symbol-cli は アーカイブ化 されたため、symbol-cli は使用できない可能性があります。 1.0.3.4 以降は、mijin アカウント作成 を参照してください。

#### 注釈:

mijin Catapult(v.2) を操作するにはアカウントを作成し、そのアカウントを使ってトランザクションを送信します。

また、デプロイ時の指定で手数料ありモードの場合は、アカウントに基軸通貨 (cat.currency) の残高を持 つ必要があることに注意してください。

## 3.2.3.1 symbol-cli のインストール

npm より symbol-cli をインストールします。

```
$ sudo npm i -g symbol-cli@1.0.0
/usr/local/bin/symbol-cli -> /usr/local/lib/node_modules/symbol-cli/bin/symbol-cli
+ symbol-cli@1.0.0
updated 1 package in 8.724s
```

## 3.2.3.2 アカウントの作成

まずは、テストで使用する2つのアカウントを作成してみましょう。

項目	説明	值
Select the network type		MIJIN/MIJIN_TEST
	ネットワークを指定して下さい。 構築時に指定した CatapultNetwork の 値	
Do you want to save the account?	このアカウントを保存します。	yes
Select an import type	再度保存するためのインポート方法を指   定します。	PrivateKey
Enter the Symbol node URL.	Cloudformation Stack 内 に あ る Outouts タ ブ の mijinLBENdpoint または mijinEndpoint の URL を指定し ます。	<http: xxxxxx:300=""></http:>
Enter a profile name	アカウントを呼び出すプロファイル名を 指定します。	任意
Enter your wallet password	アカウントのパスワードを指定します	任意

# 一つ目のアカウント (Profile mijin-a) 作成

Value
MA36BR-7DCFZT-65BQZP-TM5QND-EZKSB7-HNE4DU-6TI
707902962A0A2E32226243D1E7B98D2DD40261E9D3649543E7C28A0F024D4A38
2515EDCAAA3985F30D6E758ED139823290DAB11034BF4113849FF5CB9355B9C9

Password

SUCCESS Stored mijin-a profile

Test1234

# この時点で mijin-a のアカウントを確認しても、存在しないことになっています。 ブロックチェーン上で、このアドレスのやり取りがないため、存在を確認できないからです。

## 2つ目のアカウント (Profile mijin-b) 作成

\$ symbol-cli account generate ✓ Select the network type: > MIJIN ✓ Do you want to save the account? ... yes ✓ Select an import type: > PrivateKey ✓ Enter the Symbol node URL. (Example: http://localhost:3000): ... http:// →xxxxxxxxxxxxxxx.elb.ap-northeast-1.amazonaws.com:3000 ✓ Enter a profile name: ... mijin-b ✓ Enter your wallet password: ... ******* ✓ Do you want to set the account as the default profile? ... no Account Property Value Address MCL063-LBWG6V-PLJD40-MADZ37-W6QXQE-DPC3H3-EGQ

Public Key	2D2AC0FF30FABEFC12CB3FBB2323F8CD079ED1055FAAF2581CA29697130292FA
Private Key	654065E33D00446F1FAAF2CF7D72CC287BDD91E55E9489AEC42769EDDB7A9759
Password	Test1234

SUCCESS Stored mijin-b profile

1つ目と同じで、mijin-b アカウントは存在しません。

# TIP The account has to receive at least one transaction to be recorded on the network

# 手数料について

手数料とは、以下の項目で発行するトランザクションによって手数料が発生し、基軸通貨 (cat.currency) にて支払う必要があります。

- トランザクション手数料
- Mosaic レンタル手数料
- Namespace レンタル手数料
- ・ロック保証金

マーケットプレイスで提供する mijin では手数料モードは標準で なしに設定されています。 手数料なしでは、アカウントに基軸通貨の残高を持つ必要がありませんので特に意識することはありませんが、<span style="color: #d61b09">symbol-cli や sdk にて手数料設定を都度 0 を設定する必要があります。</span>

手数料ありで mijin を作成した場合は、「手数料ありモード時の、基軸通貨の移動方法」の項目を参照し、 Profile mijin-a に基軸通貨を送信してください。

#### 3.2.3.3 Mosaic を作成し転送

まずは基本となる Mosaic を作成し、転送してみましょう。

#### Mosaic の作成

Profile mijin-a のアカウントにて 1amount の Mosaic(Token) を発行します。

項目	説明	値
Enter your wallet password	設定したパスワードを指定してください	任意
Do you want a non-expiring mosaic	Mosaic の期限を指定するか、無期限か を指定します。Yes で無期限にします。	yes
Enter the mosaic divisibility	Mosaic の可分性を指定します。0 でなし にします。	0
Do you want this mosaic to have a mutable supply?	Mosaic の最大発行数を変更できるよう にするかを指定します。なしにします。	no
Do you want this mosaic to be transferable?	Mosaic の転送を許可するか指定します。	yes
Do you want this mosaic to be restrictable?	Mosaic の制限を許可するか指定します。	yes
Amount of mosaics units to create	Mosaic の発行数を指定します。ここで は1で発行します、	1
Enter the maximum fee (absolute amount)	トランザクション手数料を指定します。 これは手数料モードで変わります。 手数料あり 20000 程度 (0.2cat.currency) 手数料なし 0	0
Select the transaction announce mode	トランザクションをアナウンスする方法 を指定します。	normal

## Listing 1: Symbol CLI によるモザイク定義

1	\$	symbol-cli transaction mosaicprofile mijin-a
2		
3	$\checkmark$	Enter your wallet password: … *******
4	√	Do you want a non-expiring mosaic? … yes
5	$\checkmark$	Enter the mosaic divisibility: … 0
6	$\checkmark$	Do you want this mosaic to have a mutable supply? no
7	$\checkmark$	Do you want this mosaic to be transferable? … yes
8	$\checkmark$	Do you want this mosaic to be restrictable? … yes
9	$\checkmark$	Amount of mosaics units to create: 1
10	$\checkmark$	Enter the maximum fee (absolute amount): 0
11	$\checkmark$	Select the transaction announce mode: > normal
12	$\checkmark$	Do you want to announce this transaction? … yes
13		
14	SU	CCESS Transaction announced correctly
15		
16	ΤI	P To check <b>if</b> the network confirms or rejects the transaction, run the command
	$\hookrightarrow$	'symbol-cli transaction status'

再度 `mijin-a`のアカウントを確認すると、Balance Information にて、MosaicId 3BF3AF8B22CB53D8 を 1 だけ持っていることが確認できます。

ļ	Property	Value			
	Address	MA36BR-7DCFZT-65	BQZP-TM5QND-EZKSB7	-HNE4DU-6TI	
	Address Height	959			
	Public Key	   707902962A0A2E32	226243D1E7B98D2DD4	0261E9D3649543E7C28A	0F024D4A38
	Public Key Height	959			
	Importance	0			
	Importance Height	0			
В	Balance Information				
	Mosaic Id	Relative Amount	Absolute Amount	Expiration Height	
	3BF3AF8B22CB53D8	1	1	Never	
					-

# Mosaic を転送する

先程作成した Mosaic 3BF3AF8B22CB53D8 を Profile mijin-a から mijin-b に転送します。

項目	値
Mosaic	3BF3AF8B22CB53D8
転送 amount	1
転送先アドレス (Profile mijin-b のアドレス)	MCLO63-LBWG6V-PLJD4O-MADZ37-W6QXQE-DPC3H3-EGQ

項目	説明	值
Enter your wallet password	設定したパスワードを指定してください	任意
Mosaics to transfer in the format (mosaicId(hex) @aliasName)::absolu	転送するモザイク ID(またはエイリアス)にコ It <b>eArbroをuit</b> つ追加した後、転送 amount を指定しま す。	3BF3AF8B22CB53D8::1
Enter the recipient address or @alias	転送先アドレスを指定します	MCLO63-LBWG6V-PLJD4O- MADZ37-W6QXQE-DPC3H3-EGQ
Enter a message	転送トランザクションにメッセージを追加できま す	任意
Enter the maximum fee (absolute amount)	トランザクション手数料を指定します(モードに より異なります)。 - 手数料あり: 約 ^{、、} 20000、、(= 0.2cat.currency) - 手数料なし: 0	0
Select the transaction announce mode	トランザクションをアナウンスする方法を指定し ます。	normal

```
$ symbol-cli transaction transfer --profile mijin-a
```

```
? Enter your wallet password: > (node:97) [DEP0091] DeprecationWarning: crypto.DEFAULT_
```

→ENCODING is deprecated. (node:97) [DEP0010] DeprecationWarning: crypto.createCredentials is deprecated. Use tls.createSecureContext instead. (node:97) [DEP0011] DeprecationWarning: crypto.Credentials is deprecated. Use tls. →SecureContext instead. ✓ Enter your wallet password: … ******* ✓ Mosaics to transfer in the format (mosaicId(hex)|@aliasName)::absoluteAmount, (Ex: sending 1 symbol.xym, @symbol.xym::1000000). Add multiple mosaics separated by commas: ··· 3BF3AF8B22CB53D8::1 ✓ Enter the recipient address or @alias: … MCLO63-LBWG6V-PLJD40-MADZ37-W6QXQE-DPC3H3-⇔EGQ ✓ Enter a message: … ✓ Enter the maximum fee (absolute amount): ··· 0 ✓ Select the transaction announce mode: > normal TRANSFER Max fee: 0 Network type: MIJIN Deadline: 2021-05-17 16:40:45.212 Recipient: MCLO63-LBWG6V-PLJD4O-MADZ37-W6QXQE-DPC3H3-EGQ Message: N/A Mosaic (1/1): 1 3BF3AF8B22CB53D8 Signature details B000000000000000639B3F893989DC69FA3DF8D9BA294FD787F7644D68918FA8 Payload: F5E6A5162FE57CD34B03E944AA48F4B22790A50ECEA4A130EBB89299BDB173A4 556BBAFF8092A20B707902962A0A2E32226243D1E7B98D2DD40261E9D3649543 E7C28A0F024D4A38000000001605441000000000000000000000C1F80300E000000 6096EF6D61B1BD57AD23E39801E77FB7A178106F16CFB21A000001000000000 D853CB228BAFF33B0100000000000000 Hash: 0CAB966B0E7090AA19AE4D4F2BD2334A7F7466E5661107A15F8831EA48A5CE88 Signer: 707902962A0A2E32226243D1E7B98D2DD40261E9D3649543E7C28A0F024D4A38  $\checkmark$  Do you want to announce this transaction?  $\cdots$  yes SUCCESS Transaction announced correctly TIP To check **if** the network confirms or rejects the transaction, run the command →'symbol-cli transaction status'

# Profile **mijin-a** の状態を確認します。 Mosaic **3BF3AF8B22CB53D8** を持っていないことを確認できます。

symbol-cli account info --profile mijin-a

	(continued non previous page)
" Processing(node:10 deprecated.	8) [DEP0091] DeprecationWarning: crypto.DEFAULT_ENCODING is
(node:108) [DEP0010]	DeprecationWarning: crypto.createCredentials is deprecated. Use
<pre>tls.createSecureConto (node:108) [DEP0011] →SecureContext insto : Processing Account Information</pre>	ext instead. DeprecationWarning: crypto.Credentials is deprecated. Use tls. ead.
Property	Value
Address	MA36BR-7DCFZT-65BQZP-TM5QND-EZKSB7-HNE4DU-6TI
Address Height	959
Public Key	707902962A0A2E32226243D1E7B98D2DD40261E9D3649543E7C28A0F024D4A38
Public Key Height	959
Importance	
Importance Height	0

# 続いて、Profile **mijin-b** の状態を確認します。 Mosaic **3BF3AF8B22CB53D8** を所有していることが確認できます。

<pre>symbol-cli account infoprofile mijin-b " Processing(node:119) [DEP0091] DeprecationWarning: crypto.DEFAULT_ENCODING is deprecated. (node:119) [DEP0010] DeprecationWarning: crypto.createCredentials is deprecated. Use tls.createSecureContext instead. (node:119) [DEP0011] DeprecationWarning: crypto.Credentials is deprecated. Use tls.</pre>		
Property	Value	
Address	MCLO63-LBWG6V-PLJD4O-MADZ37-W6QXQE-DPC3H3-EGQ	
Address Height	17912	
Public Key		
Public Key Height	0	
Importance	0	
Importance Height	0	
Balance Information	(continues on next page)	

Mosaic Id	Relative Amount	Absolute Amount	Expiration Height
3BF3AF8B22CB53D8	1	1	Never

#### 3.2.3.4 (番外)手数料ありモード時の、基軸通貨の移動方法

手数料ありモードで mijin を作成した場合、トランザクション送信などに全て基軸通貨から手数料が取ら れるようになります。

そのため、アカウント作成時に基軸通貨の残高を送信する手順を説明します。

## 基軸通貨の保持アカウント登録

## 注釈:

AWS Marketplace の | mijin | を使用した場合、AWS Cloudformation Stack 内にある Outouts タブを選択 します。

Key「mijinLBEndpint」又は「mijinbEndpoint」の URL を控えてください。

次に、Key「HarvestAddress」の Value の URL をクリックしてください。

api の中にある一番はじめの Vaule にある private_key の値を控えてください。(ここでは 055E で始まる 値)

詳しくは、既存 VPC 上に、mijin をデプロイする 又は新規 VPC を作成し、mijin をデプロイする を参照し てください。

## symbol-cli にて上記の private_key を import します。

項目	入力値
Select the network type	
	ネットワークを指定して下さい。
	構築時に指定した CatapultNetwork の値
	MIJIN <b>または</b> MIJIN_TEST
Enter the Symbol hode URL	
	控えていた mijinEndpoint もしくは
	ロードバランサーが有効であれば mijinLBEndpoint の URL を入力
Enter a profile name	任意のプロファイル名を入力
Enter your wallet password	任意のパスワードを入力
Do you want to set the account as	DefaultProfile にするかどうか、ここでは Yes を選択
the default profile	
Select an import type	PrivateKey
Enter your account private key	控えていた private_key を入力

\$ symbol-cli profile import

✓ Select the network type: > MIJIN
✓ Enter the Symbol node URL. (Example: http://localhost:3000): … http://MIJIN-→ CATAPULT-E1-nlb-rest-XXXXXXX.elb.ap-northeast-1.amazonaws.com:3000

$\checkmark$	Enter a profile name: … mijin-harvest
$\checkmark$	Enter your wallet password: … *******
$\checkmark$	Do you want to set the account as the default profile? … no
$\checkmark$	Select an import type: > PrivateKey
$\checkmark$	Enter your account private key: …
* *	***************************************
Ac	count

Property	Value
Address	MAQUY5-KOJVPE-DDCTD6-3SZYHM-EQOFF4-HTUYZU-3WQ
Public Key	29800CB9DF988622AD4B940F578569321F4B7F08127C478A0C0C28ACC61B8A2C
Private Key	2EC8FF52B5B922E0F509FBEE6CE3C4B3512E9347DB800A76A6EF993C43C0D5BC
Password	Test1234
L	

SUCCESS Stored mijin-harvest profile

これでインポートすることができました。 アカウントの情報を確認します。 このアカウントでは、初期時 2 つの Mosaic を持っています。 Balance Information の 04A125F887094D2A が基軸通貨 (cat.currency) となり、49DB43B9FA374EF2 は Harvest 有効用の Mosaic です。

## 警告: この ID は作成毎に違う値になり、基軸通貨は Amount が大きい ID となります。

\$ symbol-cli account infoprofile mijin-harvest		
" Processing(node:141) [DEP0091] DeprecationWarning: crypto.DEFAULT_ENCODING is		
deprecated.		
(node:141) [DEP0010]	DeprecationWarning: crypto.createCredentials is deprecated. Use	
tls.createSecureConte	ext instead.	
(node:141) [DEP0011]	DeprecationWarning: crypto.Credentials is deprecated. Use tls.	
→SecureContext inst	ead.	
: Processing		
Account Information		
r		
Property	Value	
Address	MAQUY5-KOJVPE-DDCTD6-3SZYHM-EQOFF4-HTUYZU-3WQ	
Address Height	1	
Public Key	29800CB9DF988622AD4B940F578569321F4B7F08127C478A0C0C28ACC61B8A2C	
   Public Key Height	1	

Importance	2850000	2850000		
Importance Height	18080	18080		
Balance Information				
Mosaic Id	Relative Amount	Absolute Amount	   Expiration Height	
04A125F887094D2A	1,799,799,999.6	1799799999600000	Never	
49DB43B9FA374EF2	3,000	3000000	Never	

## 基軸通貨の確認

項目	説明	值
Enter the mosaic id in hexadecimal format	モザイク ID を指定することで対象モザ イクの情報を取得できます	ここでは 04A125F887094D2A

# Divisibility にて可分性が確認できます。 以下では可分性が6となります。

<pre>symbol-cli mosaic in: ? Enter the mosaic id crypto.DEFAULT_ENCOD (node:163) [DEP0010] tls.createSecureContex (node:163) [DEP0011] →SecureContext inste ✓ Enter the mosaic i Mosaic Information</pre>	fo d <b>in</b> hexadecimal format: > (node:163) [DEP0091] D ING is deprecated. DeprecationWarning: crypto.createCredentials is ext instead. DeprecationWarning: crypto.Credentials is deprec ead. d <b>in</b> hexadecimal format: 04A125F887094D2A	eprecationWarning: deprecated. Use ated. Use tls.
Property	Value	
Record Id	609DEB554D4B851AA429AE2C	
Mosaic Id	04A125F887094D2A	
Divisibility	6	
Transferable	true	
Supply Mutable	false	
Height	1	
		(continues on next nage)

Owner     MAO5AR-GSMLGK-ZCDV35-IJWDVL-JFOCZT-XHM3KJ-RHA       Supply (Absolute)     8998999998000000       Supply (Belative)     8 998 999 998	Expiration		Never
Supply (Absolute) 8998999998000000	Owner		MAO5AR-GSMLGK-ZCDV35-IJWDVL-JFOCZT-XHM3KJ-RHA
Supply (Relative) 8 998 999 998	Supply	(Absolute)	8998999998000000
	Supply	(Relative)	8,998,999,998

## 基軸通貨を送信する

登録したアカウント (Profile mijin-a) に対して、基軸通貨を 1000cat.currency を送信してみます。

項目	值
Mosaic	cat.currency
転送 amount	100000000(可分性が6なので 1000.000000)
転送先アドレス (Profile mijin-a)	MA36BR-7DCFZT-65BQZP-TM5QND-EZKSB7-HNE4DU-6TI

項目	説明	值
Enter your wallet password	設定したパスワードを指定してください	任意
Mosaics to transfer in the format (mosaicId(hex) @aliasName)::absoluteA	転送するモザイク ID(またはエイリア moountにコロンを二つ追加した後、転送 amountを指定します。	@cat.currency::1000000000
Enter the recipient address or @alias	転送先アドレスを指定します	MA36BR-7DCFZT-65BQZP-TM5QND- EZKSB7-HNE4DU-6TI
Enter a message	転送トランザクションにメッセージを追 加できます	任意
Enter the maximum fee (absolute amount)	トランザクション手数料を指定します。 これは手数料モードで変わります。 手数料あり 20000 程度 (0.2cat.currency) 手数料なし 0	0
Select the transaction announce mode	トランザクションをアナウンスする方法 を指定します。	normal

ここではトランザクション手数料を0にしていますが、手数料ありの場合、トランザクション手数料がか かりますので0ではなく、200000程度必要になります。 手数料の計算方法は、Symbol と同様となります。

<https://docs.symbol.dev/concepts/fees.html#transaction-fee>

	<pre>→SecureContext instead. ✓ Enter your wallet password: … ******* ✓ Mosaics to transfer in the format (mosaicId(hex) @aliasName)::absoluteAmount, (Ex: sending 1 symbol.xym, @symbol.xym::1000000). Add multiple mosaics separated by commas: … @cat.currency::1000000000 ✓ Enter the recipient address or @alias: … MA36BR-7DCFZT-65BQZP-TM5QND-EZKSB7-HNE4DU- → 6TI ✓ Enter a message: … ✓ Enter the maximum fee (absolute amount): … 0 ✓ Select the transaction announce mode: &gt; normal</pre>				
	Max fee:	0			
	Network type:	MIJIN			
	Deadline:	2021-05-17 17:46:42.643			
Recipient: MA36BR-7DCFZT-65BQZP-TM5QND-EZKSB7-HNE4DU-6TI					
Message: N/A					
	Mosaic (1/1): 1,000,000 cat.currency (85BBEA6CC462B244)				
	Signature details				
Payload:         B000000000000000000000000000000000000					
	Hash:	864EF99D58E8DA837879D85DE08DF29398766E04F967F09732A2FD02115469FB			
	Signer:	29800CB9DF988622AD4B940F578569321F4B7F08127C478A0C0C28ACC61B8A2C			
✓ Do you want to announce this transaction? … yes SUCCESS Transaction announced correctly					
TIP To check if the network confirms or rejects the transaction, run the command $\rightarrow$ 'symbol-cli transaction status'					

# Profile mijin-a のアカウントを確認します。 1000cat.currency を持っていることを確認できます。

symbol-cli account info --profile mijin
" Processing(node:207) [DEP0091] DeprecationWarning: crypto.DEFAULT_ENCODING is
deprecated.
(node:207) [DEP0010] DeprecationWarning: crypto.createCredentials is deprecated. Use
tls.createSecureContext instead.

<pre>(node:207) [DEP0011] DeprecationWarning: crypto.Credentials is deprecated. Use tls. →SecureContext instead. i Processing Account Information</pre>				
Property	Value			
Address	MA36BR-7DCFZT-65	MA36BR-7DCFZT-65BQZP-TM5QND-EZKSB7-HNE4DU-6TI		
Address Height	959			
Public Key	707902962A0A2E32226243D1E7B98D2DD40261E9D3649543E7C28A0F024D4A38			
Public Key Height	959			
Importance	0			
Importance Height	0			
Balance Information				
Mosaic Id	Relative Amount	Absolute Amount	Expiration Height	
04A125F887094D2A	1,000	100000000	Never	

# 3.3 トラブルシューティング

mijin Catapult(v.2)のトラブルシューティングを纏めます。

# 3.3.1 mijin Catapult(v.2) のノード再同期

mijin Catapult(v.2) ノードが何らかの理由で同期が止まってしまったなどの場合の再同期手順を説明します。

# 3.3.1.1 対象

- ノードのブロックが進まない
- /node/peers で確認できなくなった
- docker-compose ps コマンドで一部コンテナが Exit になっており、復旧できない。

## 3.3.1.2 Step.1

ノードにリモートログインします。 AWS MarketPlace の mijin Catapult(v.2) にログインしたい場合は、mijin Catapult(v.2) EC2 インスタンス ログイン方法 を参照してください。

## 3.3.1.3 Step.2

mijin を起動している「catapult」ユーザーにスイッチします。

```
$ sudo su - catapult
catapult@api1:~$
```

#### 3.3.1.4 Step.3

mijin の起動ファイルがあるディレクトリに移動します。 API,PEER ノードでディレクトリが違うことに注意してください。

ノード	ディレクトリ
API/Dual	mijin-catapult-package/package/ <b>api</b> /catapult/
PEER	mijin-catapult-package/package/ <b>peer</b> /catapult/

API ノードの場合は、以下のように移動します。

catapult@api1:~\$ cd mijin-catapult-package/package/api/catapult/ catapult@api1:~/mijin-catapult-package/package/api/catapult\$

#### 3.3.1.5 Step.4

mijin では docker による複数コンテナを起動しています。 複数のコンテナを確認します。

#### API ノードの場合

4つのコンテナが起動しています。全ての State が Up であることを確認してください。何かしら異常が ある場合は、Up`ではなく`Exit になり、ダウンした状態になります。

catapult@api1:~/mijin-catapul	t-package/package/api/catapult\$	docker-co	mpose ps
Name	Command	State	Ports
catapult_api-node-broker_1	bash -c /bin/bash /scripts	Up	
catapult_api-node_1 ⇔7900/tcp	<pre>bash -c perl /scripts/wait</pre>	Up	0.0.0.0:7900->
catapult_db_1	docker-entrypoint.sh bash	Up	27017/tcp
catapult_rest-gateway_1 ⊶3000/tcp	docker-entrypoint.sh ash	Up	0.0.0.0:3000->

#### PEER ノードの場合

1つのコンテナが起動しています。 全ての State が Up であることを確認してください。 何かしら異常がある場合は、Up`ではなく`Exit になり、ダウンした状態になります。

#### 3.3.1.6 Step.5

docker-compose を使って、mijin コンテナすべてを停止し、再度起動します。

## コンテナ全停止

```
catapult@api1:~/mijin-catapult-package/package/api/catapult$ docker-compose down
Stopping catapult_rest-gateway_1 ... done
Stopping catapult_db_1 ... done
Removing catapult_api-node=broker_1 ... done
Removing catapult_rest-gateway_1 ... done
Removing catapult_api-node_1 ... done
Removing catapult_db_1 ... done
Removing catapult_db_1 ... done
Removing catapult_db_1 ... done
Removing catapult_db_1 ... done
```

## 停止後、ロックファイルを確認し、あれば削除する

```
catapult@api1:~/mijin-catapult-package/package/api/catapult$ ls -la /mnt/mijin/blocks/

→data/*.lock
------ 1 catapult catapult 0 Jul 14 02:17 /mnt/mijin/blocks/data/broker.lock
------ 1 catapult catapult 0 Jul 14 02:17 /mnt/mijin/blocks/data/server.lock
```

# 上記のように停止後にロックファイルが存在する場合、異常停止したと考えられます。 そのため、ロックファイルを削除します。

\$ rm -rf /mnt/mijin/blocks/data/broker.lock /mnt/mijin/blocks/data/server.lock

再度起動

\$ docker-compose up -d Creating network "catapult_default" with the default driver Creating catapult_db_1 ... done Creating catapult_rest-gateway_1 ... done Creating catapult_api-node-broker_1 ... done Creating catapult_api-node_1 ... done

起動後、Step.4 の項目を実行し、すべてのコンテナが Up になっていることを確認してください。 もし、同様な状態になっている場合は、Step.9 に進んでください。

#### 3.3.1.7 Step.6

ノードのブロックデータをリセットし復旧します。冗長化されている状態であれば、データは自動で他ノ ードから取得し復旧することができます。

## コンテナ全停止

```
$ docker-compose down
Stopping catapult_rest-gateway_1 ... done
Stopping catapult_api-node_1 ... done
Removing catapult_api-node-broker_1 ... done
Removing catapult_rest-gateway_1 ... done
Removing catapult_api-node_1 ... done
Removing catapult_db_1 ... done
Removing network catapult_default
```

#### ブロックデータ及び mongo データを削除

mijin が起動している catapult ユーザーには sudo 権限がなく、ディレクトリを削除することができません。 そのため、catapult ユーザーに sudo 権限を付与します。 この sudo 設定は初回のみ必要です。

```
catapult ユーザーからログアウト Log out of catapult user
$ logout
root ユーザーにスイッチ Switch to root user
$ sudo su -
catapult ユーザーに sudo 権限を付与する Grant sudo privileges to the catapult user
echo "catapult ALL=(ALL) NOPASSWD:ALL" > /etc/sudoers.d/catapult
catapult ユーザーにスイッチ Switch to catapult user
su - catapult
```

sudo 権限でブロックデータを削除します。

\$ sudo rm -rf /mnt/mijin/blocks/data

#### API ノードが対象の場合は、mongo データの削除も必要です

\$ sudo rm -rf /mnt/mijin/mongo/db

## リカバリーコマンドの実行

docker-compose のあるファイルに移動し、recover スクリプトを実行します。

#### API ノードが対象の場合

catapult@api1:~\$ cd mijin-catapult-package/package/api/catapult/ catapult@api1:~/mijin-catapult-package/package/api/catapult\$

#### PEER ノードが対象の場合

```
catapult@peer1:~$ cd mijin-catapult-package/package/peer/catapult/
catapult@peer1:~/mijin-catapult-package/package/peer/catapult$
```

#### スクリプトの実行をします。(共通)

```
catapult@api1:~/mijin-catapult-package/package/api/catapult$ bash scripts/recover.sh
2021/07/14 02:52 Start: mijin Recovery
2021/07/14 02:52 Check: /home/catapult/mijin-catapult-package/package/api/catapult
2021/07/14 02:52 Check: /home/catapult/mijin-catapult-package/package/api/catapult OK
2021/07/14 02:52 Check: Started docker?
2021/07/14 02:52 Check: Started docker Stop OK
2021/07/14 02:52 Check: Block Directory
2021/07/14 02:52 Check: Block Directory Empty OK
2021/07/14 02:52 Start: Make Block Directory
2021/07/14 02:52 Check: mongo Directory
2021/07/14 02:52 Check: mongo Directory OK
2021/07/14 02:52 Start: Make mongo Directory
2021/07/14 02:52 Start: Create mongo Init Data
about to fork child process, waiting until server is ready for connections.
forked process: 10
child process started successfully, parent exiting
[+] Preparing db
MongoDB shell version v4.2.5
connecting to: mongodb://localhost:27017/catapult?compressors=disabled&
→gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("c296d026-dfdf-4ea2-ba03-91d7404e21c9") }
MongoDB server version: 4.2.5
Loading LockHash
Loading LockSecret
Loading Metadata
Loading Mosaic
Loading Multisig
Loading Namespace
Loading RestrictionAccount
Loading RestrictionMosaic
===== accountRestrictions INDEXES =====
{ "_id" : 1 }
{ "accountRestrictions.address" : 1 }
===== accounts INDEXES =====
```

```
{ "_id" : 1 }
{ "account.publicKey" : 1 }
{ "account.address" : 1 }
===== addressResolutionStatements INDEXES =====
{ "_id" : 1 }
{ "statement.height" : 1, "statement.unresolved" : 1 }
===== blocks INDEXES =====
{ "_id" : 1 }
{ "block.signerPublicKey" : 1 }
{ "block.timestamp" : -1 }
{ "block.height" : -1 }
{ "block.type" : 1, "block.height" : -1 }
{ "block.signerPublicKey" : 1, "block.height" : -1 }
{ "block.beneficiaryAddress" : 1, "block.height" : -1 }
===== finalizedBlocks INDEXES =====
{ "_id" : 1 }
{ "block.finalizationEpoch" : -1 }
{ "block.height" : -1 }
===== hashLocks INDEXES =====
{ "_id" : 1 }
{ "lock.hash" : 1 }
{ "lock.ownerAddress" : 1 }
===== metadata INDEXES =====
{ "_id" : 1 }
{ "metadataEntry.compositeHash" : 1 }
{
 "metadataEntry.sourceAddress" : 1,
 "metadataEntry.metadataType" : 1,
 "metadataEntry.scopedMetadataKey" : 1
}
{
 "metadataEntry.targetAddress" : 1,
 "metadataEntry.metadataType" : 1,
 "metadataEntry.scopedMetadataKey" : 1
}
===== mosaicResolutionStatements INDEXES =====
{ "_id" : 1 }
{ "statement.height" : 1, "statement.unresolved" : 1 }
===== mosaicRestrictions INDEXES =====
{ "_id" : 1 }
{ "mosaicRestrictionEntry.compositeHash" : 1 }
===== mosaics INDEXES =====
{ "_id" : 1 }
{ "mosaic.id" : 1 }
{ "mosaic.ownerAddress" : 1 }
===== multisigs INDEXES =====
{ "_id" : 1 }
{ "multisig.accountAddress" : 1 }
===== namespaces INDEXES =====
{ "_id" : 1 }
{ "namespace.level0" : 1, "meta.index" : 1, "namespace.depth" : 1 }
{
 "meta.latest" : -1,
 "meta.index" : 1,
 "namespace.level0" : 1,
 "namespace.depth" : 1
```

```
(continued from previous page)
```

```
{ "meta.latest" : -1, "namespace.level1" : 1, "namespace.depth" : 1 }
{ "meta.latest" : -1, "namespace.level2" : 1, "namespace.depth" : 1 }
{ "meta.latest" : -1, "namespace.ownerAddress" : 1 }
===== partialTransactions INDEXES =====
{ "_id" : 1 }
{ "transaction.signerPublicKey" : 1, "_id" : -1 }
{ "transaction.recipientAddress" : 1, "_id" : -1 }
{ "meta.hash" : 1 }
{ "meta.addresses" : 1 }
{ "meta.aggregateId" : 1 }
{ "meta.aggregateHash" : 1 }
===== secretLocks INDEXES =====
{ " id" : 1 }
{ "lock.compositeHash" : 1 }
{ "lock.ownerAddress" : 1 }
===== system.profile INDEXES =====
===== transactionStatements INDEXES =====
{ "_id" : 1 }
{
 "statement.height" : 1,
 "statement.source.primaryId" : 1,
 "statement.source.secondaryId" : 1
}
===== transactionStatuses INDEXES =====
{ "_id" : 1 }
{ "status.hash" : 1 }
{ "status.deadline" : -1 }
===== transactions INDEXES =====
{ "_id" : 1 }
{ "transaction.signerPublicKey" : 1, "_id" : -1 }
{ "transaction.recipientAddress" : 1, "_id" : -1 }
{ "meta.hash" : 1 }
 "meta.addresses" : 1 }
{ "meta.aggregateId" : 1 }
{ "meta.height" : -1 }
{ "transaction.deadline" : -1 }
{ "transaction.cosignatures.signerPublicKey" : 1 }
{ "transaction.id" : 1, "transaction.type" : 1 }
===== unconfirmedTransactions INDEXES =====
{ "_id" : 1 }
{ "transaction.signerPublicKey" : 1, "_id" : -1 }
{ "transaction.recipientAddress" : 1, "_id" : -1 }
{ "meta.hash" : 1 }
{ "meta.addresses" : 1 }
 "meta.aggregateId" : 1 }
{ "meta.aggregateHash" : 1 }
bve
[.] (exit code: 0)
/
[+] db prepared, checking account indexes
MongoDB shell version v4.2.5
connecting to: mongodb://localhost:27017/catapult?compressors=disabled&
→qssapiServiceName=mongodb
Implicit session: session { "id" : UUID("a9d9d654-d0c9-4760-bf52-94138f4c6871") }
MongoDB server version: 4.2.5
[
```
```
"v" : 2,
 "key" : {
 "_id" : 1
 },
 "name" : "_id_",
 "ns" : "catapult.accounts"
 },
 {
 "v" : 2,
 "key" : {
 "account.publicKey" : 1
 },
 "name" : "account.publicKey_1",
 "ns" : "catapult.accounts"
 },
 {
 "v" : 2,
 "unique" : true,
 "key" : {
 "account.address" : 1
 },
 "name" : "account.address_1",
 "ns" : "catapult.accounts"
 }
2021-07-14T02:52:54.345+0000 I CONTROL [main] Automatically disabling TLS 1.0, to
force-enable TLS 1.0 specify --sslDisabledProtocols 'none'
2021-07-14T02:52:54.349+0000 W ASIO [main] No TransportLayer configured during
NetworkInterface startup
killing process with pid: 10
2021/07/14 02:52 Start: mijin docker-compose
Creating network "catapult_default" with the default driver
Creating catapult_db_1 ... done
Creating catapult_api-node-broker_1 ... done
Creating catapult_rest-gateway_1 ... done
Creating catapult_api-node_1
 ... done
2021/07/14 02:52 End: ALL Success
```

#### 3.3.1.8 Step.7

動作を確認します。

### コンテナの動作確認

State がすべて Up であることを確認してください。

\$ docker-compose ps Name	Command	State	Ports
<pre> catapult_api-node-broker_1 catapult_api-node_1</pre>	bash -c /bin/bash /scripts bash -c perl /scripts/wait	Up Up	0.0.0.0:7900->

catapult_db_1	docker-entrypoint.sh bash	Up	27017/tcp
catapult_rest-gateway_1	docker-entrypoint.sh ash	Up	0.0.0:3000->
-→3000/tcp			

### ブロックがすすんでいるかを確認

ブロックが積み上がっているか確認します。 他のノードでも実行し、同じブロックで進んでいるかを確認します。

## API ノードの場合

### PEER ノードの場合

\$ docker-compose logs --tail=20 peer-node| grep heights peer-node_1 | 2021-07-14 03:00:26.168343 0x00007fb1c396a700: <debug> (disruptor::Disruptor.cpp043) disruptor queuing element 942 (1 blocks (heights 403 -403) [0000000] from Remote_Push with size 376B) peer-node_1 | 2021-07-14 03:00:26.257224 0x00007fb1a57fa700: <info> (disruptor::ConsumerDispatcher.cpp044) completing processing of element 942 (1 blocks (heights 403 - 403) [5C675B6B] empty from Remote_Push with size 376B), last consumer is 0 elements behind peer-node_1 | 2021-07-14 03:00:26.317777 0x00007fb1c3169700: <debug> (disruptor::Disruptor.cpp043) disruptor queuing element 943 (1 blocks (heights 403 -403) [0000000] from Remote_Push with size 376B) peer-node_1 | 2021-07-14 03:00:26.368263 0x00007fb1a57fa700: <info> (disruptor::ConsumerDispatcher.cpp044) completing processing of element 943 (1 blocks (heights 403 - 403) [5C675B6B] from Remote_Push with size 376B), last consumer is 0 elements behind

### API ノードの場合、rest が接続できるか確認

#### ノード情報が取得できるか確認します。

```
$ curl -Ss http://localhost:3000/node/info | jq -r
{
 "version": 16777216,
 "publicKey": "E4BF3706483B4D42243F3DCB2625021C3E3AE7C253CC466154EEDF9775012C20",
 "networkGenerationHashSeed":
 "2DE20B93EBE048A3BA132CC9874BCABBC21C87E18FE9836B8D5D002E57640D4B",
```

```
"roles": 70,
"port": 7900,
"networkIdentifier": 96,
"host": "api1.mijin.internal",
"friendlyName": "api1.mijin.internal",
"nodePublicKey": "27E7EEAF5819493D60CA848BAA48145A1A97DF63596ED41394563C791303C778"
```

## 3.3.2 mijin Catapult(v.2) のノードログを確認する

mijin Catapult(v.2) ノードのログを確認するまでの手順となります。

### 3.3.2.1 対象

- ノードのブロックが進まない
- エラーが出て原因がわからない

#### 3.3.2.2 Step.1

ノードにリモートログインします。 AWS MarketPlace の mijin Catapult(v.2) にログインしたい場合は、mijin Catapult(v.2) EC2 インスタンス ログイン方法 を参照してください。

#### 3.3.2.3 Step.2

mijin を起動している「catapult」ユーザーにスイッチします。

```
$ sudo su - catapult
catapult@api1:~$
```

#### 3.3.2.4 Step.3

mijin の起動ファイルがあるディレクトリに移動します。 API,PEER ノードでディレクトリが違うことに注意してください。

ノード	ディレクトリ
API/Dual	mijin-catapult-package/package/ <b>api</b> /catapult/
PEER	mijin-catapult-package/package/ <b>peer</b> /catapult/

#### API ノードの場合は、以下のように移動します。

```
catapult@api1:~$ cd mijin-catapult-package/package/api/catapult/
catapult@api1:~/mijin-catapult-package/package/api/catapult$
```

## 3.3.2.5 Step.4

mijin では docker による複数コンテナを起動しています。複数のコンテナを確認します。

### API ノードの場合

4つのコンテナが起動しています。 全ての State が Up であることを確認してください。

catapult@api1:~/mijin-catapult-package/package/api/catapult\$ Name Command			ompose ps Ports
↔			
catapult_api-node-broker_1	bash -c /bin/bash /scripts	Up	
catapult_api-node_1 ⇔7900/tcp	bash -c perl /scripts/wait	Up	0.0.0.0:7900->
catapult_db_1	docker-entrypoint.sh bash	Up	27017/tcp
catapult_rest-gateway_1 ⊶3000/tcp	docker-entrypoint.sh ash	Up	0.0.0.3000->

## PEER ノードの場合

1つのコンテナが起動しています。 全ての State が Up であることを確認してください。

catapult@peer1:~/mijin-catapult-package/package/peer/catapult\$ docker-compose ps					
Name Command		State	Ports		
catapult_peer-node_1	bash -c /bin/bash /scripts	Up	0.0.0.0:7900->7900/tcp		

#### 3.3.2.6 Step.5

ログを確認するのは docker-compose を使った以下のコマンドで確認できます。

docker-compose logs

docker-compose について詳しく知りたい場合は、以下のサイトをご確認ください。 https://docs.docker.com/compose/

以下では、確認する方法の例を示します。

## 各コンテナの直近 XX 行ログを確認します

docker-compose logs --tail=10

```
catapult@api1:~/mijin-catapult-package/package/api/catapult$ docker-compose logs --
→tail=10
Attaching to catapult_rest-gateway_1, catapult_api-node_1, catapult_api-node-broker_1,
catapult_db_1
api-node-broker_1 | 2021-06-14 00:41:32.447154 0x00007f59efb66700: <debug>
(subscribers::BrokerMessageReaders.h@90) preparing to process 1 messagesfrom /data/
⇔spool/block_change
api-node-broker_1 | 2021-06-14 00:41:32.447137 0x00007f59ef365700: <debug>
(subscribers::BrokerMessageReaders.h@90) preparing to process 2 messagesfrom /data/
→ spool/state_change
api-node-broker_1 | 2021-06-14 00:41:50.955992 0x00007f59efb66700: <debug>
(subscribers::BrokerMessageReaders.h@90) preparing to process 1 messagesfrom /data/
→spool/block_change
api-node-broker_1 | 2021-06-14 00:41:50.955975 0x00007f59ef365700: <debug>
(subscribers::BrokerMessageReaders.h@90) preparing to process 2 messagesfrom /data/
⇔spool/state_change
api-node-broker_1 | 2021-06-14 00:42:05.963816 0x00007f59efb66700: <debug>
(subscribers::BrokerMessageReaders.h@90) preparing to process 2 messagesfrom /data/
⇔spool/state_change
api-node-broker_1 | 2021-06-14 00:42:05.966239 0x00007f59ef365700: <debug>
(subscribers::BrokerMessageReaders.h@90) preparing to process 1 messagesfrom /data/
⇔spool/block_change
api-node-broker_1 | 2021-06-14 00:42:23.972546 0x00007f59efb66700: <debug>
(subscribers::BrokerMessageReaders.h@90) preparing to process 2 messagesfrom /data/
→spool/state_change
api-node-broker_1 | 2021-06-14 00:42:23.972928 0x00007f59ef365700: <debug>
(subscribers::BrokerMessageReaders.h@90) preparing to process 1 messagesfrom /data/
→spool/block_change
api-node-broker_1 | 2021-06-14 00:42:41.982065 0x00007f59ef365700: <debug>
(subscribers::BrokerMessageReaders.h@90) preparing to process 1 messagesfrom /data/
→spool/block_change
api-node-broker_1 | 2021-06-14 00:42:41.982048 0x00007f59efb66700: <debug>
(subscribers::BrokerMessageReaders.h@90) preparing to process 2 messagesfrom /data/
⇔spool/state_change
 | 2021-06-14 00:42:41.523496 0x00007f1bda429700: <debug>
api-node_1
(chain::CompareChains.cpp@119) comparing chain scores: 7676859692801638166 (local) vs
7676974281840495032 (remote)
 | 2021-06-14 00:42:41.523569 0x00007f1bda429700: <debug>
api-node_1
(chain::CompareChains.cpp@145) comparing hashes with local height 67006, starting height
66976, max hashes 1440
 | 2021-06-14 00:42:41.529042 0x00007f1bdac2a700: <debug>
api-node_1
(chain::ChainSynchronizer.cpp@309) pulling blocks from remote with common height 67006
(fork depth = 0) from DEC1EF1767E76BC31DF2FDADC75C23F6FDA6ECCB22554E4F4C790F81F869F797
@ 10.0.3.199
api-node 1
 | 2021-06-14 00:42:41.618591 0x00007f1bdac2a700: <info>
(chain::ChainSynchronizer.cpp@217) peer returned 1 blocks (heights 67007 - 67007)
 | 2021-06-14 00:42:41.618764 0x00007f1bdac2a700: <debug>
api-node_1
(chain::ChainSynchronizer.cpp@223) completing chain synchronization with1 blocks (fork
depth = 0)
api-node_1
 | 2021-06-14 00:42:41.618805 0x00007f1bdac2a700: <debug>
(disruptor::Disruptor.cpp@43) disruptor queuing element 66554 (1 blocks (heights 67007
- 67007) [00000000] from Remote_Pull with size 376B)
 | 2021-06-14 00:42:41.618960 0x00007f1bdac2a700: <info>
api-node_1
```

(chain::RemoteApiForwarder.h@69) completed 'synchronizer task' (peer2.mijin.internal @ peer2.mijin.internal:7900) with result Success | 2021-06-14 00:42:41.676588 0x00007f1bb0ff9700: <debug> api-node_1 (cache::SupplementalDataStorage.cpp@32) wrote last recalculation height 67000 last finalized height 66976 dynamic fee multiplier 0 total transactions 34 (score = [0, 7676974281840495032], height = 67007) | 2021-06-14 00:42:41.714555 0x00007f1bb07f8700: <info> api-node_1 (disruptor::ConsumerDispatcher.cpp@44) completing processing of element 66554 (1 blocks (heights 67007 - 67007) [833B95DA] from Remote_Pull with size 376B), last consumer is 0 elements behind api-node_1 | 2021-06-14 00:42:50.247964 0x00007f1bdac2a700: <debug> (chain::RoundContext.cpp@89) not completable - Erv == g(Vrv) and descendant can reach g(Crv) (total weight 15000000, cumulative precommit weight 3000000) db_1 | 2021-06-02T09:37:24.913+0000 I NETWORK [listener] connection accepted from 172.20.0.9:47924 #16 (15 connections now open) db 1 | 2021-06-02T09:37:24.913+0000 I NETWORK [conn16] received client metadata from 172.20.0.9:47924 conn16: { driver: { name: "nodejs", version: "3.6.0" }, os: { type: "Linux", name: "linux", architecture: "x64", version: "5.4.0-1029-aws" }, platform: "'Node.js v12.18.1, LE (legacy)" } db_1 | 2021-06-02T09:39:05.968+0000 I NETWORK [listener] connection accepted from 172.20.0.9:47942 #17 (16 connections now open) db 1 | 2021-06-02T09:39:05.969+0000 I NETWORK [conn17] received client metadata from 172.20.0.9:47942 conn17: { driver: { name: "nodejs", version: "3.6.0" }, os: { type: "Linux", name: "linux", architecture: "x64", version: "5.4.0-1029-aws" }, platform: "'Node.js v12.18.1, LE (legacy)" } db 1 | 2021-06-02T09:42:43.866+0000 I NETWORK [listener] connection accepted from 172.20.0.9:47948 #18 (17 connections now open) | 2021-06-02T09:42:43.869+0000 I NETWORK [conn18] received client db 1 metadata from 172.20.0.9:47948 conn18: { driver: { name: "nodejs", version: "3.6.0" }, os: { type: "Linux", name: "linux", architecture: "x64", version: "5.4.0-1029-aws" }, platform: "'Node.js v12.18.1, LE (legacy)" } | 2021-06-08T05:37:00.594+0000 I NETWORK [listener] connection db_1 accepted from 172.20.0.9:58874 #19 (18 connections now open) | 2021-06-08T05:37:00.596+0000 I NETWORK [conn19] received client db 1 metadata from 172.20.0.9:58874 conn19: { driver: { name: "nodejs", version: "3.6.0" }, os: { type: "Linux", name: "linux", architecture: "x64", version: "5.4.0-1029-aws" }, platform: "'Node.js v12.18.1, LE (legacy)" } | 2021-06-12T02:52:21.305+0000 I NETWORK [listener] connection db_1 accepted from 172.20.0.9:37814 #20 (19 connections now open) db 1 | 2021-06-12T02:52:21.310+0000 I NETWORK [conn20] received client metadata from 172.20.0.9:37814 conn20: { driver: { name: "nodejs", version: "3.6.0" }, os: { type: "Linux", name: "linux", architecture: "x64", version: "5.4.0-1029-aws" }, platform: "'Node.js v12.18.1, LE (legacy)" } > node _build/index.js "/userconfig/rest.json" rest-gateway_1 rest-gateway_1 rest-gateway_1 [winston] Attempt to write logs with no transports {"message": ↔ "loading config from /userconfig/rest.json", "level": "info"} rest-gateway_1 | info: loading config from /userconfig/rest.json | verbose: finished loading rest server config { "network": { "name": rest-gateway_1 -- "mijin", "description": "mijin network"}, "port": 3000, "crossDomain": {"allowedHosts": ["* ←,"],"allowedMethods":["GET","POST","PUT","OPTIONS"]},"extensions":["accountLink", ↔ "baseRetryDelay":750}, "apiNode": { "host": "api-node", "port":7900, → "tlsClientCertificatePath": "/userconfig/resources/cert/node.crt.pem", → "tlsClientKeyPath": "/userconfig/resources/cert/node.key.pem", "tlsCaCertificatePath":

```
→ "/userconfig/resources/cert/ca.cert.pem", "timeout":1000, "networkPropertyFilePath":"/
→api-node-config/config-network.properties", "nodePropertyFilePath": "/api-node-config/
→config-node.properties"}, "websocket":{"mq":{"host":"api-node-broker", "port":7902,

-- "monitorInterval":500, "connectTimeout":10000, "monitorLoggingThrottle":60000},
→":{"formats":["colorize", "simple"], "level": "verbose", "handleExceptions":true}, "file":
→{"formats":["prettyPrint"],"level":"verbose","handleExceptions":true,"filename":
→ "catapult-rest.log", "maxsize":20971520, "maxFiles":100}}, "numBlocksTransactionFeeStats
→":300, "timestamp": "2021-06-02T09:27:19.336Z" }
rest-gateway_1
 | info: connecting to mongodb://db:27017/ (database:catapult) {

→ "timestamp": "2021-06-02T09:27:19.407Z"
}
rest-gateway_1
 | (node:24) DeprecationWarning: current Server Discovery and
Monitoring engine is deprecated, and will be removed in a future version. To use the new
Server Discover and Monitoring engine, pass option { useUnifiedTopology: true } to the
MongoClient constructor.
 | verbose: connected to mongo at mongodb://db:27017/catapult {
rest-gateway_1

witimestamp":"2021-06-02T09:27:19.443Z"

rest-gateway_1 | info: registering routes {"timestamp":"2021-06-02T09:27:19.445Z"}
rest-gateway_1
 | info: listening on port 3000 {"timestamp":"2021-06-02T09:27:19.4622
\rightarrow "}
```

## 指定したコンテナごとに確認します

コンテナのホスト名を指定することで、特定のコンテナに絞ってログを出力します。 ホスト名は固定されています。

- api-node
- peer-node
- api-node-broker
- db
- rest-gateway

docker-compose logs [コンテナ名 Container name]

```
catapult@api1:~/mijin-catapult-package/package/api/catapult$ docker-compose logs rest-
⇔gateway
Attaching to catapult_rest-gateway_1
rest-gateway_1
rest-gateway_1
 > catapult-api-rest@0.0.0 start /app/catapult-rest/rest
 | > node _build/index.js "/userconfig/rest.json"
rest-gateway_1
rest-gateway_1
 rest-gateway_1
 | [winston] Attempt to write logs with no transports {"message":

→"loading config from /userconfig/rest.json", "level":"info"}

 | info: loading config from /userconfig/rest.json
rest-gateway_1
rest-gateway_1
 verbose: finished loading rest server config {"network":{"name":
-- "mijin", "description": "mijin network"}, "port": 3000, "crossDomain": {"allowedHosts": ["*
→"],"allowedMethods":["GET","POST","PUT","OPTIONS"]},"extensions":["accountLink",
→"aggregate","lockHash","lockSecret","mosaic","metadata","multisig","namespace",
← "baseRetryDelay":750}, "apiNode": { "host": "api-node", "port":7900,
```

```
→ "tlsClientCertificatePath": "/userconfig/resources/cert/node.crt.pem",
→ "tlsClientKeyPath": "/userconfig/resources/cert/node.key.pem", "tlsCaCertificatePath":
→api-node-config/config-network.properties","nodePropertyFilePath":"/api-node-config/

config-node.properties"}, "websocket": { "mq": { "host": "api-node-broker", "port": 7902,
→ "monitorInterval":500, "connectTimeout":10000, "monitorLoggingThrottle":60000},
→":{"formats":["colorize", "simple"], "level": "verbose", "handleExceptions":true}, "file":
→{"formats":["prettyPrint"],"level":"verbose","handleExceptions":true,"filename":
rest-qateway_1
 | info: connecting to mongodb://db:27017/ (database:catapult) {

witimestamp":"2021-06-02T09:27:19.407Z"

rest-gateway_1
 (node:24) DeprecationWarning: current Server Discovery and
Monitoring engine is deprecated, and will be removed in a future version. To use the new
Server Discover and Monitoring engine, pass option { useUnifiedTopology: true } to the
MongoClient constructor.
 | verbose: connected to mongo at mongodb://db:27017/catapult {
rest-gateway_1

witimestamp":"2021-06-02T09:27:19.443Z"

rest-gateway_1 | info: registering routes {"timestamp":"2021-06-02T09:27:19.445Z"}
rest-gateway 1
 | info: listening on port 3000 {"timestamp":"2021-06-02T09:27:19.462Z
→ " }
```

## 3.3.3 ノード間の暗号化通信の更新

本章では、ノード間通信で使用する証明書を更新したい場合の更新方法を説明します。

### 注釈:

AWS MarketPlace 経由にて、mijin Catapult(v.2) を deploy した場合、AWS Systems Manager Parameter Store に初期データがバックアップされています。

以下のパラメータ値が更新対象となり、ノードと AWS Systems Manager Parameter Store で差分が発生 することに留意してください。

差分があることで、動作に影響することはありません。

・/デプロイ時に指定した冠名/shares/new-cert/各ノード/CA/[*].pem

・/デプロイ時に指定した冠名/shares/nemesis_addresses_harvesting.json

・/デプロイ時に指定した冠名/shares/nemesis_addresses_harvesting_voting.json

・/デプロイ時に指定した冠名/shares/nemesis_addresses_harvesting_vrf.json

## 3.3.3.1 mijin Catapult(v.2) のノード間の暗号化通信について

mijin Catapult(v.2) のノード間通信は、**TCP ポート/7900** にて TLS1.3 による SSL 暗号化通信を行い、認 識しているノードのみ正しいノードとして通信を行います。

SSL 通信には、自己署名の証明書が各ノードに適用されており、ノードは全てのノードの自己署名の証明 書の KeyPair の公開鍵を事前に登録しています。

また、この自己署名の証明書から作成された KeyPair は、ブロックチェーンを生成できる権限を持つアカ ウントとして使用されます。



## 3.3.3.2 ノードの SSL 証明書の更新方法

ノード間で使用する SSL 証明書の更新の手順は以下です。

- 1. CA 及び署名ノード SSL 証明書作成 (ノード間の通信で使用)
- 2. 1の SSL 証明書の KeyPair から秘密鍵を取り出し、mijin Catapult(v.2) ブロックチェーン上にノード のブロック生成を有効にするトランザクションを発行
- 3. 2 の秘密鍵に紐づいた秘密鍵とファイナライズ用 dat ファイルを作成し、2 の秘密鍵と紐づけるトラ ンザクションを発行
- 4. 該当ノードにて、SSL 証明書と dat ファイルの置き換える。
- 5. 全ノードのコンフィグにある対象ノードの公開鍵を2の鍵に置き換える。

#### 警告:

2022/10 現在、SSL 証明書の更新手順は複雑であり、環境ごとに異なるため、mijin Support にお問合せください。

将来、ツールによる容易な更新方法を提供する予定です。

## 3.4 mijin Catapult(v.2) データディレクトリ構造

mijin のデータ構造について説明します。

## 3.4.1 データ配置のディレクトリ

ディレクトリ	前明
/home/catapult/mijin-catapult-package	mijin の config ファイルなどパッケージに関するディレクトリ
/mnt/mijin/blocks	mijin のブロックデータディレクトリ
/mnt/mijin/mongo	mijin の mongo データディレクトリ

## 3.4.2 mijin パッケージの構造

## 3.4.2.1 API ノード

/home/catapult/mijin-catapult-package
├── default # mijin 構築時に使用するディレクトリ (データとしては不要) Directory to be used when
building mijin (not required as data)
- catapult
bin-mount
│  │  │  └── waitmongo
mongo
mongoDbDrop.js
mongoDbPrepare.js
mongoDeploy.sh
mongoLockHashDbPrepare.js
mongoLockSecretDbPrepare.js
mongoMetadataDbPrepare.js
mongoMosaicDbPrepare.js
mongoMultisigDbPrepare.js
mongoNamespaceDbPrepare.js
mongoRestrictionAccountDbPrepare.js
mongoRestrictionMosaicDbPrepare.js
mongors.sh
nemgen
nemgen.sh
Scripts
prepare.sh
- runServerRecover.sh
- startApiServer.sh
│
startServer.sh
tools
└── clean-all.sh
clean-data.sh
L bin-mount
—— package # mijin 構築パッケージ Package for building mijin
L api # ani /ード用のパッケージ Package for ani pode
catapult
L docker-compose yml # docker コンテナを記動する compose ファイル compose file
to start docker container
scripts
l l hrenare sh
$\mu$ prepare.sn   $\mu$ recover sh # ブロックリセット時に使用するスクリプト carint to be used when





## 3.4.2.2 PEER ノード







## 3.4.3 ブロックデータの構造



## 3.4.4 Mongo データ構造

<u> </u>	collection-1381310205274663118138.wt
	collection-1451310205274663118138.wt
	collection-1501310205274663118138.wt
	collection-1621310205274663118138.wt
	collection-1671310205274663118138.wt
 	collection-21310205274663118138 wt
	collection-281310205274663118138 wt
	collection=35==1310205274663118138 wt
	collection $-4 - 1310205274663118138$ wt
 	collection $= 1310205274663118138$ wt
	collection 50 1510205274603110130.wt
	collection = 66 = -1210205274663118138. wt
	collection = 71 = -1210205274663118138. wt
	collection-721210205274663118138.wt
	collection-/813102052/4003110130.wt
	COTTECTION-813102052/4663118138.wt
	COTTECTION-913102052/4663118138.wt
	COllection-9313102052/4663118138.Wt
	diagnostic.data
	index-11310205274663118138.wt
	index-13/14664905013916938.wt
	index-101310205274663118138.wt
	index-1021310205274663118138.wt
	index-1041310205274663118138.wt
	index-1061310205274663118138.wt
	index-1091310205274663118138.wt
	index-111310205274663118138.wt
	index-1101310205274663118138.wt
	index-1131310205274663118138.wt
	index-1161310205274663118138.wt
	index-1171310205274663118138.wt
	index-1201310205274663118138.wt
	index-1231310205274663118138.wt
	index-1241310205274663118138.wt
	index-1271310205274663118138.wt
	index-131310205274663118138.wt
	index-1301310205274663118138.wt
	index-1311310205274663118138.wt
	index-1341310205274663118138.wt
	index-1361310205274663118138.wt
	index-1391310205274663118138.wt
	index-1401310205274663118138.wt
	index-1431310205274663118138.wt
	index-1461310205274663118138.wt
	index-1471310205274663118138.wt
	index-1511310205274663118138.wt
	index-1521310205274663118138.wt
	index-1541310205274663118138.wt
	index-1561310205274663118138.wt
	index-1581310205274663118138.wt
	index-161310205274663118138.wt
	index-1601310205274663118138.wt
	index-1631310205274663118138.wt
	index-1641310205274663118138.wt
	index-1681310205274663118138.wt
	index-1691310205274663118138.wt
	index-191310205274663118138.wt
<u> </u>	index-221310205274663118138.wt

· · · ·	
├── index-291310205274663118138.wt	
├── index-31310205274663118138.wt	
├── index-301310205274663118138.wt	
└── index-321310205274663118138.wt	
└── index-361310205274663118138.wt	
└── index-371310205274663118138.wt	
index-391310205274663118138.wt	
index-411310205274663118138.wt	
index-441310205274663118138.wt	
index-461310205274663118138.wt	
index-481310205274663118138.wt	
index-51310205274663118138.wt	
index-501310205274663118138.wt	
index-521310205274663118138.wt	
index-541310205274663118138.wt	
index-571310205274663118138.wt	
index-581310205274663118138.wt	
index-61310205274663118138.wt	
index-621310205274663118138.wt	
index-631310205274663118138.wt	
index-671310205274663118138.wt	
index-681310205274663118138.wt	
index-721310205274663118138.wt	
index-731310205274663118138.wt	
index-751310205274663118138.wt	
index-791310205274663118138.wt	
index-801310205274663118138.wt	
index-821310205274663118138.wt	
index-841310205274663118138.wt	
index-871310205274663118138.wt	
index-891310205274663118138.wt	
index-911310205274663118138.wt	
index-941310205274663118138.wt	
index-951310205274663118138.wt	
index-971310205274663118138.wt	
index-991310205274663118138.wt	
journal	
- mongod.lock	
- sizeStorer.wt	
storage.bson	
-	

# 3.5 mijin Catapult(v.2) 環境構築オプション表

通常 mijin 環境の構築は Ansible を使用しており、Playbook の ansible オプションをここで示します。

No	項目名	Default 值	入力値	説明
1	service	peer	peer,api	
				ansible で構築するモードを指定します。 [peer] peer モードで構築します [api] api モードで構築します

次のページに続く

No	項目名	Default 值	入力値	説明
2	share_mode	dir	dir,ssm,s3	
				Nemesis ブロック作成時に使用した初期アドレ スを配置します
				「dir] share directory のみに保存されます。
				[ssm] AWS SSM パラメータストアに保存します。
				[s3] AWS S3 に保存します。
3	aws_region	ap-northeast-1	String	
				AWS のリージョンを指定します。リージョンを
				指定することで、S3 及ひ SSM への取得に使わ わます。
				<a href="https://docs.aws.amazon.com/ja_jp/">https://docs.aws.amazon.com/ja_jp/</a>
				AWSEC2/latest/UserGuide/
				using-regions-availability-zones.html# concepts-available-regions>
				concepts available regions.
4	ssm_ps_name			share_mode: ssm の場合、SSM パラメータスト アに保存するパスを指定します。
5	s3_bucket_name			share_mode: s3 の場合、S3 バケット名を指定 します。
6	api_dual_mode	FALSE	true,false	
				service: api の場合、dual モードで起動するか を指定します。
				[Yes] dual モードで起動します。(Harvest の有 かん)
_		[100.100.00.101]		
(	api_nosts	[192.168.96.131]	Array[String, String]	API ノートのホスト名文は IP アトレスを配列で 指定します。
8	peer_hosts	[192.168.96.132, 192.168.96.133]	Array[String, String]	PEER ノードのホスト名又は IP アドレスを配列   で指定します。
9	unix_user	catapult	String	mijin を立ち上げるユーザーを指定します。
10	repo_name	mijin-catapult-	String	
		раскаде		mijin のアプリケーションを配置するディレクト
				リを指定します。
				/home/[unix_user]/[repo_name]
11	catapult_version	v10038	v10038,v10037	
				catapult のバージョンを指定します。
				v10038
12	network identifier	mijin	miiin.miiin-	
	-	,	test	す。
13	mongo_docker_version	4.2.5	String	
				API ノードが使用する mongo のコンテナイメー
				ジのハージョンを指定します。
				<pre>&gt;nttps://nub.docker.com/_/mongo?tab=tags&gt;</pre>
14	mongo_host	db	String	API ノードの mongo コンテナ名を指定します。
15	mongo_max_connection	7	Int	API ノードの mongo への接続制限を指定しま す。
16	mongo_base_retry_delay	750	Int	API ノードの mongo への接続リトライ時間を指 定します。
17	python_docker_version	3.9.0	String	python の docker version を指定します。
18	enable_pip_install	FALSE	true,false	pip install を実行するかを指定します。
19	share_directory	/opt/mijin/shares	String	
				Nemesis ブロック作成時にアドレスデータを保
				管するディレクトリを指定します。
				s3/ssm を指定しても、一時保存場所として使用
				C116 90
20	block_directory	/opt/mijin/blocks	String	各ノードのブロックデータの保存ディレクトリ
L			-	を指定します。
21	mongo_directory	/opt/mijin/mongo	String	API ノードの mongo データの保存ディレクトリ   を指定します。

表 1 – 前のページからの続き

次のページに続く

		衣1-前のページ.	いらの形は	
No	項目名	Default 值	入力値	説明
22	docker_network_range	172.20.0.0/24	String	docker が使用するネットワークレンジを指定し ます
23	rest_ip_address	172.20.0.9	String	docker ネットワーク上での rest-gateway コン     テナのIP アドレフです
24	domain	mijin.internal	String	各ノード間の通信で使用する SSL 証明書の CN を指定します。
25	node_port	7900	Int	各ノード間通信で使用する TCP ポート番号を指
26	enable cache database storage	TRUE	true.false	定します。
20	enable_caene_cacabase_storage			キャッシュデータ保存を有効化します。 無効化すると処理性能が向上する場合がありま
				す。
27	enable_auto_sync_cleanup	TRUE	true,false	一時同期ファイルの自動削除を指定します。
28	base_namespace	cat	String	基軸通貨で使用するルートネームスペースを指 定します。
29	base_currency_name	currency	String	基軸通貨で使用するサブネームスペースを指定   します。
30	base_harvest_name	harvest	String	Harvest モザイクのネームスペースを指定しま す。
31	currency_supply	8'998'999'998'000'000	String	基軸通貨の発行量を指定します。クォート付き。
32	harvest_supply	15'000'000	String	Harvest モザイクの発行量を指定します。
33	block_generation_target_time	15s	String	ブロック生成間隔を指定します(例: 15s)
34	rest_gateway_private_key_num	1	Int	rest_gateway のアドレス作成数を指定します。
35	nemesis_generation_hash_num	1	Int	Nemesis 用の GenerationHash アドレス作成数。
36	nemesis_addresses_harvesting_num	4	Int	Harvest 用アドレス作成数(ノード数と同数必 要)
37	nemesis signer private key num	1	Int	署名用アドレス作成数(通常は1)
38	nemesis addresses num	10	Int	空アドレス(未使用)の作成数。
39	transaction_selections_trategy	oldest	String	oldest, maximize-fee, minimize-fee などから選択。
40	max_time_behind_pull_transaction_start	5m	String	Pull トランザクションの最大遅延許容時間。
41	min_fee_multiplier	100	Int	最小手数料乗数。0 で無料。
42	default_dynamic_fee_multiplier	1'000	String	動的手数料のベース係数。
43	root_namespace_rental_fee_per_block	1	Int	ルートネームスペースのブロック単位のレンタ ル料。
44	child_namespace_rental_fee	100	Int	子ネームスペースのレンタル料。
45	mosaic_rental_fee	500	Int	モザイクのレンタル料。
46	rest_throttling_burst	100	Int	バースト時の REST 接続最大数。
47	rest throttling rate	30	Int	通常時の REST 接続処理レート。
48	unconfirmd cache max response size	20MB	String	未承認トランザクション受信の最大サイズ。
49	unconfirmd_cache_max_size	5MB	String	未承認トランザクションのキャッシュ最大サイズ。
50	cache_max_response_size	20MB	String	(旧バージョン向け)未承認 Tx の応答最大サイ ズ。
51	cache_max_size	50'000	String	(旧バージョン向け)キャッシュ Tx の最大数。
52	block_disruptor_slot_count	4096	Int	ブロック処理用スロット数。
53	block_element_trace_interval	1	Int	ブロック要素のトレース間隔。
54	block_disruptor_max_memory_size	300MB	String	ブロックディスラプターの最大メモリ。
55	transaction disruptor slot count	8192	Int	トランザクション処理用スロット数。
56	transaction element trace interval	10	Int	トランザクション要素のトレース問題。
57	max transaction per block	6'000	String	1ブロック内の最大トランザクション数
58	min transaction failures count for ban	8	Int	
50	min transaction failures percent for bar	10	Int	トランザクション牛b家の RAN 關値
50	partial cache may response size	5MR	String	コンシリンションへ成年の DAN 國胆。 如公トランザクション内然中ノブの早上店
61	partial_cache_max_size	20MB	String	<ul> <li>         ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・</li></ul>
62	enable_finalization	TRUE	true,false	ファイナライゼーションプラグインを有効化す るか。
63	max_rollback_blocks	0	Int	ロールバック可能な最大ブロック数(0 で確定 的)。
64	enable_voting	TRUE	true,false	Voting機能を有効にするか。
65	voting_set_grouping	160	Int	Voting ラウンドのブロック数 (importance_groupingの倍数)。
66	votingkey_start_epoch	1	Int	VotingKey の最小有効期間(エポック単位)。

表 1 - 前のページからの続き

次のページに続く

No	項目名	Default 值	入力値	説明
67	votingkey_end_epoch	26280	Int	VotingKey の最大有効期間(例: 約 821 日)。
68	voting_key_dilution	128	Int	投票キー希釈レベル(再利用の間隔)。
69	enable_revote_on_boot	FALSE	true,false	再起動時に自動的に再投票するか。
70	importance_grouping	40	Int	Importance ラウンド数(影響度スコア更新間 隔)。
71	max_transaction_lifetime	24h	String	トランザクションの有効期間(例: 24h)。
72	max_block_future_time	500ms	String	未来ブロック受付の最大許容時間。
73	max_transactions_per_aggregate	1'000	String	アグリゲート Tx に含められる最大 Tx 数 (Symbolでは100)。
74	max_cosignatures_per_aggregate	25	Int	アグリゲート Tx に署名できる最大署名数。
75	max_bonded_transaction_lifetime	48h	String	アグリゲートボンデッド Tx の有効期間。
76	locked_funds_per_aggregate	10'000'000	String	アグリゲート Tx のロック保証金。
77	max_hash_lock_duration	2d	String	ハッシュロックの有効期間。
78	max_secret_lock_duration	30d	String	シークレットロックの最大有効期間。
79	min_proof_size	1	Int	シークレットプルーフの最小バイト数。
80	max_proof_size	1000	Int	シークレットプルーフの最大バイト数。
81	max_meta_value_size	1024	Int	メタデータ Value の最大サイズ(バイト)。
82	max_cosignatories_per_account	25	Int	アカウントの連署者数の上限。
83	max_cosigned_accounts_per_account	25	Int	1 アカウントが連署できるアカウント数の上限。
84	max_multisig_depth	3	Int	マルチシグの階層の深さの上限。
85	max_mosaics_per_account	1'000	String	1 アカウントが保有可能なモザイク数。
86	max_mosaic_duration	3650d	String	モザイクの最大有効期間(日単位)。
87	max_mosaic_divisibility	6	Int	モザイクの小数点以下の最大桁数。
88	max_name_size	64	Int	ネームスペース名の最大長(文字数)。
89	max_child_namespaces	256	Int	親ネームスペースが持てる子ネームスペース数。
90	max_namespace_depth	3	Int	ネームスペース階層の最大深度。
91	min_namespace_duration	1m	String	ネームスペースの最小有効期間。
92	max_namespace_duration	3650d	String	ネームスペースの最大有効期間。
93	namespace_grace_period_duration	30d	String	ネームスペース期限後の猶予期間。
94	max_account_restriction_values	512	Int	アカウント制限に設定できる最大値数。
95	max_mosaic_restriction_values	20	Int	モザイク制限に設定できる最大値数。
96	max_message_size	1024	Int	トランザクションメッセージの最大バイト数。

表 1 – 前のページからの続き