

徹底攻略

試験
番号

LPI 304

Virtualization & High Availability Exam



Linux
Professional
Institute

LPI 出題範囲準拠教材
ロゴ使用承認番号 LPIJ160329

LPI C

Level 3 304 [Version 2.0] 対応

教科書 + 問題集

米山学 [著]

株式会社ソキウス・ジャパン [編]

1冊で合格力が身に付く
受験者必携の書

Version
2.0

対応版

[詳しい解説] + [演習問題] + [模擬試験
問題 2回分]

「最重要ポイント」
がひと目でわかる。



シリーズ

100万部 突破!!

インプレス

この書籍はLPI認定教材ではありません。

LPIロゴの使用に関しては、出版元がLPIのロゴを使用する許可を得ている教材であることを示すものであり、LPIが教材の内容につき何ら保証をするものではありません。

また該当の教材で学習することにより合格を保証するものではありません。

本書は、LPI技術者認定試験（LPIC）レベル3の受験対策用の教材です。著者、株式会社インプレスは、本書の使用によるLPI技術者認定試験（LPIC）レベル3への合格を一切保証しません。

本書の内容については正確な記述につとめました。著者、株式会社インプレスは本書の内容に基づくいかなる試験の結果にも一切責任を負いません。

Linuxは、Linus Torvaldsの米国およびその他の国おける登録商標または商標です。

その他、本文中の製品名およびサービス名は、一般に開発メーカーおよびサービス提供元の商標または登録商標です。なお、本文中には™、®、©は明記していません。

インプレスの書籍ホームページ

書籍の新刊や正誤表など最新情報を随時更新しております。

<http://book.impress.co.jp/>

Copyright © 2016 Socius Japan, Inc. All rights reserved.

本書の内容はすべて、著作権法によって保護されています。著者および発行者の許可を得ず、転載、複写、複製等の利用はできません。

まえがき

近年のシステム開発では、クラウドファーストという言葉が表すように、システム基盤やアプリケーションプラットフォームの構築において、まずはクラウドの採用を検討するという事例が増えています。現代のITシステムはクラウドを抜きにして語ることはできないと考えても間違いのないでしょう。

本書が取り上げる「LPI 304 Virtualization & High Availability Exam」では、そのクラウドシステムの裏側を支える最も重要な「仮想化技術」と「高可用性技術」を理解し、それらのシステムを構築・運用することのできるスキルを習得することを目標としています。

本試験に合格するための学習を通じて仮想化の概念や基本的な技術の仕組みを理解し、Linux上でオープンソースソフトウェアを利用した仮想化システムや負荷分散システムを構築することができるようになるでしょう。また、クラスタ管理やクラスタストレージの管理・運用のスキルを習得することができます。

本書は試験対策向けということでポイントを絞って解説しています。この試験の学習をきっかけに、ぜひとも読者の皆さん各自が、個々の技術に興味を持ってさらなる探求を進めていってもらえることを願っています。

最後に、いつも理想的なサポートと編集をしていただけるソキウス・ジャパンと坂田様および作業に関わったすべての関係者の方々に御礼を申し上げます。

2016年4月 著者 米山 学

LPI技術者認定試験について

LPIC (エルピック) はLPI (Linux Professional Institute : Linux技術者認定機関) が実施している世界共通のLinux技術者認定制度です。特定のLinuxディストリビューションに依存しない汎用性の高い認定資格制度であることが知られています。2016年1月時点では日本国内での累計受験者数は27万人を超え、9万6,000人以上が認定を取得している人気の資格です。

LPICのレベルには、「LPIC-1 (LPICレベル1)」「LPIC-2 (LPICレベル2)」「LPIC-3 (LPICレベル3)」の3つがあります。2016年3月現在のLPIC資格体系は以下のとおりです。

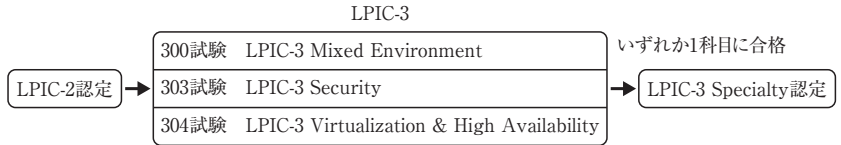
【LPIC資格体系】

資格	試験 (バージョン)	認定されるスキル
LPIC-1	101試験 (4.0)	実務に必要なLinuxの基本操作とシステム管理が行えるエンジニアであること
	102試験 (4.0)	
LPIC-2	201試験 (4.0)	Linuxのシステムデザイン、ネットワーク構築において、企画、導入、維持、トラブルシューティング、キャパシティプランニングができるエンジニアであること
	202試験 (4.0)	
LPIC-3	300試験 (1.0)	Linux、Windows、Unixが混在するシステムの設計、構築、運用・保守ができるエキスパートエンジニアであること
	303試験 (1.0/2.0)*	セキュリティレベルの高いコンピュータシステムの設計、構築、運用・保守ができるエキスパートエンジニアであること
	304試験 (2.0)	クラウドコンピューティングシステム (クラウド) の設計、構築、運用・保守ができるエキスパートエンジニアであること

LPIC-3 と 304 試験について

現在、LPICレベル3に対応する試験は、「LPIC-3 Mixed Environment (300試験)」「LPIC-3 Security (303試験)」「LPIC-3 Virtualization & High Availability (304試験)」の3つが実施されています。3試験のうちいずれかに合格し、かつ、有意なLPICレベル2の認定を持っていると、「LPICレベル3 Specialty」に認定されます。

【LPIC-3資格体系】



※ バージョン1.0は2016年8月31日で配信終了

本書では、LPIC-3の試験科目「LPIC-3 Virtualization & High Availability (304試験)」を扱います。LPIC-3 Virtualization & High Availabilityに合格することによって、クラウドコンピューティングシステム（クラウド）の設計、構築、運用・保守ができるエキスパートエンジニアであることを証明できます。

●304試験 (LPI 304 Virtualization & High Availability Exam)

- ・ 試験時間 : 90分
- ・ 問題数 : 約60問
- ・ 受験料 : 30,000円 (税抜き)

●出題範囲について

304 試験では、仮想化および高可用性に関する知識とスキルを問われる問題が出題されます。出題範囲は、「仮想化」「高可用クラスタ管理」「高可用クラスタストレージ」の3つのカテゴリに分かれています。

出題範囲の各項目には重要度が記されています。この重要度はそれぞれの目標の相対的な重要性を重み付けしたものであり、重みの範囲は1～10で設定されています。重要度が高い項目ほど、試験において多くの問題が出題される傾向にあります。

主題 330 : 仮想化

- 330.1 仮想化の概念と理論 (重要度 : 8)
- 330.2 Xen (重要度 : 9)
- 330.3 KVM (重要度 : 9)
- 330.4 その他の仮想化ソリューション (重要度 : 3)
- 330.5 Libvirtと関連ツール (重要度 : 5)
- 330.6 クラウド関連ツール (重要度 : 2)

主題 334 : 高可用クラスタ管理

- 334.1 高可用性の概念と理論 (重要度 : 5)
- 334.2 ロードバランスクラスタ (重要度 : 6)
- 334.3 フェイルオーバークラスタ (重要度 : 6)
- 334.4 エンタープライズLinuxディストリビューションにおける高可用性 (重要度 : 1)

主題 335 : 高可用クラスタストレージ

- 335.1 DRBD/cLVM (重要度 : 3)
- 335.2 クラスタファイルシステム (重要度 : 3)

【304試験の出題範囲詳細】

<http://www.lpi.or.jp/lpic3/range/range304.shtml>

●試験方法

試験会場に用意されたコンピュータを使って行われます。試験問題のほとんどはマウスによる多岐選択方式ですが、キーボードから入力して解答する問題も一部出題されます。

●合否結果と資格認定

試験終了後、合否と得点が印刷されたスコアレポートが渡されます。

●認定要件

LPIC-3の試験科目のうちいずれかに合格し、かつ、有意なLPICレベル2の認定を持っていること。

受験申し込み方法

●LPI-IDの取得

LPI認定試験の受験を申し込むには、事務局が指定するWebページからLPI-IDを取得する必要があります。

- ・登録用Webページ……<https://bloom.lpi.or.jp/caf/Xamman/register>

●受験申し込み先

LPI認定試験は、ピアソンVUEのWebサイトまたはコールセンターを利用して申し込みができます。いずれの場合も希望する受験会場および受験日時を選択することができます。ただし、予約状況によっては希望どおりの会場や日時を選択できない場合もありますので、必ず予約状況を確認してください。

●ピアソンVUE

URL : <http://www.pearsonvue.co.jp/>

TEL : 0120-355-173 または 0120-355-583

初めてピアソンVUEに申し込む場合は、受験者情報を登録する必要があります。その後、以下のいずれかの方法で申し込むことができます。

・ Webからの申し込み

<http://www.pearsonvue.co.jp/>からログインして予約

・ 電話での申し込み

上記電話番号に直接申し込み（受付時間：平日午前9時～午後6時）

・ テストセンターに直接申し込み

テストセンター一覧から受験希望のテストセンターを確認したうえで直接連絡

LPI-Japanの連絡先

〒106-0041 東京都港区麻布台1-11-9 CR神谷町ビル 7F

TEL :03-3568-4482

FAX :03-3568-4483

E-Mail :info@lpi.or.jp

URL :<http://www.lpi.or.jp/>

最新の試験情報は、必ず上記のWebサイトで確認してください。

LPI再受験ポリシーについて

2010年11月18日より、LPI認定試験に関する再受験ポリシーが下記のとおり実施されています。最新の再受験ポリシーについては、必ずLPI-JapanのWebサイトで確認してください。

- ・ LPIC同一科目を再受験する際、2回目の受験については、受験日の翌日から起算して7日目以降（土日含む）より可能
- ・ 3回目以降の受験については、最後の受験日の翌日から起算して30日目以降より可能
- ・ 一度合格した科目については、受験日から2年間は再受験できない
- ・ 受験後2年間を経た場合、再受験ポリシーはいったんクリアになり、その後の同一試験の最初の再受験は一度目の受験としてカウントされる

本書の特長と活用方法

本書は「LPI 304 Virtualization & High Availability Exam (LPICレベル3 304試験)」の合格を目指す方を対象とした受験対策教材です。第1章から第3章は、出題範囲に沿った解説・演習問題で構成されています。第4章と第5章は、模擬問題の位置付けとなる「総仕上げ問題」です。各章で学習したのちに、実戦形式の総仕上げ問題で受験対策の仕上げをしましょう。

【教科書と問題集を兼ね備えた1冊】

解説は出題範囲を完全網羅し、出題傾向に沿ってわかりやすくポイントを押さえています。また、各セクションの演習問題と模擬問題2回分で合計245問の問題を掲載していますので、この1冊で万全の受験対策が可能です。

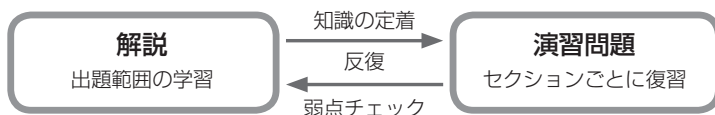


【試験対策がわかりやすい】

本書は、実際の試験を徹底分析した結果を反映した内容になっています。本文中の「試験対策」欄には試験の重要項目が記載されていますので、効率的に学習を進めることができます。

【演習問題で知識を定着】

第1章から第3章までは、出題範囲のカテゴリ別の章立てで、出題傾向に沿って解説しています。各セクションには復習を目的とした演習問題を設け、「解説を読み終えてから記憶が薄れないうちに演習問題を解く」を繰り返して知識を確実に定着させます。演習問題の解答には解説ページへの参照が記載されていますので、不正解の問題は本文に戻って復習することができます。



【模擬問題2セット分を掲載】

第4章と第5章の「総仕上げ問題」には、本試験と同レベルの問題を各60問掲載しています。本試験と同じ90分で解いてみましょう。正解した問題はチェックボックスにチェックしていき、全問正解できれば合格目前です。不正解の問題は、本文に戻って見直しをしましょう。

●解説

重要語句、重要事項

本文中の重要用語や重要語句は太字で示しています。

参考

試験対策とは直接関係はありませんが、知っておくと有益な情報を示しています。

試験対策

試験のために理解しておかなければならないことや、覚えておかなければいけない重要事項を示しています。

RHELの高可用性アドオン

大規模システムに対応した商用サポートのあるエンタープライズLinuxの分野ではRed Hat社が提供する「RHEL (Red Hat Enterprise Linux)」が圧倒的な市場シェアを持っており、RHELではエンタープライズシステムの可用性を高めるための「**High Availability Add-On (高可用性アドオン)**」というソフトウェアコンポーネントのセットを提供しています。このアドオンは、クラスタ管理の基盤を提供する**CMAN**や**DLM**を中心に以下のようなコンポーネントから構成されています。

【RHEL High Availability Add-Onの構成要素】

アドオン	構成要素
クラスタインフラストラクチャ	・ CMAN (Cluster Manager) ・ DLM (Distributed Lock Manager) ・ フェンシング機能
高可用性サービス管理	RGManager (Resource Group Manager)
クラスタ管理ツール	・ Conga (クラスタ設定と管理のための包括的なUI) ・ Luci (Conga UIを提供するアプリケーションサーバ) ・ Ricci (クラスタ設定の分配を管理するためのサービスデーモン)



RHEL High Availability Add-Onは従来、「Red Hat Cluster」と呼ばれていました。



RHEL High Availability Add-Onは、フェイルオーバークラスタやロードバランスクラスタのサポート、ストレージコンポーネントなど、他のLinuxディストリビューションにおいても一般的に利用可能なコンポーネントをベースに構成されています。

●演習問題・解答

問題

問題は選択式もしくは記述式です。

1 仮想化のメリットに関する説明として正しいものを2つ選びなさい。

- A. ユーザーはインターネットを介してアプリケーションを利用できる
- B. 物理的なマシン上で独立した複数のOSを動かすことができる
- C. システムのパフォーマンスを向上させることができる
- D. サーバの運用コストを削減することができる
- E. セキュリティの強度を高めることができる

解答

正解の選択肢は太字で示しています。
関連する内容の記載されたページを示しています。

1 B, D

仮想化テクノロジーによって、ユーザーは物理的なハードウェアの制約に縛られることなく、さまざまなリソースを利用することができるようになります。具体的には、1台のマシン上で複数のOSを動かすことができます(選択肢B)。また、仮想化によって提供される仮想マシンは論理的なマシンであり、簡単に作成したりコピーや削除したりすることができます。これにより、運用コストを削減することができます(選択肢D)。(▶P18)。

●総仕上げ問題

チェックボックス

確実に理解している問題のチェックボックスを塗りつぶしながら問題を解き進めると、2回目からは不確かな問題だけを効率よく解くことができます。すべてのチェックボックスが塗りつぶされれば合格は目前です。

選択式の問題

正解を1つだけ選ぶものと複数選ぶものがあります。

2. P2Vにおいて仮想マシンにコピーしなければならないデータとして正しいものを選びなさい。

- A. ファイルシステムおよびそれらのコンテンツ
- B. CPUフラグ
- C. ブロックデバイスバッファ
- D. 仮想マシンの設定ファイル

→ P222

解答ページ

問題の右下に、解答ページが表示されています。ランダムに解くときも、解答ページを探すのに手間取ることがありません。

記述式の問題

問題文を読んで、コマンド名、オプション、ファイル名などを記述するものがあります。

18. 仮想マシンにアタッチされたCD-ROMイメージファイルを変更するために使用するvirshのサブコマンドを記述しなさい。

→ P225

問題ページ

解答の右端に、問題ページが表示されています。確認したい問題を簡単に探すことができます。

解答

正解の選択肢は太字で示されています。

7. C → P207

XenのDomain-0に割り当てられるメモリの総容量はブートローダの設定ファイルに記述します。したがって、選択肢**C**が正解です。なお、GRUB 1とGRUB 2では設定の記述の仕方が異なります。

参照 → P.44

本文のページ

関連する本文のページが表示されています。正解できなかった問題は、本文に戻って復習しましょう。

目次

まえがき	3
LPI技術者認定試験について	4
受験申し込み方法	6
本書の特長と活用方法	8

第 1 章 仮想化

1-1 仮想化の概念と理論	16
仮想化とは?	16
仮想化のメリットとデメリット	18
仮想化方式の種類	19
仮想マシンの移行 (マイグレーション)	22
準仮想化と完全仮想化	23
CPUモードと特権モード	25
CPUの仮想化支援機能とCPUフラグ	27
クラウドコンピューティング	28
クラウドコンピューティングのサービスモデル	30
演習問題	33
解答	37
1-2 Xen	40
Xenとは?	40
Xenの基本的なアーキテクチャ	41
Xenの設定	43
Xenのユーティリティとツールスタック	46
XenStore	51
演習問題	53
解答	57
1-3 KVM	60
KVMとは?	60
KVMの基本的なアーキテクチャ	60
KVMのセットアップ	62
KVMのユーティリティ	63
KVMのネットワーキング	69
演習問題	71
解答	75

1-4	その他の仮想化ソリューション	78
	OpenVZ	78
	LXC	80
	その他の仮想化テクノロジー	81
	演習問題	86
	解答	88
1-5	Libvirtおよび関連ツール	90
	libvirtとは?	90
	libvirtの基本アーキテクチャ	91
	virshによるlibvirtの操作	94
	oVirt	99
	演習問題	101
	解答	103
1-6	クラウド管理ツール	105
	OpenStack	105
	CloudStack	107
	その他のクラウド構築・管理ツール	108
	演習問題	109
	解答	111

第 2 章 高可用クラスタ管理

2-1	高可用性の概念と理論	114
	高可用クラスタとは?	114
	ロードバランスクラスタ	116
	フェイルオーバークラスタ	118
	クラスタアーキテクチャの設計	119
	演習問題	124
	解答	127
2-2	ロードバランスクラスタ	129
	LVSとは?	129
	keepalived	134
	ldirectord	137
	HAProxy	139
	演習問題	145
	解答	149

2-3	フェイルオーバークラスタ	151
	Pacemakerとは?	151
	Pacemakerのアーキテクチャ	153
	リソースクラス	156
	Pacemakerの管理とユーティリティ	157
	リソースの制約	162
	STONITH対応デバイス	165
	Corosyncの設定と管理	166
	演習問題	168
	解答	171
2-4	エンタープライズLinux	
	ディストリビューションの高可用性	173
	RHELの高可用性アドオン	173
	SUSE Linuxエンタープライズの	
	高可用性拡張	174
	演習問題	175
	解答	176

第3章 高可用クラスタストレージ

3-1	DRBDおよびcLVM	178
	DRBDとは?	178
	DRBDの管理用ユーティリティ	183
	cLVMとは?	185
	cLVMのユーティリティ	186
	演習問題	188
	解答	190
3-2	クラスタファイルシステム	192
	クラスタファイルシステムとは?	192
	GFS2(Global File System 2)	194
	OCFS2(Oracle Cluster Filesystem ver.2)	195
	O2CBクラスタスタック	197
	その他の分散ファイルシステム	
	(クラスタファイルシステム)	199
	演習問題	201
	解答	203

第4章 総仕上げ問題①

問題	206
解答	222

第5章 総仕上げ問題②

問題	236
解答	252
索引	265

第1章

仮想化

- 1-1 仮想化の概念と理論
 - 1-2 Xen
 - 1-3 KVM
- 1-4 その他の仮想化ソリューション
 - 1-5 Libvirtおよび関連ツール
 - 1-6 クラウド管理ツール

1-1 仮想化の概念と理論

この節では仮想化の基本的な仕組みと用語について、また仮想化を活用するクラウドコンピューティングについて学習します。

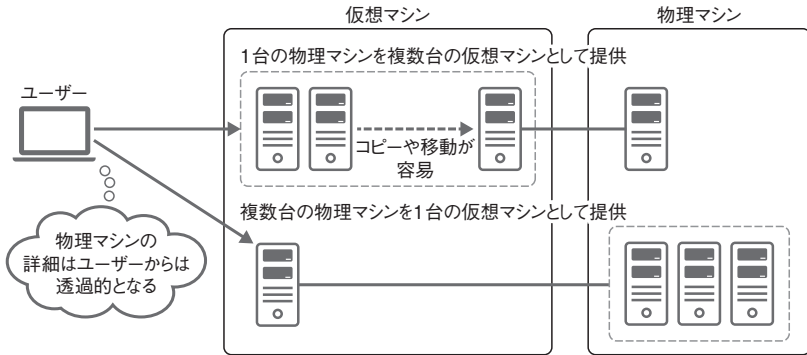
仮想化とは？

仮想化 (Virtualization) とは、**物理的なハードウェアリソース** (CPUやメモリ、ストレージなど) をユーザーから隠蔽し、代わりとなる**論理的なリソース**を提供するための技術・仕組みです。仮想化される個々のハードウェアリソースは、その対象に応じて「仮想CPU」や「仮想メモリ」、「仮想ネットワーク」などと呼ばれますが、1台のコンピュータマシンそのものを仮想化した場合、それらは「**仮想マシン (VM : Virtual Machine)**」や「**仮想ホスト**」あるいは「**仮想サーバ**」などと呼ばれます。

仮想マシンはソフトウェアの仕組みによって物理マシン上に作り出された論理的なマシンですが、ユーザーからは仮想マシンがあたかも物理マシンのように見え、表面的な違いがわからないようになっています (透過的性質)。このような仮想化テクノロジーを用いることによって、1台の物理マシン上で複数の仮想マシンを動かしたり、複数台の物理マシンをあたかも1台の仮想マシンとしてユーザーに提供したりすることができます。また、仮想マシンは**複製や移動が容易**であり、**運用管理の手間を大きく削減**することができます。

このような仮想化テクノロジーは、ハードウェアおよびソフトウェア技術の目覚ましい進歩やインターネットを中心としたアーキテクチャの変遷により、近年急速に普及することとなりました。とりわけ、後述する**クラウドコンピューティング**のアーキテクチャを支えるためのテクノロジーとして、非常に重要なものとなっています。

【仮想マシンと物理マシン】



試験対策

仮想化テクノロジーは物理的なリソースの詳細を隠蔽し、透過的に利用することができる仕組みを提供します。このような仮想化テクノロジーを利用することで、従来の物理マシンだけで構成されたシステムよりも柔軟で拡張性の高いシステムを構築することが可能となります。



参考

仮想化は近年急速に普及した感がありますが、広義の仮想化テクノロジーの歴史は非常に古く、1960年代のコンピュータ黎明期の時代から研究が行われています。



参考

ネットワーク分野では、「SDN (Software Defined Network)」と呼ばれるソフトウェアによるネットワークの仮想化技術が普及してきています。

仮想化のメリットとデメリット

仮想化にはさまざまなメリットがあり、現代のITシステムを語る上で欠かせないテクノロジーとして普及してきたわけですが、実際にシステムに仮想化を導入する際にはメリットだけでなくデメリットに関しても十分に検討する必要があります。

●仮想化のメリット

仮想化の最大のメリットは**システムの柔軟性と拡張性を向上させる**ことであり、同時にその副次的なメリットとして、以下に挙げるようなさまざまな**コスト削減**を実現することができます。

- ・ 物理サーバおよびサーバの設置スペース
- ・ 物理サーバの調達や運用・管理コスト
- ・ 消費電力などのランニングコスト

●仮想化のデメリット

仮想化はメリットばかりが強調されがちですが、導入にはデメリットも考えられます。とりわけ仮想環境では、**性能劣化**が発生しやすいため注意が必要です（特にネットワークやストレージI/O）。また、性能劣化以外にも以下のような事項がデメリットになる可能性を考慮しておきましょう。

- ・ 障害発生時に原因や問題を特定することが困難である
- ・ アプリケーションが仮想化環境で正常に動作するかどうかを検証する必要がある
- ・ 物理環境とは異なる運用管理者のスキル、ノウハウの蓄積が必要である
- ・ 仮想化環境固有のセキュリティの問題

仮想化方式の種類

先述したように、仮想化は主にソフトウェアによって提供される仕組みであり、サーバを仮想化するためのソフトウェアを「仮想化ソフトウェア」や「仮想マシンモニタ(VMM: Virtual Machine Monitor)」とといいます。Linuxでは、XenやKVMという仮想化ソフトウェアが一般的に使われています(XenやKVMについては後述します)。

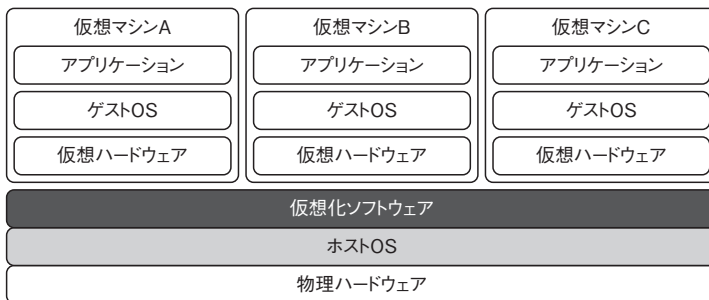
仮想化ソフトウェアの実装方式は、いくつかのタイプに分類することができます。現在一般的に使用されている3つの分類を見ていきましょう(実際には観点によって若干異なる分類も存在しますが、本試験では以下の3つだけ押さえておきましょう)。

- ・ホストOS型
- ・ハイパーバイザー型
- ・コンテナ型

●ホストOS型

ホストOS型は、物理的なハードウェア上にOSをインストールし、そのOS上で仮想化ソフトウェアと仮想マシンを動かす方式です。物理マシン上で直接動作するOSを**ホストOS**、仮想マシンにインストールされるOSを**ゲストOS**とといいます。

【ホストOS型の基本的な仕組み】



ホストOS型の具体的な仮想化ソフトウェアとしては、以下に挙げるようなものが有名です。

- ・VMware Workstation
- ・Microsoft Virtual PC
- ・VirtualBox



サーバの仮想化方式として昔はホストOS型が主流でしたが、現在はあまり使われなくなってきており、次に説明するハイパーバイザー型が主流となっています（クライアントPCではホストOS型が現在でもよく使われています）。

●ハイパーバイザー型

ハイパーバイザー (Hypervisor) 型は、ホストOSを必要としないタイプの仮想化方式です。この方式では**ハイパーバイザー**というソフトウェアを物理ハードウェア上で直接動かし、ハイパーバイザー上で個々の仮想マシンを動かします。

【ハイパーバイザー型の基本的な仕組み】



ハイパーバイザー型仮想化では、ハイパーバイザーは仮想マシンのリソースとホストシステムのリソースを対応付け、仮想マシンがあたかも独立したホストシステムのように振る舞えるようにします。また、ホストシステム上で動く複数の仮想マシンがお互いに干渉しないようにすることもハイパーバイザーの役割です。

ハイパーバイザー型では、ホストOSを介する処理オーバーヘッドが存在しないため、**ホストOS型よりもパフォーマンスが高い**というメリットがあります。現在のサーバの仮想化テクノロジーにおいて主流となっている方式です。

ハイパーバイザー型の具体的な仮想化ソフトウェアとしては、以下に挙げるようなものが有名です。

- ・ VMware ESXi
- ・ Xen/Citrix XenServer
- ・ Microsoft Hyper-V

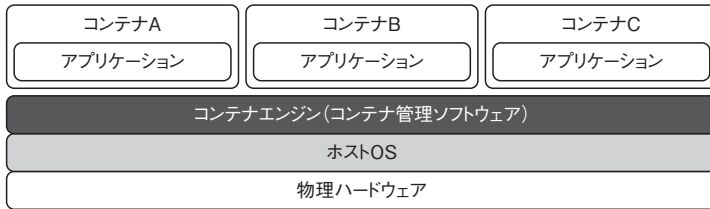


ハイパーバイザーは直接ハードウェア上で動作することから、「**ベアメタル型**」と呼ばれることもあります（Bare metal：むき出しの金属）。

● コンテナ型

コンテナ型は、近年急速に普及してきた比較的新しいタイプの仮想化方式です。コンテナ型はホストOS型と同じように、物理マシンにインストールされているOS上で仮想環境（**コンテナ**）を動かすため、ホストOS型の一種として分類されることもあります。コンテナには**ゲストOSや仮想ハードウェアは含まれない**ことがホストOS型との一番大きな相違点となります（コンテナ間でホストOSを共有）。コンテナ型では一般的に仮想化ソフトウェアのことを「**コンテナエンジン**」や「**コンテナ管理ソフトウェア**」と呼びます。

【コンテナ型の基本的な仕組み】



コンテナ型仮想化では、ホストOSとしてインストールされたLinuxカーネルが同じであれば、それぞれのコンテナを異なるディストリビューションのLinux OSとして動かすことができます。



コンテナのことを「パーティショニング」ということもありますが、パーティショニングの元となったUNIXやLinuxの技術として、ファイルシステムの見える範囲を他のプロセスから分離するためのchrootやcgroups、jail (FreeBSD) などがあります。

コンテナ型の具体的な仮想化ソフトウェアとしては、以下に挙げるようなものが有名です。

- ・ OpenVZ
- ・ LXC
- ・ Linux VServer
- ・ Docker
- ・ Oracle Solaris Zones

【コンテナ (container)】 アプリケーションの実行環境のみを閉じ込める軽量の「容器」という意味で使われる。コンテナ型仮想化ではOSや仮想ハードウェアなどをコンテナに含まないため「仮想マシン」という用語は使われない。



試験対策

コンテナ型の仮想化では仮想環境にゲストOSは含まれず、複数の仮想環境で1つのホストOS (Linuxカーネル) を共有します。したがって、同一ホスト上の複数のコンテナで異なるOSを使用することはできません。ただし、ホストOSと同一のLinuxカーネルであれば、コンテナで異なるLinuxディストリビューションを使用することができます。

ホストOS型やハイパーバイザー型では、仮想マシンにおけるゲストOSの実行やハードウェアエミュレーションなどの処理オーバーヘッドが発生しますが、コンテナ型ではこのようなオーバーヘッドがない分、**パフォーマンスに優れ、リソースの消費量も少なく、構成管理の自動化や運用管理が容易**というメリットがあります。

仮想マシンの移行(マイグレーション)

仮想化によるサーバ統合 (コンソリデーション) では物理サーバを仮想化基盤に移行したり、すでに一部で仮想化が導入されている場合には、物理マシン間で仮想サーバを移行するような運用管理作業が発生します。このような仮想マシンの移行を**マイグレーション**といい、P2VもしくはV2Vという2種類の分類で扱われています。

●P2V

P2Vとは「Physical to Virtual」の略であり、**物理サーバから仮想サーバへのマイグレーション**となります。P2Vは主に、既存の物理サーバの老朽化やサポート切れなどに伴って実施されることが多く、物理サーバの台数を減らすことによって、さまざまなコストや運用管理の手間を削減することができます。



試験対策

物理マシンで稼働しているシステムを仮想マシンにマイグレーションする場合、仮想環境の構築後にファイルシステムおよびそれらのコンテンツをコピーする必要があります。移行前の物理マシンにおけるCPUなどのハードウェアやネットワーク関連のアドレスなどに対しては、仮想リソースが割り当てられることとなります。

●V2V

V2Vとは「Virtual to Virtual」の略であり、**仮想化プラットフォーム間**における仮想マシンのマイグレーションとなります。V2Vでは、物理サーバを停止せずに仮想サーバを移行する「**ライブマイグレーション**」という機能を利用することもできます。



試験対策

仮想マシンにおいてデータを共有する方法としては、NFS（Network File System）やFTP（File Transfer Protocol）を使用するのが一般的です。

1

準仮想化と完全仮想化

ここまで仮想化の代表的な3つの方式を紹介しましたが、ここではさらに、サーバの仮想化において現在主流となっているハイパーバイザー型の仮想化を詳しく見ていきましょう。ハイパーバイザー型は、仮想マシン上で動かすゲストOSのタイプによって、準仮想化と完全仮想化という2つの実装方式に分類することができます。

●準仮想化

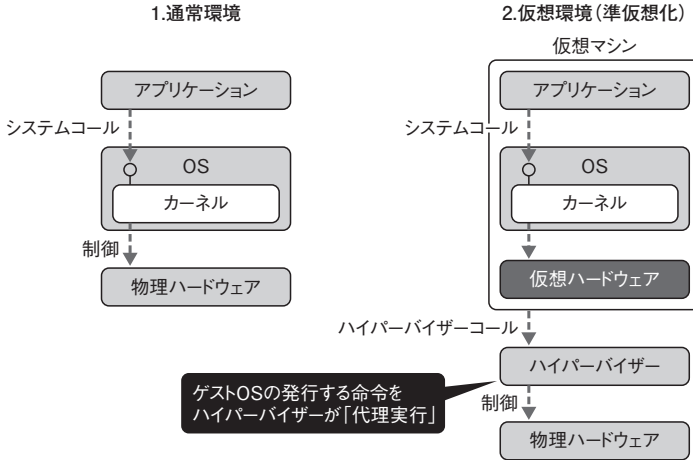
仮想マシン上で動かす**ゲストOSのコードに改変（修正）が必要**となるタイプの仮想化を**準仮想化（PV：Para-Virtualization）**といいます。

仮想化されていない通常的环境では、アプリケーションからのシステムコールはOSのカーネルが処理しますが（次ページの図中1）、準仮想化環境ではゲストOSからのシステムコールをハイパーバイザーが処理します（次ページの図中2）。この命令を**ハイパーバイザーコール**といいます。このため準仮想化では、ゲストOSのカーネルにおけるハードウェア制御のためのシステムコールをハイパーバイザーコールに置き換えるようにコードを改変する必要があり、基本的にLinuxのようなオープンソースのOSを利用することが前提となります。

【システムコール】 カーネルがアプリケーションに対して提供するAPIであり、スーパーバイザーコールともいう。

【ハイパーバイザーコール】 アプリケーションから見た場合に、スーパーバイザーのさらに上位（ハイパー）に対するシステムコールという意味で名付けられた用語。準仮想化におけるハイパーバイザーコールは、完全仮想化に比べてパフォーマンスが高いというメリットがある。

【通常環境のシステムコールと準仮想化環境のハイパーバイザーコール】



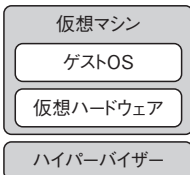
準仮想化では、デバイスドライバはゲストOS上にインストールされます。

●完全仮想化

OSのコードに修正が必要な準仮想化に対して、**完全仮想化 (Full Virtualization)** は**ゲストOSのコードに一切手を加えることなく動かすことができる**タイプの仮想化方式です。完全仮想化方式を利用することで、Windowsのような商用のプロプライエタリなOSを**変更することなくそのままゲストOSとして動かすことができる**ようになります。こうした利便性がある反面、準仮想化に比べて**処理オーバーヘッドが大きく、パフォーマンスが劣化する**可能性があります。

完全仮想化では仮想ハードウェアによる「**ハードウェアエミュレーション**」と呼ばれる仕組みを用いて、I/O命令や割り込み、メモリ管理など、ハードウェアに対する処理をソフトウェアレベルで忠実に再現します。

【完全仮想化】





試験対策

ハードウェアエミュレーションの仕組みにより、マシンアーキテクチャが異なったとしてもOSを実行することができるようになります。

CPUモードと特権モード

CPUが備えている機能をプログラムがどの程度利用できるかを表す概念を**CPUモード**、あるいは**特権レベル**といますが、CPUモードは大きく以下の2つに分類することができます。

- ・ 特権モード（カーネルモード）
- ・ ユーザーモード

特権モードは、CPUが備えるすべての機能を**無制限**に利用できるCPUモードであり、特権モードでのCPUに対する命令は「**特権CPU命令**」とも呼ばれています。完全仮想化ではゲストOSの特権CPU命令をハイパーバイザー上の仮想ハードウェアがエミュレートします。対する**ユーザーモード**は、**制限**が課せられたCPUモードであり、アプリケーションの実行にはこのユーザーモードが使用されます。

通常、**OSやデバイスドライバ**などのプログラムは特権モードで動作しますが、OS上で動作するアプリケーションプログラムが特権CPU命令を実行できてしまうと、リソースの占有や破壊などのようなシステム全体に致命的な問題を発生させてしまう可能性があるため、アプリケーションプログラムはユーザーモードで動作するようになっています。したがって、アプリケーションがハードウェアリソースを操作するような特権CPU命令を発行した場合には例外を発生させ、割り込みによってOSがその処理を代行する仕組みになっています。

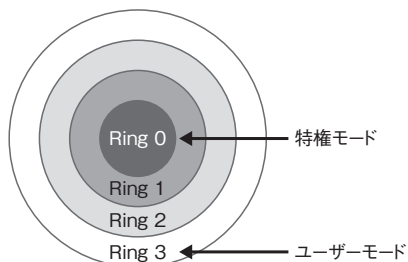
物理マシン上のホストOSであれば特権CPU命令を実行することが可能ですが、仮想マシン上のゲストOSはアプリケーションとなるため、特権CPU命令を直接実行することはできません。この問題を解決するためのテクノロジーとして、ハイパーバイザーコール以外に以下のようなテクノロジーが利用されています。

- ・ バイナリトランスレーション
- ・ CPU仮想化支援機能

【バイナリトランスレーション】 仮想マシンから発行されたカーネルモードのCPU命令を仮想マシンのユーザーモード命令として処理し、その結果を仮想マシンのOSに返す方式。ハードウェアの完全なエミュレーション。トランスレーション処理がオーバーヘッドとなり性能が劣化しやすい。



代表的なIntel互換CPUでは、ユーザーモードをさらに3つのレベルに分割してCPUモード全体をリングのように表しています。この概念は「リングプロテクション」とも呼ばれ、特権モードに近いほうから「Ring 1」～「Ring 3」までの呼称が付けられています。現在のWindowsやLinux OSでは一般的にはRing 1とRing 2は使われていません。



CPUの仮想化支援機能とCPUフラグ

ハイパーバイザーはソフトウェアレベルの仮想化テクノロジーですが、**ハードウェアレベル**で仮想化を支援するための技術も利用されています。具体的には「CPUの仮想化支援機能」と呼ばれるもので、ソフトウェアレベルの仮想化に比べて処理パフォーマンスを向上させることができます。

●CPUの仮想化支援機能

CPUの仮想化支援機能はCPUエミュレーションともいい、ソフトウェアレベルで行っていた仮想化の仕組みをCPUで実現するための拡張機能です。CPU仮想化支援機能の実装として、代表的なCPUベンダーであるIntel社とAMD社は以下のテクノロジーを提供しています。

- ・ Intel VT (Intel Virtualization Technology)
- ・ AMD-V (AMD Virtualization)

●CPUフラグ

CPUフラグ (CPU Flags) は、CPUがサポートしている機能を示すフラグ（目印）のことであり、Linuxシステムでは**/proc/cpuinfo**ファイルにこの情報が格納されます。現在動かしているシステムにおいてCPUの仮想化支援機能が利用できる場合には、/proc/cpuinfoファイル中の「flags」行に「**vmx**」（Intelの場合）もしくは「**svm**」（AMDの場合）という文字列が記述されます。以下のようなコマンドを使用して確認することができます。

例 /proc/cpuinfoファイル内に「vmx」もしくは「svm」の記述があるかどうかを調査

```
# grep '(vmx|svm)' /proc/cpuinfo
# egrep '(vmx|svm)' /proc/cpuinfo
```

あるいは、Xenを使用している場合にはXenの管理コマンドxlを使用して、以下のコマンド例のようにシステムの起動メッセージを調べることで確認可能です。

例 システム起動時のメッセージから、CPUフラグを確認

```
# xl dmesg | grep -i hvm
```